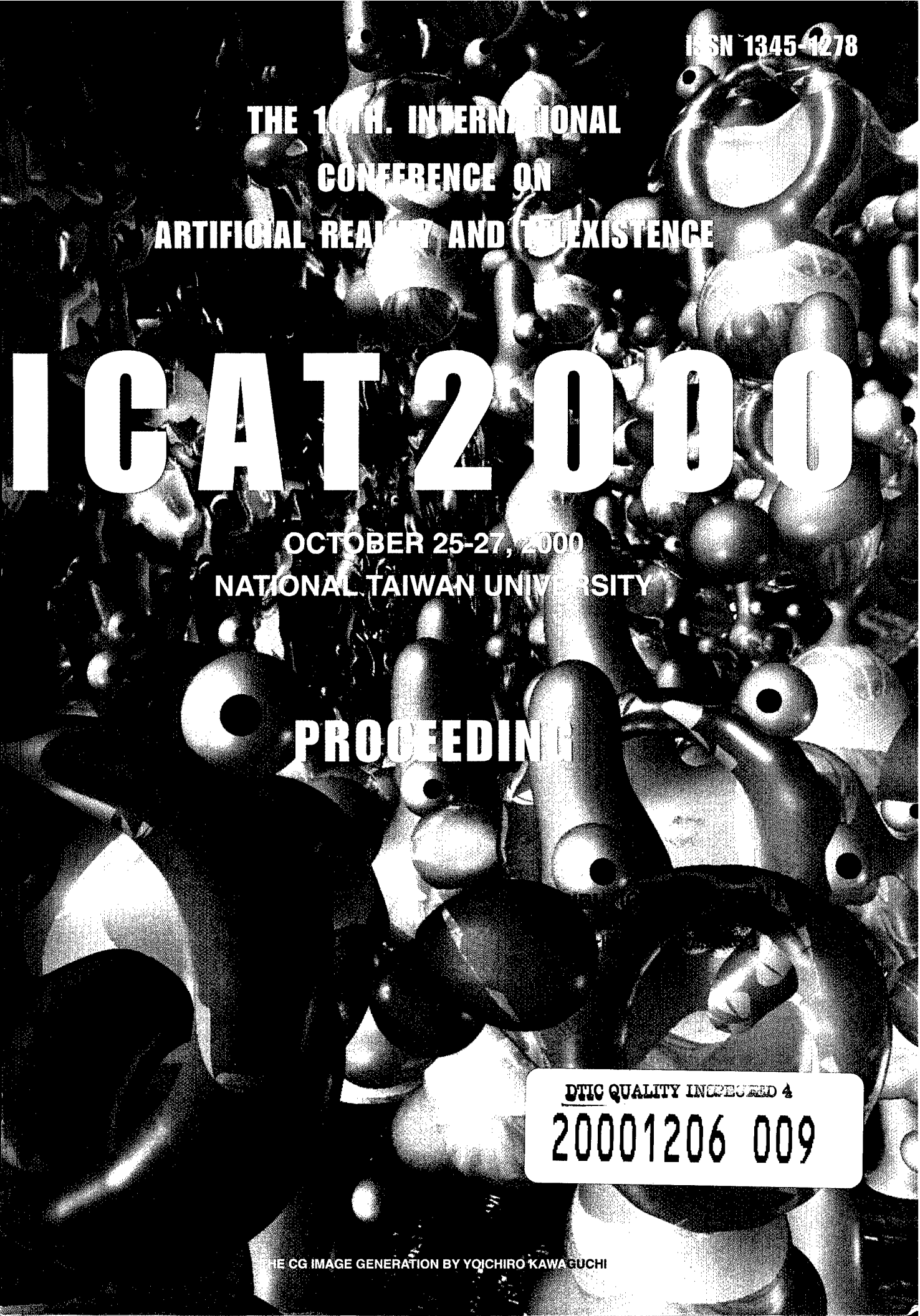


ISSN 1345-1278

THE 10TH. INTERNATIONAL CONFERENCE ON ARTIFICIAL REALITY AND TELEXISTENCE

ICAT 2000

OCTOBER 25-27, 2000

NATIONAL TAIWAN UNIVERSITY

PROCEEDINGS

THE CG IMAGE GENERATION BY YOICHIRO KAWAGUCHI

Corrections of IC-AT 2000 Proceeding

We are sorry for these print mistakes in the IC-AT 2000 proceeding.

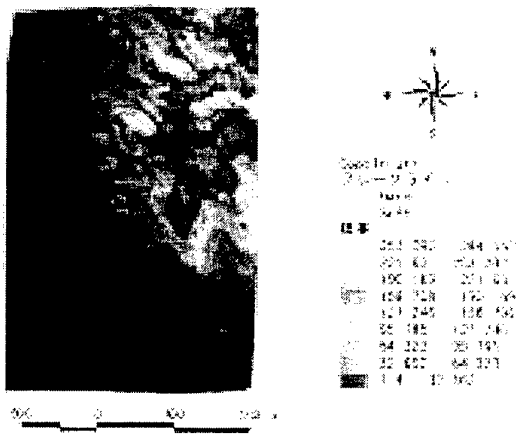


Fig. 5 TIN made by points and contour

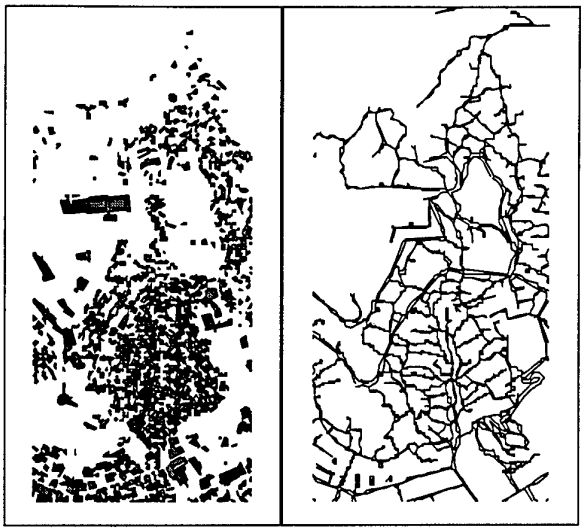


Fig. 7 Spatial data (building & road).

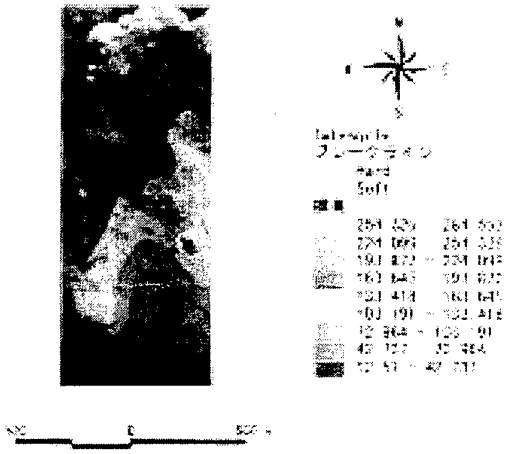


Fig. 6 Finished TIN in this study

Fig. 8 VRML image of Tateyama region

and coastline information as a polygon data. And, the following are input as a line data: Road and stair and railway and train route of spatial data. Figure.7 is spatial data that show building and road. The attribution data of the building input classification (wooden construction and non-wooden construction and concrete, etc.), nameplate and building application (it is classified into 22 types such as housing, store, public facility) with the rank. And, vacant land and plowed field, parks and planned road and plan parks, etc. are added information.

4.2.4 3D image (VRML)

The 3D-preparation image was made to be VRML format that could grasp the whole town. Figure.8 is 3D image that converted as a VRML format. By adding the feature, the image on IPT used it.

5. Simulation of Urban View in IPT

Observers can experience virtual world by Immersive Projection Technology (IPT), which is constituted of multi wide-screens and stereo system utilizing liquid-crystal-shutter or polarized plastic framed glasses. In this study to simulate urban views, we used IPT having front, both sides and floor screens. Graphics work station (Onyx2) stored VRML files and projected them using Performer library. Figure.9 shows the simulation of whole town. As original maps have cross-sections of buildings with height information only, all buildings seem like simple boxes. To make more reality, we will have to add roofs of Japanese houses and so on. Figure.10 shows different scale views. Observers can see the town from aerial view and enter the same scale town as real world. Building's windows and entrances were obtained by texture mapping using digital photograph taken at the places. The advantages of simulation GIS in IPT are that observers can see views from various angles and change scale size as if they are in the town. Furthermore, if we set a treadmill in the inside of IPT and the rotation speed of the belt synchronizes images, observers can experience

the content authors, service providers, and end users. Another significant feature of MPEG-4 is the ability of Synthetic Natural Hybrid Coding (SNHC). This not only enriches the content of MPEG-4 scenes, but also leads to more reasonable manipulation of limited bandwidth. To accomplish the above features, MPEG-4 must draw up a scene description language to describe the structure of the scene. The language takes VRML97[2] as the basis and adds some new nodes for other purposes. The rendering module composites and renders the scene according to the structural information and the media samples dealt by the visual codec. Furthermore, the rendering module has to implement several important mechanisms so that the MPEG-4 system can bring its ability into full play, such as navigation in the scene, changing the viewpoint, individually adjusting playing quality of video objects, and the animation mechanism.

1.1 System Overview

In essence, our system is an implementation of a VRML browser under the MPEG-4 architecture. The difference between other VRML browsers and ours is the VRML scene data acquired through the BIFS Decoder (Binary Format for Scene Stream Decoder). The video/audio data required by scenes are processed through a video/audio decoder in our system.

Our rendering module consists of the following tasks. Two of them are about composition and displaying the scene onto a screen, and others are about cooperation with other modules in the system:

1. To control the 2D/3D rendering engine.
2. To interpret the scene tree structure, compose the scene, and set up the geometry framework.
3. To support the node definition and the structural mechanism of scene description language.
4. To link up with the media codec, get the visual media sample, and manage buffers.
5. To interact with users, provide navigation ability, and feedback users' requests to the system.

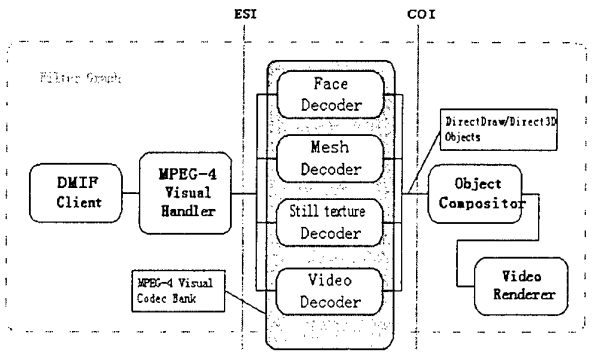


Figure 2: The figure above is the implementation of our rendering module which is the part to the right of the line of COI (Composition Interface).

Before the final MPEG-4 system integration currently, we have our own independent testing environment. In this testing system, MPEG-4 scenes are described in the VRML grammar, and then are interpreted by the parser. The decoder for still images/video can read the necessary texture data in advance for testing.

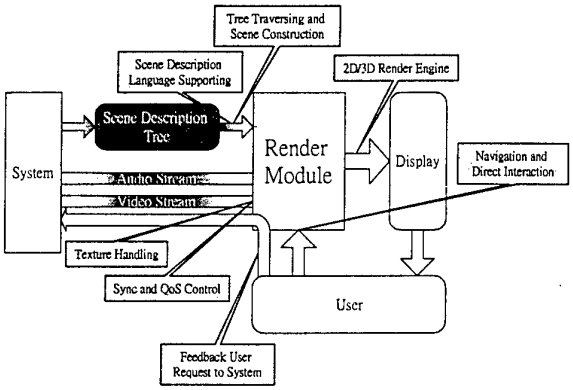


Figure 3: Illustration of the I/O flow of the whole rendering module.

In the implementation of MPEG-4, we use the Microsoft multimedia architecture, Directshow. From software points of view, the kernel of DirectShow is a modularized pluggable system, based on the usage of the so-called filters. The most significant advantage of DirectShow comes from its ability to make the multimedia application design more clear and easy. By carefully dividing the work into connected filters in the DirectShow architecture, each filter can be implemented by different program developers. Another advantage of DirectShow is the filter re-use, which speedups the developing of new multimedia applications. So our program of rendering can be independent from other parts in the system, and is wrapped to be a filter according to the DirectShow architecture.

2. Implementation

The rendering module is developed on the Microsoft Windows 98/2000 platform. OpenGL and DirectX are used to implement rendering. In order for the convenience of cross-platform compatibility, we wrapped our program in a new interface for the use of OpenGL and DirectX. In actual implementation, when there are more video textures in the scene, we can have greater performance by adopting DirectX for rendering. Because via DirectX interface, most display cards provide 2D image hardware acceleration which helps create video textures in real time. Currently there are no special functions designed for 2D image processing in OpenGL, so 2D image processing is processed purely by software. If there are not many video textures, the performance in

REPORT DOCUMENTATION PAGE

Form Approved

OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
30-11-2000	Conference Proceedings	25-27 Oct 00

4. TITLE AND SUBTITLE

10th International Conference on Artificial Reality and Tele-Existence, 25-27 Oct 00, National Taiwan University

5a. CONTRACT NUMBER

F6256200M9189

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Conference Committee

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

National Taiwan University
1, Sec. 4, Roosevelt
Taipei 106
Taiwan

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AOARD
UNIT 45002
APO AP 96337-5002

10. SPONSOR/MONITOR'S ACRONYM(S)

AOARD

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

CSP-00-28

12. DISTRIBUTUION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13 SUPPLEMENTARY NOTES

14. ABSTRACT

Conference Proceedings Includes the Following Sessions:
Session 1: Interaction Technology
Session 2: Computer Graphics/Rendering
Session 3: Medical Application & Artificial Life
Session 4: Networked Virtual Reality
Session 5: Virtual Reality and Mixed Reality
Session 6: Device Development
Session 7: Interactive Art
3 Invited Speeches, Poster Session, and Banquet/Dinner Talk

15. SUBJECT TERMS

augmented reality, Display Technologies, Virtual Reality

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE		279	Terence J. Lyons, M.D.
U	U	U	UU		19b. TELEPHONE NUMBER *(Include area code)* +81-3-5410-4409

DEPARTMENT OF THE AIR FORCE
ASIAN OFFICE OF AEROSPACE RESEARCH AND DEVELOPMENT
AIR FORCE RESEARCH LABORATORY/OFFICE OF SCIENTIFIC RESEARCH
AOARD UNIT 45002, APO AP 96337-5002

30 Nov 00

MEMORANDUM FOR Defense Technical Information Center (DTIC)
8725 John J. Kingman Road, Suite 0944
Fort Belvoir VA 22060-6218

FROM: AOARD
Unit 45002
APO AP 96337-5002

SUBJECT: Submission of Document

1. Conference Proceedings from the "The 10th International Conference on Artificial Reality and Telexistence", held 25-27 Oct 00, at the National Taiwan University, Taipei, Taiwan, is attached as a DTIC submittal.

2. Please contact our Administrative Officer, Dr. Jacque Hawkins, AOARD, DSN: 315 229-3388, DSN FAX: 315-229-3133; Commercial phone/FAX: 81-3-5410-4409/4407; e-mail: hawkinsj@aoard.af.mil, if you need additional information.

MARK L. NOWACK, Ph.D.
Acting Director, AOARD

Attachments:
1. AF Form 298/Documentation Page (CSP-00-28)
2. DTIC Form 50/DTIC Accession Notice
3. Conference Proceedings of "The 10th International Conference on Artificial Reality and Telexistence"

10th International Conference on Artificial Reality and Telexistence

October 25-27, 2000

National Taiwan University

Sponsored by:
Instiute of Information and Compaling Machiery
The Virtual Reality Society of Japan (VRSJ),
National Science Council
Academia Sinica.Taiwan
U.S. AFOSR/AOARD
Industry Donations
Chung Yuan Christian University

ICAT2000 PROCEEDING

CONTACT

Department of Computer Science and Information Engineering,National Taiwan University
c/o Prof. Ming Ouhyoung,
#1,Roosevelt Road, section 4, Taipei, Taiwan

TEL: +886-2-2362-5336 # 203
FAX: +886-2-2365-8741
e-mail address:ming@csie.ntu.edu.tw

Message from the General Co-chairs

On behalf of the Organizing Committee of the Tenth International Conference on Artificial Reality and Telexistence (ICAT2000), it is our great pleasure to welcome all of you to National Taiwan University, Taipei, Taiwan. This is the first time this conference will be held in Taiwan. We wish all of the attendants enjoy this conference and have a nice time in Taipei.

Applications of virtual reality and telexistence are now being sought world wide, and these technologies are expected to be most promising generic technologies in the 21st century. It is therefore quite timely and significant to have this series of international forum annually for the exchange of new concepts, ideas and experimental results, and to discuss deeply among experts who represent various fields that are regarded as entirely different at least twenty years ago and are aiming at the same goal of virtual reality and telexistence today.

The new features in this conference include a special session in interactive art and virtual art, as well as audio and haptic art, organized by Professor Peisuei Lee. Accordingly, there will also be live performance during the conference.

We really hope that you will find all aspects of virtual reality and telexistence in this conference, and also enjoy the demonstration of the new apparatus and/or products. We would like to thank all the members of the organizing committee for their efforts in ICAT2000. Enjoy your ICAT.

October 24, 2000

Ming Ouhyoung, Ph.D.
General Co-Chair

Susumu Tachi, Ph.D.
General Co-Chair

Message from the Program Co-Chairs

On behalf of the Program Committee of ICAT'2000, we would like to express our thanks to all the contributors, whose high-quality works and presentations are essential to the success of this conference. The technical program of this conference consists of four invited speeches (including one dinner talk), one poster session and six oral sessions. Furthermore, we are grateful to have Professor Peisuei Lee, Professor Masahiro Miwa and Professor Kumiko Kushiyama help us organize a few special sessions on interactive art, virtual art, audio art, haptic art, and musical performance. We believe this conference will be more colorful and enjoyable with these vivid activities.

The review process of the technical submissions started from mid-June and completed at the beginning of August. Each of the submitted papers we received was reviewed by at least two members of the program committee and additional reviewers. Based on the reviews, 32 technical papers were selected and published in this volume of Proceedings. The reviewers of the technical submissions have done an excellent job within a very tight schedule, and we cordially appreciate their time and effort.

Finally, we especially thank Professor Shi-Nine Yang and Professor Jung-Hong Chuang for their invaluable help and advice in paper review and in organizing the technical program. Also, we would like to take this opportunity to thank Julia Huang and Hidenori Maruta who serve as the committee secretaries. Without their professional assistance, we could not have accomplished the task of the Program Committee on schedule.

Welcome to ICAT'2000 in Taipei! We hope that all the participants will enjoy the conference and have fruitful time.

October 24, 2000

Makoto Sato
Tokyo Institute of Technology

Yi-Ping Hung
Academia Sinica

ORGANIZING COMMITTE MEMBERS

General Co-Chairs

Susumu Tachi(University of Tokyo)
Ming Ouhyoung (Taiwan University)

Program Co-Chairs

Makoto Sato(Tokyo Institute of Technology)
Yi-Ping Hung(Academia Sinica, Taiwan)

Organizing Committee

Steve Bryson (NASA)
Jung-Hong Chuang (National Chiao Tung University)
Stephen R. Ellis (NASA)
Toshio Fukuda (Nagoya University)
Yukio Fukui (AIST, MITI)
Martin Goebel (FIG)
Mark Green (University of Alberta)
Hiroshi Harashima (University of Tokyo)
Michitaka Hirose (University of Tokyo)
Kouichi Hirota (University of Tokyo)
Larry Hodges (Georgia Institute of Technology)
Takayuki Itoh (NHK)
Hiroo Iwata (Tsukuba University)
Robert Jacobson (World Design)
Fumio Kishino (Osaka University)
Myron Krueger (Artificial Reality)
Jurgen Landauer (GFT Informationssysteme)
Carl Loeffler (CMU)
Taro Maeda (University of Tokyo)
Ryo Mochizuki (NHK Engineering)
Marc Raibert (BDI)
Warren Robinett (Virtual Reality Games)
Makoto Sato (Tokyo Institute of Technology)
S. K. Semwal (University of Colorado)
Thomas Sheridan (MIT)
Li-Sheng Shen (CCL/ITRI, Taiwan)
Robert Stone (Virtual Presence Ltd.)
Susumu Tachi (University of Tokyo)
Haruo Takemura (NAIST)
Daniel Thalmann (Swiss Federal Institute of Technology)
Shi-Nine Yang (National Tsing-Hwa University)

International Program Committee

Fumihito Arai (Nagoya University)
Stephen R. Ellis (NASA)
Toshio Fukuda (Nagoya University)
Yukio Fukui (AIST, MITI)
Mark Green (University of Alberta)
Hiroshi Harashima (University of Tokyo)
Michitaka Hirose (University of Tokyo)
Kouichi Hirota (Toyohasi Institute of Technology)
Larry Hodges (Georgia Institute of Technology)
Tohru Ifukube (Hokkaido University)
Yasushi Ikei (Tokyo Metropolitan Institute of Technology)
Takayuki Itoh (NHK)
Hiroo Iwata (Tsukuba University)
Youichirou Kawaguchi (Tokyo University)
Ryugo Kijima (Gifu University)
Fumio Kishino (Osaka University)
Yoshifumi Kitamura (Osaka University)
Kiyoshi Kiyokawa (CRL)
Myron Krueger (Artificial Reality)
Kun-Pyo Lee (KAIST.)
Pei-Suei Lee (Chung Yuan Christian University)
Tong Yee Lee (National Cheng Kong University)
Carl Loeffler(CMU)
Taro Maeda (University of Tokyo)
Fumihito Arai (Nagoya University)
Stephen R. Ellis (NASA)
Toshio Fukuda (Nagoya University)
Yukio Fukui (AIST, MITI)
Mark Green (University of Alberta)
Hiroshi Harashima (University of Tokyo)
Michitaka Hirose (University of Tokyo)
Kouichi Hirota (Toyohasi Institute of Technology)
Larry Hodges (Georgia Institute of Technology)
Tohru Ifukube (Hokkaido University)
Yasushi Ikei (Tokyo Metropolitan Institute of Technology)
Takayuki Itoh (NHK)
Hiroo Iwata (Tsukuba University)
Youichirou Kawaguchi (Tokyo University)
Ryugo Kijima (Gifu University)
Fumio Kishino (Osaka University)
Yoshifumi Kitamura (Osaka University)
Kiyoshi Kiyokawa (CRL)
Myron Krueger (Artificial Reality)
Kun-Pyo Lee (KAIST)
Pei-Suei Lee (Chung Yuan Christian University)
Tong Yee Lee (National Cheng Kong University)

ORGANIZING COMMITTE MEMBERS

International Program Committee

Carl Loeffler(CMU)
Taro Maeda (University of Tokyo)
Tadao Maekawa (ATR)
Tsutomu Miyasato (ATR)
Ryo Mochizuki (NHK Engineering)
Ryohei Nakatsu (ATR)
Emi Nishina (National Institute of Multimedia Education)
Haruo Noma (ATR)
Jyunji Nomura (Matushita Electric Works, Ltd.)
Tetsurou Ogi (University of Tokyo)
Yuichi Ohta (Tsukuba University)
Jyun Ohya (ATR)
Takeo Ojika (Gifu University)
Ming Ouhyoung (National Taiwan University)
Jong-Il Park (Han Yang University)
Makoto Sato (Tokyo Institute of Technology)
Kosuke Sato (Osaka University)
S.K. Semwal (University of Colorado)
Zen-Chung Shih (National Chiao Tung University)
Harry Shum (Microsoft Research China)
Robert Stone (Virtual Presence Ltd.)
Susumu Tachi (University of Tokyo)
Takashi Takeda (Nagasaki Institute of Applied Science)
Haruo Takemura (NAIST)
Daniel Thalmann (Swiss Federal Institute of Technology)
Hiroyuki Yamamoto (MR System Lab. Inc.)
Juri Yamashita (AIST, MITI)
Yasuyuki Yanagida (University of Tokyo)
Hiroaki Yano (Tsukuba University)
Shigeki Yokoi (Nagoya University)
Yasuyoshi Yokokouji (Kyoto University)
Naokazu Yokoya (NAIST)
David Zeltzer (Sernoff Corporation)
Zhengyou Zhang (Microsoft Research)
Michael Zyda (NPS)

Local Program Committee

Lih-Shyang Chen (Ministry of Education)
Hang-Ming. Chen (National Taiwan University)
Cheng-Chin Chiang (CCL/ITRI)
Zen Chen (National Chiao Tung University)
Chu-Song Chen (Academic Sinica, Taiwan)
Y.T. Ching (National Chiao Tung University)
B.S. Chung (Chung Yuan University)

Local Program Committee

Chin-Shyurng Fahn (National Taiwan University of Technology)
Jiung-Yao Huang (Tamkang University)
Yuan Kang (Chung Yuan Christian University)
Hao-Ren Ke (National Chiao Tung University)
Tong Yee Lee (National Cheng Hong University)
Chung-Nan Lee (National Sun Yat-sen University)
C. M. Wang (National Chung Hsing University)

Local Arrangement Co-Chairs

Pei-Suei Lee(International Academy of Media Arts and Sciences)
Ming Ouhyoung (National Taiwan University)

Exhibition Chair

Pei-Suei Lee (International Academy of Media Arts and Sciences)
Diing-Wuu Vale,Wu(Chung Yuan Christian University)

Local Arrangement Secretary

Julia Huang (National Taiwan University)

CONTENTS

Invited Speech 1

Session 1 : Interaction Technology

Session 2 : Computer Graphics/Rendering

Session 3 : Medical Application & Artificial Life

Invited Speech 2

Session 4 : Networked Virtual Reality

CONTENTS

Session 4 : Networked Virtual Reality

Poster Session

Poster Session

Session 5 : Virtual Reality and Mixed Reality

Banquet/Dinner Talk

Invited Speech 3

CONTENTS

Session 6 : Device Development

Session 7 : Interactive Art

Session 7 : Interactive Art

Haptic Art

Bringing the Real to Virtual Reality

Anselmo Lastra
University of North Carolina, USA
lastra@cs.unc.edu

Abstract

Photographs, movies, and video have been used not only to tell stories, but also to transport a viewer to distant, and sometimes to exotic places. These media enable the viewer to passively observe realistic images of the world, but have not allowed free exploration. The director has determined what you'll see next. In contrast, interactive computer graphics allows free exploration of spaces, but usually at the cost of a sense of realism. Recent advances in computer vision and computer graphics have made it feasible to achieve both *exploration* and *realism*, to allow users to freely explore spaces that look as real as video.

In this talk, I will survey the technologies necessary to achieve this interactivity and realism: scene capture, model representation, processing, and image generation. I will briefly present the results obtained by our team in image-based modeling and rendering. Then I will discuss what I see as the research opportunities in this emerging sub-field of graphics.

The Evolution of Virtual Humans in NVE Systems

Nadia Magnenat-Thalmann, Chris Joslin
MIRALab – University of Geneva
24 rue du General-Dufour, CH1211, Geneva-4, Switzerland
{thalmann, joslin}@miralab.unige.ch

Abstract

In this paper we present the evolution of Virtual Humans in Networked Virtual Environment Systems (NVE) where advanced interaction is involved. Starting with the most basic pioneering NVE Systems, we demonstrate the evolution through architecture, and operating systems and basic improvements made to the system, problems and limitations. We present five main examples, the basic original systems, two advanced scenarios involving multiple Virtual Humans, the State of the Art in NVE Systems and our latest development under the Windows Operating System (OS). We conclude with our case study demonstrating the teaching of Virtual Dance over the Internet.

Key words: Networked, Networked Virtual Environments, Internet, Virtual Humans, Advanced Interactivity, Dance

1. Landmark Systems

In this paper we shall present, not only our main system (Virtual Life Network or VLNET), but also many of the systems that gave precedent to the available Networked Virtual Environment Systems currently in use today.

The original, so called, NVE Systems arrived in the early 1990's [1, 2]. These systems were basically text-based environments (or Multi-User Dungeons - MUDS) connected together via a network. The first general NVE Systems with actual graphics were introduced about a year later. One of the first systems, called dVS [4], was developed as a commercial system to enable the visualization/manipulation of CAD data, however this visualization is extended across a network and therefore large collaboration can exist between multiple viewers. However, this original system lacked virtual humans and therefore focused on the interaction with the CAD data and not interaction between virtual humans. DIVE [5] was presented in 1993 and developed more in the direction of an actual NVE System rather than a Networked CAD viewing application. NPSNET [6,7,8] represents the first NVE System to incorporate virtual humans into the environment. The NPSNET system was developed for the purpose of military simulation and combat training purposes and therefore the requirements of the scenario required actual virtual humans, although simplified in terms of anatomical accuracy, this system represents really the first step towards virtual human interaction. 1995 saw the explosion of many NVE systems being introduced to both the research and commercial world. VISTEL [9], MASSIVE [10] and BrickNet [11] were all introduced at the same time as our own VLNET [12,13] NVE System, each emphasizing on different aspects. VLNET for instance was able to represent much more realistic virtual humans, whereas VISTEL used simplified virtual humans and BrickNet and MASSIVE had no human representation at all. VISTEL an abbreviation of Virtual Space Teleconferencing, and although limited two virtual environments linked together by a network, presented other solutions such as tracking of facial features and enabled talking. BrickNet introduced the use of object sharing to enable the user more interactivity within the environment. Blaxxun [14] was introduced in 1995, although still in its infancy it introduced the use of Virtual Environments using web browsers as access portals to the server, this enabled a more general access to NVE Systems. SPLINE [15] was introduced in 1997 and used a broader range of capabilities to enable better depth perception in the virtual environment (3D Sound, and multiple device input).

Trends have also moved from what was once a totally UNIX dominated area towards the PC domain, although not for all systems. This has meant several changes and certainly has changed the way in which the systems are created. Improvements in speed also have meant that the rendering quality has been improved along with the speed and real time aspect of the system. Also the introduction of standards has helped improve the range of models that can be used and not limited only to a specific lab. In this paper we examine the evolution of NVE Systems from the point where Virtual Humans were first introduced. We shall present the limitations of such systems and step through the advancing stages until present day. We then present our latest system, W-VLNET, which suggests the latest in Advanced Networked Virtual Environment Systems and the Virtual Humans residing within them.

2. Virtual Life Network – VLNET

2.1 The Precedent

In 1995 we presented our first NVE System, called Virtual Life Network (or VLNET) [12], this was completely based on the UNIX OS, the only operating system, at the time, capable of running such software. It was based on a broadcast type network topology and presented one of the first uses of Virtual Humans in the NVE Systems. The system was quite basic, but actions such as walking [16] and grasping were possible, which allowed the real user simple interactions with the world.

Navigation was performed using either direct mouse or Space-Ball interaction. Facial actions were done using a texture mapped streamed video, allowing each participant to see the expressions on the others faces. Figure 2a shows a screen shot from the first interaction, in a virtual environment, using VLNET.

As can be seen, this system is quite basic, it provides one of the first glimpses of a NVE with advanced virtual human interaction, but there are many facets of this system that needed to be improved. The body motor functions [17], although basic by today's standards, were quite advanced. Improvements in the overall system were constantly in progress, this was in terms of the network module, the functionality and the basic quality of the system.

Figure 2a – VLNET first test for NVE System

2.2 The Improved System

In 1996 [18] continued development provided a new improved version of the VLNET System. Motion engines creating more realistic movements were introduced and additional drivers were incorporated to enable greater interaction with the environment. The network has been improved to include Client/Server network architecture. One major improvement that was introduced was the use of a face that could be animated. Providing an enormous leap forward in facial communication for low bandwidth. Real-Time Tracking [19] was also introduced and enabled greater interaction with the environment than the previous mouse/space-ball systems. Virtual Humans using Metaballs [20,21] were also introduced into the system, enabling more realistic representation of muscle movements. Figure 2b shows a screen shot from the more advanced system.

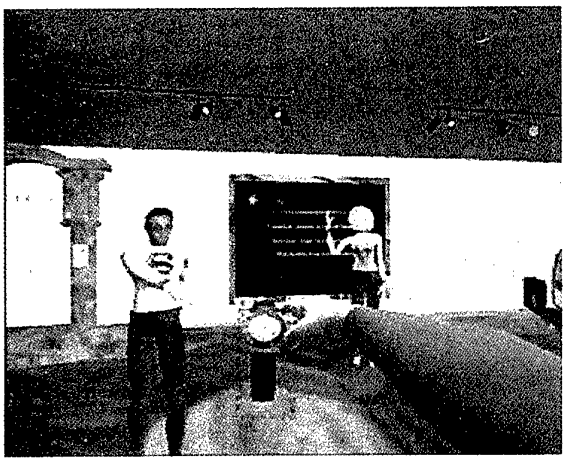


Figure 2b – VLNET Interactive Systems in 1996

2.3 Cyber Tennis

Anyone for Tennis [22], shown in 1997 at Telecom Interactive in Geneva, is a classic example of a fully interactive NVE System. Based on previous work on collaborative games [23], the Tennis Game scenario itself was quite simple, but the realization was much more difficult. The improved version of the VLNET System [24,25] was used and both players were linked to a Motion Tracking System and placed at different locations in Switzerland (one in EPFL, Lausanne and one in Telecom Interactive, Geneva). An ATM Network Connection was used to transmit both the data between the NVE System and the live video showing the demonstration at the EPFL end. Each player was able to visualize the scene using a Head Mounted Display (HMD) unit, with Stereo ability; this video was also transmitted to a screen on the stage. The audience could therefore see what both players were seeing, the real stage at both ends and the Virtual Tennis court; several of those views can be seen in Figure 2c. The players were to play a game of tennis, however the court, the judge and the racket and ball existed only in the virtual environment, only the players themselves were real. Both players were tracked and their virtual counterparts moved mimicking their movements, therefore if they swung their arm the virtual racket would move also.

The entire scenario lasted for one game, the rules of the game were observed and an autonomous judge (Marilyn Monroe) was used to determine faults. She also used a limited vocabulary to announce the score.

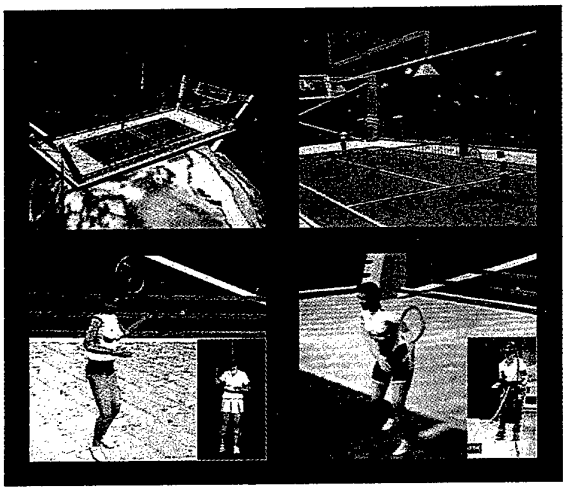


Figure 2c – Cyber Tennis, A Tennis Match over ATM Networks

The realization of the scenario was extremely complex. The scene was limited mainly by computing power and some very powerful UNIX machines were used so that the scene could be rendered in real-time. The Tracking System also posed problems due to the physical limits, in terms of cable length and accuracy for tracking distance. Hence a dynamic navigation adjust was implemented so that each Player had a restricted movement zone and when leaving this zone the global position was altered by an factor to enable the Player to freely move around the court. The HMD displays were also limited in resolution (247x230) to preserve good rendering speeds; hence both the ball and the rackets were enlarged to enable the players to see and hit the ball. The ball also had certain physical limits imposed on it so that it would act like a tennis ball, but the equation used to calculate its trajectory did not use too much CPU time.

3. Cyber Dance

Cyber Dance was a performance, shown many times, involving the interaction between many real and virtual humans, performed as a combination of real-time and autonomous virtual humans. VLNET again was used as the Virtual Environment System. The performance was based on a dance sequence where Virtual Humans interacted with the real humans on stage. Obviously due to the complexity of having multiple Virtual and Real Humans it was possible to track all the real dancers on the stage. The actual scenario involved a choreographed dance sequence for the real dancers, shown in the bottom right of the Figure 3a. The virtual dancers followed a pre-recorded dance sequence (also choreographed), which can be seen in the top right and the bottom left of the screen. One of the dancers was tracked and the top left section of the screen his virtual counterpart can be seen mimicking his movements.

The number of Virtual Humans in the scene, and the complexity of the sequence, makes this quite a complex demonstration of Virtual Environments. The Virtual Humans themselves were completely deformable, which enables the use of reflective body surfaces (as can be seen in the top left picture). The introduction of shadowing into a real-time system also enabled a more realistic feel to the entire experience.

Figure 3a – Cyber Dance performance, a mix of real and virtual dancers

This performance improved the aesthetics of the basic scene, enabling multiple virtual humans to perform at once, additional rendering of shadows and an intricate choreography that was performed not only by the real dancers, but also the virtual ones. The technical achievements lie mainly within the scene management itself.

4. State of the Art

4.1 Introduction

Currently, in the world of NVE Systems, there exist many up-to-date systems. Some of them are developments from the original systems discussed in Section 1, some are spin-offs of the original systems and the rest are new. In this Section we discuss the current systems available, their capabilities and limitations and where they are most effective.

4.2 Blaxxun

Blaxxun is one of the most commercially successful of all the current systems in existence today. It is based on the classic Client/Server technology and incorporates its Client software into a plugin for any of the current web browsers. The Server is based around the same principle and resides on a standard HTTP Server. This makes it extremely accessible to the public who are already familiar with the use of web browsers to do many different tasks. The implementation uses some standards, such as VRML97 [26] (for Scenes) to build a populated world, but their Virtual Humans are of a proprietary format, but they provide a complete studio (Avatar Studio) to enable Avatars to be customized and clothed. The Virtual Humans also only use an articulated rigid segment body to maintain high

rendering speeds. The system currently is limited to extremely low bandwidth connections, text chat is possible and they provide a list of predefined gestures (a combination of both body and face movements). However, audio communication is currently not available and advanced interaction, using Motion Tracking for instance, is currently unobtainable. This system is mainly based on the Windows OS.

4.3 Division Reality (dVS)

The dVS system has evolved into Division Reality [4]; this is more of a complete professional system. However it is still geared towards the CAD and Visualization end of the market. The system itself is also only available on high-end UNIX machines (from HP and SGI) and therefore maintains its dominance in the professional market. The system however is able to connect to a variety of 3D interactive devices, including simple Stereo Shutter Glasses [27], Motion Tracking Systems [28], Head Mounted Displays [29] and Large Screen Displays [30]. It supplies a library of Virtual Humans of its own proprietary format for use within the Virtual Environment. It incorporates motion planning to animate the virtual environment and high quality rendering to improve the visualization aspect.

4.4 SPLINE and Open Community

The SPLINE System was a project from the Mitsubishi Electric Research Lab (MERL) and was finished in 1997; the project was completed and not continued. However, this work has provided a basis for new work started in 1997 called Open Community [31]. Open Community is a proposed open standard middleware and API platform for multi-user virtual worlds. It consists of extensions to Java and VRML 2.0. Being in Java it permits the use on multiple platforms (although being in Java suffers performance problems in comparison to other systems). It provides all the basic requirements for an NVE System (3D Graphics, audio, system management etc). However, the project still needs to obtain the worldwide acceptance of that Blaxxun has achieved. One of the problems that were inherited from the original SPLINE system is that Virtual Humans are still quite simplified.

4.5 MASSIVE/HIVEK

The original MASSIVE System Project was finished in 1997 and two further versions have been developed since: MASSIVE-2 and MASSIVE-3 [32]. The MASSIVE-3 System is a completely multi-platform solution, running on SGI IRIX, IBM AIX and the Windows OS. The work still does not support the use of virtual human representations and the main focus of the system is on the scalability of the system and the networking aspect.

4.6 NPSNET-V

The NPSNET [6] project is currently on version 5 and currently under continued development. Improvements towards the network, scalability and object behavior are emphasized in the latest work. As with many military simulation systems, improvements made towards the realism of the situation are the most important in this context, rather than actual virtual human communication. Therefore object behavior and animation, along with virtual human representation are the most important aspects.

4.7 Others

World2World from Sense8 [33] is a Networked Visualization System, similar to the Division Reality System. The emphasis again is in the collaboration for the design of virtual equipment and not on virtual chat or interactivity in the entertainment sense. Therefore, the system does not use Virtual Humans. It runs on multiple OS and has links with popular interactive devices (such as cyber gloves, space-balls, stereo displays and motion tracking systems).

The DIVE System, although pioneering in its time has cease to be continued in any way.

5. W-VLNET

5.1 Introduction

To conclude the State of the Art in NVE systems we present an overview of our current system W-VLNET [34]. This is a system based upon the original VLNET system, as described in Section 2. However, there are many significant differences between this system and the previous system, both in underlying architecture and usability. The main difference is that the whole system was designed and executed under the Windows OS. This has made many of the fundamental architecture points unusable and therefore the use of Shared Memory and Processes has been discarded in favor of a fast message passing architecture and threads. Obviously the basic device drivers and architecture links have also been changed.

The Scene Management is basically using animation libraries placed on top the basic Scene Graphic (supplied by OpenGL Optimizer [35]). These animation libraries handle the incoming translation and animation data (both from the Network and the attached devices). It also provides additional control for collision detection/response and adding gravitational animation to objects.

5.2 Networking Improvements

The Network, even though still based on the Client-Server approach has been improved dramatically. With the introduction of standards, as described in Section 5.3, the network bandwidth usage has improved also (due to the compression technology used). The connection of multiple data channels has been included to control the flow of data between the Client and the Server. Various filtering mechanisms have been incorporated to reduce network traffic; these filters

determine whether traffic is necessary by determining the distance between the destination and the source. This can be done on a per Client basis and therefore gives the previously unloaded Server some processing and control ability. Four channels now exist to pass translation, animation, audio, video and file data between Client and Server.

A caching mechanism has also been introduced to further reduce network traffic, both at the Client side and the Server side.

5.3 Advanced Capabilities

In addition to the basic architecture that provides the Scene Managing capabilities and Networking there are also other modules in the system for advanced interaction with the environment:

- Human Motion Tracking - The Motion Tracking System [19,28] has been included and uses the latest standards, as described in Section 5.4, to animate the representative avatars in the Virtual Environment. All limb motions are tracked (including fingers) which allows the attached human complete interaction with the virtual environment. Of course the data captured from this Tracking System is also transmitted across the network to all other connected Clients.

- Audio Communication – Audio allows a large communication medium to be added, giving the entire experience in the Virtual Environment much more depth. Voice-to-Voice communication is possible using an Audio link, audio data is captured using a normal microphone, compressed with a standard compression codec (see Section 5.4), sent to the Server which distributes the data to all other Clients and decompressed at the other end and passed out using a Speaker system. The system not only permits voice communication between Clients, but also acoustical objects can be added to the scene to provide such things are radios etc.

- Speech/Text Communication [36] – Both text to display as text and text to convert into Speech can be sent across the network. The Speech part can convert the text stream into both the acoustical part and the lip animation. This enables the most natural aural communication.

5.4 Use of Standards

One of the major problems faced when developing any kind of VE System, the use of proprietary formats (for input and output) means additional development time for formats that have already been invented. Therefore the use of standards throughout the system means that there is more time to spend on the underdeveloped areas of NVE Systems. The basic formats used for both Virtual Humans (MPEG4 [38] and HANIM [39]), Objects (VRML97 [26]) and Audio (G723.1, G.728 and G.711 [39]) provide links to other packages and also means that we can use the same formats not only throughout the lab, but also with the rest of the world.

The use of the MPEG4 standard was one of the most important steps forward as this is a relatively new standard and is used not only for the compression of video and audio but also for virtual scenes and also virtual humans. HANIM is a standard for the description of a Virtual Human using the VRML97 Proto format. This format is being adopted by many commercial and research institutes alike. Both formats support not only the description of Virtual Humans, but also the animation of them as well.

5.5 Case Study – Virtual Dance

To really present and also to make a thorough test of this system we present our Virtual Dance case study. This case study involves a demonstration that linked to dance participants together in a Virtual World. The two participants are separated geographically and connected via a normal Internet network connection. As with the previous system we use representative virtual humans for the two dancers and connect them to the NVE System using the Flock Of Birds Motion Tracking System as briefly described in Section 5.3. The basic premise of this case study is that there is a Dance Teacher and Student, the Teacher must teach the Student to dance using the NVE System. Obviously certain limitations are already in effect:

- Each participant can only see the others Virtual Human Representative.

- The Flock of Birds Tracking System has a limited range for the dance area and also slightly restricts the movement of the limbs.

- The Rendering System has a limited number of polygons that it can render per frame.

Figure 5a and 5b show examples (both real and virtual) of the Virtual Dance at both sites at the same point in time.

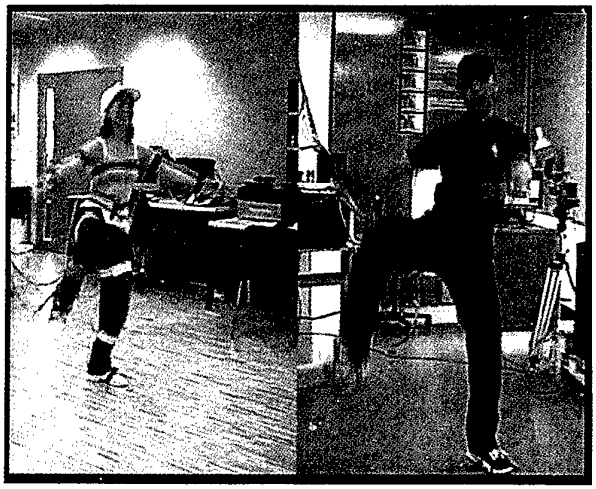


Figure 5a – Student (Left) and Teacher (Right)

Figure 5b – Student (Left) and Teacher (Right)

The session was approximately six minutes in length and the entire sequence was accompanied by dance music to enable the Teacher and Student to synchronize their movements (due to its continuous beat). The dance itself features the Teacher showing the Student how to Dance and then the Student trying to follow the movements, the Teacher giving comments on the Students performance. A Video Conferencing link was used to study delays and check for tracking comparisons between real and virtual, therefore being for analysis purposes only.

6. Conclusions and Future Work

In this paper we have presented the progress through the past five years of Virtual Humans in Networked Virtual Environment Systems. We have shown how Virtual Humans have started with simple basic tasks and progressed towards playing tennis and teaching dance. We have also demonstrated the directions in which both the Virtual Humans and the Virtual Environment have evolved and the current State of the Art both for our own system, W-VLNET and other NVE Systems. A case study has also been included to show that our current development platform is not just confined to this research institute, but has been demonstrated under normal conditions across an Internet linked network. As can be seen from these examples given, not only have Virtual Humans improved in their appearance and abilities, but also the Virtual Environment that they reside in has also improved. These improvements include both the rendering environment and the networking technology used to connect these Virtual Environments together.

As with all NVE Systems we continue to improve the balance between realism and the real-time aspect. We intend to increase the interactivity between the Virtual Human and the Virtual World even more and add other effects, such as 3D Audio and Environmental effects (such as clouds) to enable the users to feel much more immersed in their Virtual Environment. We are also aiming to integrate better parsers to enable more standard formats to be supported. The Client/Server architecture will also be upgraded to reduce the latency and improve usability. An improved graphical user interface is also being developed to enable users greater control in the environment.

7. Acknowledgements

The work done by MIRALab in this paper (Section 2,3 and 5) has been funded by many projects (VLNET, COVEN and VPARK), but collectively by the European Union. We would also like to thank Mireille Clavien and Marlene Arevalo for their work on designing the decors for the VPARK demonstration and to Gabby Rieder, Mahmood Davari, Luc Emering, Tom Molet, Nabil Sidi-Yacoub and Maja Jovovic for their participation during the Case Study demonstration.

8. References

1. Curtis, E. Pavel, "Mudding: Social P /Mid di ng: S text-based virtual realities", Proceedings of Directions and Implications of Advanced Computing, Xerox PARC 1992
2. Evard, Remy, "Collaborativ % Coll aborat Communication: MUDS as systems tools", Proceedings of Seventh Systems Administration Conference (LISA VII), *USENIX Association,* 1993
3. Towell J., Towell E., "Presence in Text-Based Networked Virtual Environments or MUDS", *Presence* 6(5), MIT Press, pp. 590-595, 1997
4. Division Solutions, http://www.division.com
5. Carlsson C., Hagsand O., "DIVE - a Multi-User Virtual Reality System", *Proceedings of IEEE VRAIS '93,* Seattle, Washington, 1993

6. Zyda M., Pratt D., Falby J., Barham P., Kelleher K., "NPS 0NPSNET and the Naval Post graduate Sc Graphics and Video Laboratory", Presence: Teleoperators and Virtual Environments, MIT Press, Vol. 2, No.3, pp. 244-258

7. Macedonia M., Zyda M., Pratt D., Barham P., Zestwitz, *"NPSNET: A Network Software Architecture for Large-Scale Virtual Environments"*, Presence: Teleoperators and Virtual Environments, MIT Press, Vol. 3, No. 4, 1994.

8. Pratt D., Barhman P., Locke J., Zyda M., "Ins -Insert ions of an Articulated Human in Networked Virtual Environment", Proceedings of AAAI Workshop on Simulation and Planning in High Autonomy Systems, Gainesville, 1994

9. Ohya J., Kitamura Y., Kishino F., Terashima N., "Virtual Space Teleconferencing: Real-Time Reproduction of 3D Human Images", *Journal of Visual Communication and Image Representation*, Vol. 6, No. 1, 1995, pp. 1-25

10. Greenhalgh, C., Benford, S., "MASSIVE, A Distributed Virtual Reality System Incorporating Spatial Trading", *Proceedings of the 15th International Conference on Distributed Computing Systems*, pp 27-34, Los Alamitos, CA, ACM, 1995

11. Singh G., Serra L., Png W., Wong A., Ng H., "BrickNet: Sharing Object Behaviors on the Net", *Proceedings of IEEE VRAIS '95*, 1995

12. Pandzic I.S., Capin T.K. Magnenat-Thalmann N., Thalmann D., "VLNET: A Net 8VLNET: ANetwork 3D Environment with Virtual Humans", Proc. Multi-Media Modeling, MMM'95, World Scientif 9 Press, Singapore, 1995

13. Capin T.K, Pandzic I.S., Magnenat-Thalmann N. Thalmann D., "Virtual Huma 8Virtual Humans for Participants in Immersive Virtual Environments", Proc. FIVE'95, Chapman and Hill, London, UK, 1995

14. Blaxxun Interactive, http://www.blaxxun.com

15. Waters R., Anderson D., Barrus J., Brogan D., Casey M., McKeown S., Nitta T., Sterns I., Yerazunis W., "Diamond Park &Diamond Park Social Virtual Reality System with 3D Animation, Spoken Interaction, and Runtime Modifiability", Presence, MIT Press, Nov. 96

16. Boulic R., Magnenat-Thalmann N., Thalmann D. "A Global Human Walking Model with Real-Time Kinematic Personification", The Visual Computer, Springer International, Vol. 6 (6), 1990

17. Pandzic I.S., Capin T.K., Magnenat-Thalmann N., Thalmann D., "Motor Functions in the VLNET Body-Centered Networked Virtual Environment", Virtual Environments and Scientific Visualization '96, Proc. Eurographics Workshop Monte Carlo, Blackwell Publishers, 1996

18. Pandzic I.S., Capin T.K., Magnenat-Thalmann N., Thalmann D., "Towards Natu 6Towar d s Natural Com Networked Collaborative Environments", Proc. FIVE'96, 96, Hsa,

19. Molet T., Boulic R., Thalmann D., "A Real-Time Anatomical Converter for Human Motion Capture", Eurographics Workshop on Computer Animation and Simulation, Blackwell Publishers, 1996, pp. 79-94

20. Shen J., Thalmann D., "Interactive Shape Des +I n Using Metaballs and Splines", Proceedings of Implicit Surfaces '95, Grenoble, pp.187-196

21. Boulic R., Capin T., Huang Z., Kalra P. Lintermann B., Magnenat-Thalmann N., Moccozet L., Molet T. Pandzic I., Saar K., Schmitt A. Shen J., Thalmann D., "The Humanoid Env 6The Humanoi Interactive Animation of Multiple Deformable Human Characters", Proceeding , P Eurographics'95, Blackwe 95, Blackwell Pub

22. Molet T., Aubel A., Capin T., Carion S., Lee E., Magnenat-Thalmann N., Noser H., Pandzic I., Sannier G., Thalmann D., "Anyone for Tennis", Presence, Vol. 8, No.2, MIT Press, April 1999, pp.140-156.

23. Noser H. Pandzic I.S., Capin T. K., Magnenat-Thalmann N., Thalmann D., "Playing Games through the Virtual Life Network", Proc. Artificial Life V, Nara, Japan, 1996, pp. 114-121

24. Capin T, Pandzic I, Noser H., Magnenat-Thalmann N., Thalmann D.," Virtual Human Representati and Communication in VLNET Networked Virtual Environments", IEEE Comp , I EEE Computer Application, Special Issue on Multimedia Highways, Vol. 17, No.2, 1997, pp.42-53

25. Pandzic I.S., Capin T.K., Lee E., Magnenat-Thalmann N., Thalmann D., " A flexible architecture for Virtual Humans in Networked Collaborative Virtual Environments", Proc. Eurographics 97, Blackwell Publishers, 1997, pp.177-188

26. Virtual Reality Markup Language, http://www.vrml.org/

27. StereoGraphics, http://www.stereographics.com

28. Ascension Technology Corporation, http://www.acension-tech.com

29. Virtual Research Systems, http://www.virtualresearch.com/

30. Pyramid Systems, http://www.pyramidsystems.com

31. Open Community Platform, Mitsubishi Electrical Research Lab., http://www.OpenCommunity.com

32. MASSIVE-3/HIVEK, University of Nottingham, http://www.crg.cs.nott.ac.uk/research/systems/MASSIVE-3/

33. Sense8, http://www.sense8.com

34. Seo H., Joslin C., Berner U., Magnenat-Thalmann N., Jovovic M., Esmerado J., Thalmann D., Palmer I., "V 8 VP A- A Windows NT Software platform for a Virtual Networked Amusement Park", Computer Graphics International 2000, IEEE Computer Society, June 2000, pp. 309-315

35. Silicon Graphics Open GL Optimizer, http://www.sgi.com/software/optimizer/

36. Kshirsagar S., Escher M., Sannier G., Magnenat-Thalmann N., "Multimodal Animation System Based on the MPEG-4 Standard", Multimedia Modeling 99, IEEE Computer Society, Ottawa, Canada, October 4-6 1999, pp. 215-232

37. Motion Picture Experts Group, http://www.cselt.it/mpeg/

38. H-ANIM Humanoid Animation Working Group, Specification for a Standard Humanoid Version 1.1, http://ece.uwaterloo.ca/~h-anim/spec1.1/

39. International Telecommunications Union, http://www.itu.ch

ICAT 2000

Self-Organized Objects with the GROWTH Model

Yoichiro Kawaguchi
The University of Tokyo
4-6-1 Komaba, Meguro-ku Tokyo 153-8904 JAPAN
yoichiro@iii.u-tokyo.ac.jp

Abstract

This paper describes an artistic approach to the generation of complex objects with the Growth model. The Growth model can easily define and transform such shapes interactively. The model utilized special density distribution functions, called "meta-balls", as its fundamental modeling primitives. Both gradual and catastrophic topological changes can be achieved with the Growth model due to the fluid, non-deterministic nature of the generating algorithm. Implementation of the model is described and illustrative examples drawn from previous and current work-in-progress are presented.

Key words: Growth Model, Self-Organization, Meta-balls

1. Introduction

The Growth model combines concepts from the self-organization with "meta-balls" techniques that carry out object modeling by means of distributed density functions[1]. These most recent techniques utilized a ray-tracing algorithm. The resulting works[2],[3].[4] are three-dimensional animation involving complex images. In this paper, we shall explain the Growth model for the efficient ray tracing of complex surfaces and organic objects. This model carries out metamorphic pattern transformations and makes possible the rendering of multiple texture-mapped surfaces with reflection or transparency areas.

Conventional three-dimensional image synthesis methods for modeling complex natural objects have usually required large amounts of object data. The procedures for generating shaded images, consisting frequently of thousands or even millions of objects data, could not be carried out without a great deal of computational expense. What has been sorely needed is the development of a new metamorphic model that will enable us generate morphologically varying shapes more efficiently flexibly [5]. Since that time we have been able to explore spiral structure generation with our Growth model. This model makes it easy to create a complex natural object based on the growth rules of shells and tendril plants, and with it we can generate a great variety of complex images[6].

Traditionally, representations of irregular, complex surfaces in computer-generated images have been based upon the assumption that those objects are fundamentally an assembly of simply defined polygons and patches, and that the objects are defined by free-form surface techniques. Subsequently, an increased degree of flexibility has been achieved by the rendering methods using a distribution function. This method has been very successful for artistic creation in rendering realistic three-dimensional shaded images of stretched objects with relatively complex characteristic representing organic surface features.

This paper focuses on these three-dimensional shaded images consisting of complex surfaces, which we call the "Growth model" (Fig. 1 and 2). At first, we describe how we structured the Growth model, including the Growth primitives, cluster structures, input parameters and the model's algorithm. Then, we explain how we carried out the growth scene simulation for making animations. After that, the Growth model is extended and improved to include reflection and refraction with density ellipsoids. (Fig.3 and 4) We also demonstrate a characteristic effect, namely multiple-texture mapped surfaces description, one of the most advanced and unique rendering techniques.

2. Structure of the Growth model

2.1 Primitives

Growth images are composed of many primitives. These primitives are defined by their center position, effective radius, weight, and other attributes. The center position parameters place a primitive on the specified local coordinates. *center(x_c,y_c)*

The effective radius is not the final image radius, so the effective radius is invisible, but rather defines the range of the density distribution. *radius (r_0)* The density reveals itself as a relative degree value for the meta-primitives potential and threshold. *weight (w_0)*, *threshold (t_0)* If the weight parameter is less than 0: *the primitive is invisible*. If the weight parameter is greater than 0: *the primitive is visible*

The attribute parameters include the following : *(r,g,b)* color of surface. *(dic)* the diffuse reflection constant for ambient light *(drc)* the diffuse reflection coefficient. *(src)* the specular reflection coefficient *(n)* the glossiness. *(trc)* the transmission coefficient. *(ph)* the reflection ratio. In addition, it is possible to vary these parameters with texture maps generated by the Growth

model. After that, we describe renderings with multiple reflection and refraction.

2.2 Cluster structures

In this section we will explain the Growth cluster structure. / Mt-balls" are are only furdamental primitives and, therefore, cluster concatenation is necessary to model complex shape structures. Clusters independently control their concatenated sub-clusters and shapes. For each cluster hierarchy it is possible to translate and rotate shapes about the cluster's local coordinate system and then transform the cluster to the global coordinate system of the scene. The "shape" den denote data of a shape file and upper-case letters indicate the generation level of each block (A "block" is is a gro clusters which are members of the same generation) For example, the 1st to 5th generation growths each develop two branches. The sub-scripts below each upper-case letter indicate whether the growth is a Troot branch"(0) o an ordinary "branch"(1), Dbranch"(1), for each stage. Only the first generation has a special main branch called the "root branch." One Growth-primitive consists of one block of shapes and clusters. A block style consists of objects having a common joint. Each cluster connects the next generation's blocks. Each cluster has up to (n)branches. At first the main branch of the new generation is related to the old generation. Shape A0 exists in first generation, and it is in the local coordinate of cluster A0. Cluster B0 is connected to cluster A . The coordinate axis (x,y,z) of its local coordinate system is shown. The tip of shape A0 is at the origin. The direction of growth of shape A0 is along its Y-axis. Shape B0 is rotated in the cluster B0's local coordinate system. As mentioned the shape B0 is connected with cluster B0. Cluster B0 is joined to cluster B1 in the same generation. They have the local coordinate system. The bottom of the main branch is the local coordinate origin (0,0,0). The direction of shape B0's growth is along this Y-axis. Shape B1 is rotated in this coordinate system.

Other branches are similar. When these cluster blocks are recursively defined, the Growth model can generate complex surfaces.

2.3 Growth parameters

It was mentioned above that a shape generated by the Growth model could be broken down into smaller parts such as branches or joints. The Growth parameter is very important as a factor, which generates the model recursively, in the detail, as far as the tip, according to the Growth principle. The principle is another example of the principle of hierarchical multiplication of a recursively expanded self-similar structure.

The Growth parameters include the following:

(1)Center coordinates of the bottom of the main root branch.
(X-bottom, Y-bottom, Z-bottom)

(2) Center coordinates of the top of the main root branch.
(X-top, Y-top, Z-top)

(3) The angle between the root and the next generation branch.
(X-angle (0). Y-angle (0). Z-angle (0))

(4) The angle between branches at the same generation.
(X-angle(1), Y-angle(1), Z-angle(1))
(X-angle(k), Y-angle(k), Z-angle(k))
(X-angle(n), X-angle(n), Z-angle(n))
(0<=k<=n, n>0)

(5) Effective extension of the vibrating angle between the root branch and then next generation branch.
(X-min-angle(0), Y-min-angle(0), Z-min-angle(0))
(X-max-angle(0), Y-max-angle(0), Z-max-angle(0))

(6) Effective extension of the vibrating angle between non-root branches
(X-min-angle(1), Y-min-angle(1), Z-min-angle(1))
(X-min-angle(k), Y-min-angle(k), Z-min-angle(k))
(X-min-angle(n), Y-min-angle(n), Z-min-angle(n))
(0<=k<=n, n>0)

(7) The portion of increase and decrease of angle between a branch and the next generation branch.
(X-step, Y-step, Z-step)

(8) The growth ratio of the next generation branch to the prior branch's joint
(scale(0))

(9) The growth ratio of the next generation branch to the prior branch.
(scale(1))
(scale(k))
(scale(n))
(0<=k<=n, n>0)

(10) The maximum radius of joint (radius).

(11) The minimum radius of joint (limit).

(12) The attribute data of branch, joint and flower.
(root branch-attribute)
(joint attribute)
(branch-attribute)
(top-branch-attribute)

2.4 Growth algorithm

In this section, the generating algorithm of the Growth model is presented.

(1) The parameters if the model is read in.

(2) The length of each branch, its thickness, growing direction and other attributes are transferred to the generating routine of the Growth model.

(3) The generating routine checks the limit radius.

(4) The data for the branches and accompanying joints are generated.

(5) The next generation branch is generated.

(6) The parameters to be transferred to the next generation branch are computed.

(7) Next a recursive call to the routine takes place.

(8) After the end of this routine's generating the next generation is the start for growing the branch.

(9) Of course, this does not affect parts that are not branched.

(10) The generation of branches is described in the next three items.

(11) First, parameters necessary for generating the branches of the next generation are computed.

(12) And once again a recursive call takes place.

(13) After the end of generating the branches, this routine is finished and returns to the origin of its call.

The Growth model is implemented with a generating algorithm like the one just described. In the actual program, several special coordinates are added for processing special case like the tip, blooming flowers, etc. The cluster structure and generating algorithm are planned as above because they involve a largely recursive structure, and therefore the program and data structure are also recursive. This makes possible a very compact, efficient program.

3. Summary

We have presented the Growth model as an application example of complex object modeling, and have shown that this model is a powerful tool for representing and rendering images dynamically. The Growth model is realized by means of special density distribution functions called "meta-balls." The use of this type of primitives enable the model to give full play to its power to render organic objects that are difficult to define with conventional modeling techniques. Since the Growth model does not completely define shapes with deterministic methods, both gradual and catastrophic (sudden) topological change can be carried out interactively by means of just a few parameters. To illustrate this technique, we presented the Growth model, which were developed in order to visualize metamorphic change on the basis of growth principles. To simulate our research we have also tried to develop an efficient hypothesis regarding natural mechanism and we have tried to investigate how this hypothesis can help model and generate various growth processes that are fundamental to many natural phenomena. We hope that this approach in trying to analyze the basis of natural growth objects can help us to develop new artistic techniques utilizing natural scientific principles.

Figure 1 Complex branching of the GROWTH model (from "Neurar",1996)

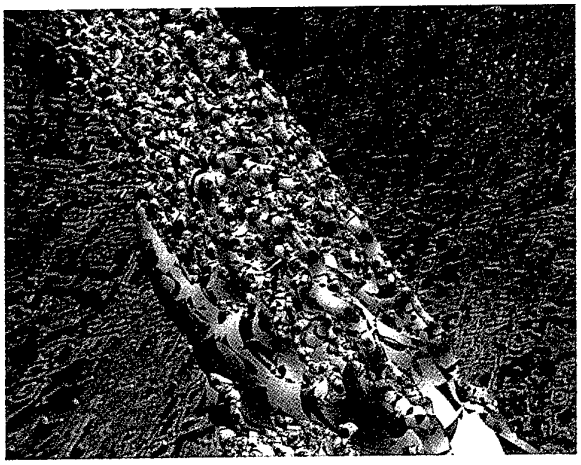


Figure 2, Highly developed branching (from "Neurar", 1996)

References

1 Yoichiro Kawaguchi, "Morphogenesis", JICC Publishing Inc., 1984.

2 Kawaguchi Yoichiro, "COACERVATER", NTT Publishing, Tokyo, 1994

3 Yoichiro Kawaguchi, "YOICHIRO KAWAGUCHI" (ggg-books 38), Trans Art, 1998.

4 Yoichiro Kawaguchi, "LUMINOUS VISIONS" (Video), Odyssey Productions, 1998.

5 Kawaguchi Yoichiro, "A Morphological Study of the Form of Nature", Proceedings of SIGGRAPH '82, Vol.16, No3, July 1982.

6 Yoichiro Kawaguchi, "The Art of Growth Algorithm with Cells", Artificial Life V, pp.159-166, 1997.

Figure 3 Surfaces created dynamically, using positive and negative meatballs. (from "Nebular" 2000)

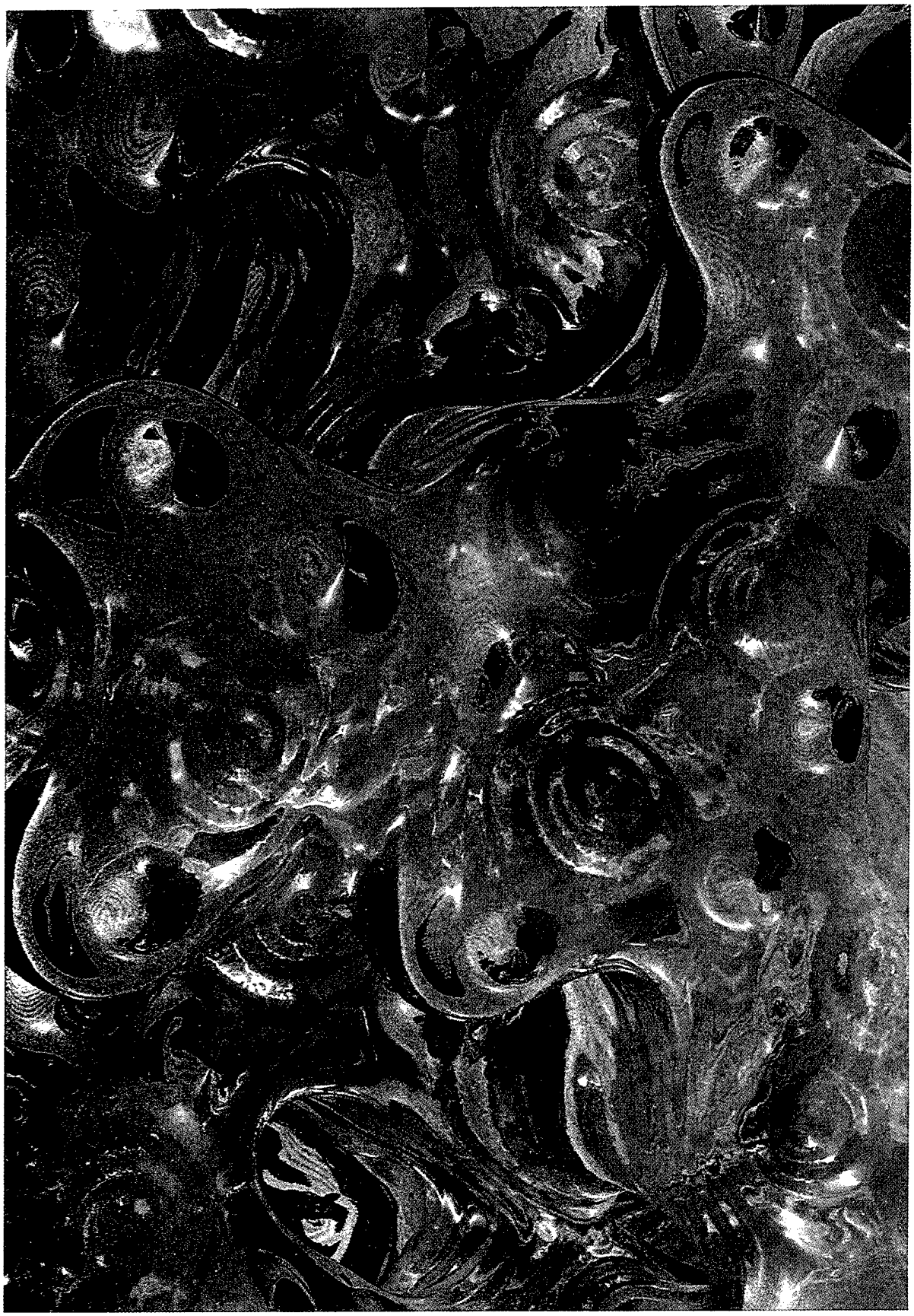


Figure 4. Enlarged view of surface created by meatballs.
(from "Nebular", 2000)

A Story of SPIDAR

Makoto SATO

Precision and Intelligence Laboratory
Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku, Yokohama, 226-8503, Japan
Tel: +81-54-924-5050 Fax: +81-45-924-5016
Email: msato@pi.titech.ac.jp
URL: http://sklab-www.pi.titech.ac.jp/

For a decade I have been working in the research area of virtual reality, especially developing the haptic interface devices. My works mostly concern with hand- and fingers-based interface device with force feedback for man-machine interaction in the virtual environment. In 1991, I had started with a proposal of a new string-based haptic display named SPIDAR [1], in which the word "SPIDAR" is stand for "SPace Interface Device for Artificial Reality". It was a kind of string-based system, which used DC motors, pulleys, and strings. The strings that attached to a finger of the user were used to calculate finger's position in the virtual space by measuring their lengths. At the same time, by controlling the tensions of the strings, force feedback can be generated at user's fingertip.

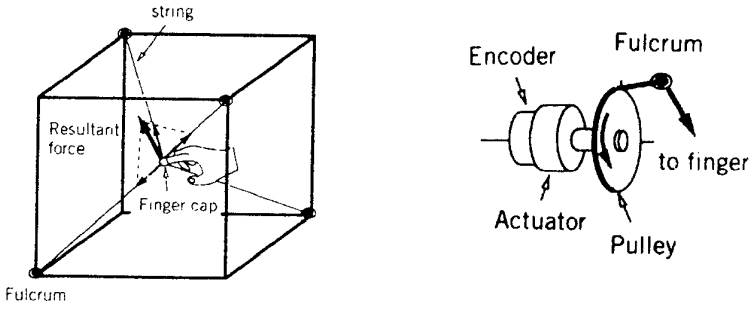


Fig. 1 SPIDAR System

In the later years, I had continued my work by proposing SPIDAR-II [2], an improved version of SPIDAR that allowed a user to use two fingers, which were a thumb and an index finger, to be able to grasp a virtual object. Two sets of this new system could be combined together to become a Both-Hands-SPIDAR [3]. A user could perform a kind of assembly task such as Fit-The-Face with the cooperative works of the left and right hands.

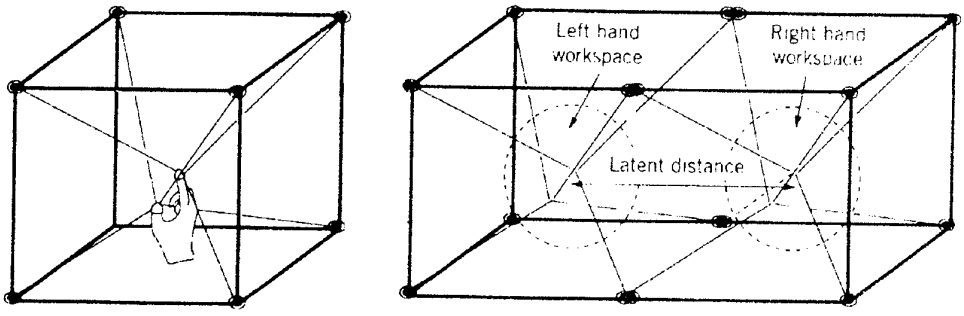


Fig. 2 SPIDAR-II and Both-Hands-SPIDAR

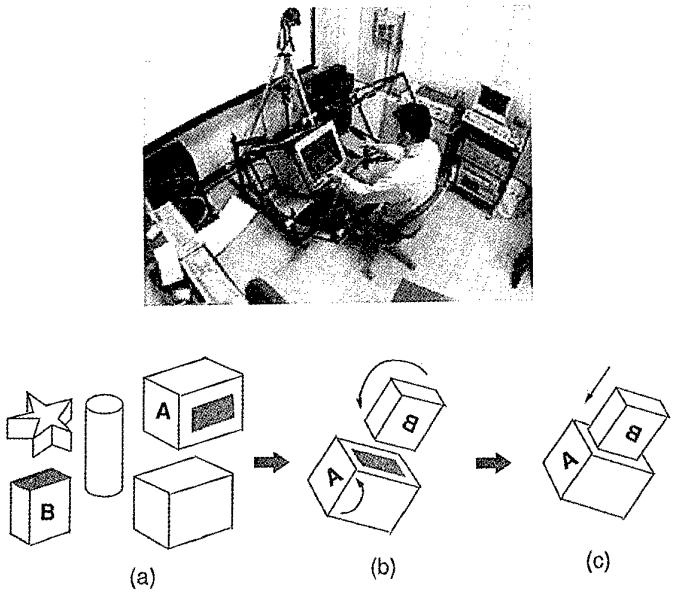


Fig. 3 Both-Hands-SPIDAR and Fit-The-Face task

Two systems of SPIDAR-II were installed in two different locations, which a user at the local site and a user at the remote site could perform a Hand-Over task successfully. This application of SPIDAR-II was a Networked-SPIDAR [4].

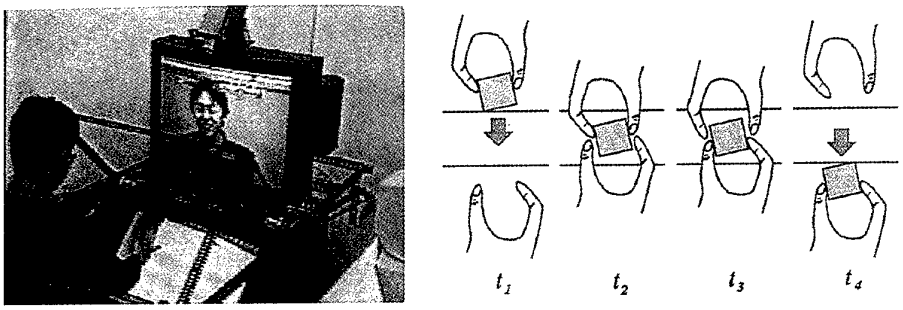


Fig. 4 Networked-SPIDAR and Hand-Over task

The frame of SPIDAR was enlarged to the human-scale where a user could completely immerse him into the simulated virtual space. The shooting task in the basketball game was simulated. A user stood within the frame of Big-SPIDAR [5], held a virtual ball with both hands, and shot the virtual basket with the realistic sensation as same as perform with the real ones. He could fell the spherical shape of the virtual ball and the simulation of weight of ball was one of the important factors that he had to consider how much strength must be applied to throw the ball to the basket and made score successfully. The user who completely immersed the virtual environment could perceive haptic, visual, as well as audio feedbacks, provided by Big-SPIDAR almost as same as the feedbacks he could perceived in the real world. The Virtual Basketball was successfully demonstrated in the Electric Garden during SIGGRAPH'97 in Los Angles, U.S.A.

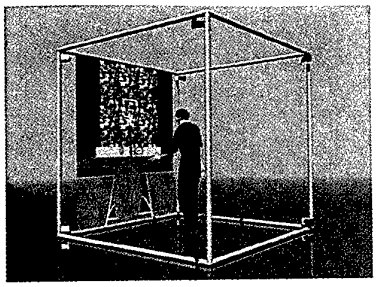


Fig. 5 Big-SPIDAR

Works to improve the SPIDAR system have been continuously developed. Recently, SPIDAR-G [6] is proposed as a haptic interface device with 6 degrees of freedom (DOF); 3 DOFs for translation and 3 DOFs for rotation. This system shows satisfactory performance as a three-dimensional interface device for 3D virtual environment interaction. Combining with a special designed of a grip, SPIDAR-G is added one more DOF when the grip is closed and released to become a new 7-DOFs string-based haptic interface device. The user can manipulate the virtual objects by translating and rotating in any direction. In addition, the weight of virtual objects can be simulated according to the physical gravity during the manipulation of the virtual objects.

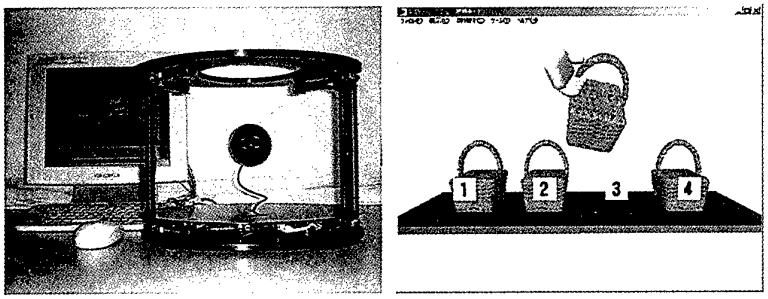


Fig. 6 SPIDAR-G and a simulation of weight of the virtual objects

Another improved version of SPIDAR system is two-handed with multi-fingers typed of SPIDAR named SPIDAR-8 [7]. This new system allows a user to use thumb, index, middle, and ring finger on both left and right hands to manipulate the virtual objects in the simulated virtual world. The user can perform the cooperative work using both hands and perceived force feedback at eight fingertips while manipulating the virtual objects. The simulation of the Virtual Rubik's Cube is implemented and obviously showed the abilities of the system. Again, SPIDAR-8 was selected to be one of the contributors of Emerging Technologies of SIGGRAPH 2000 demonstrated in New Orleans, USA.

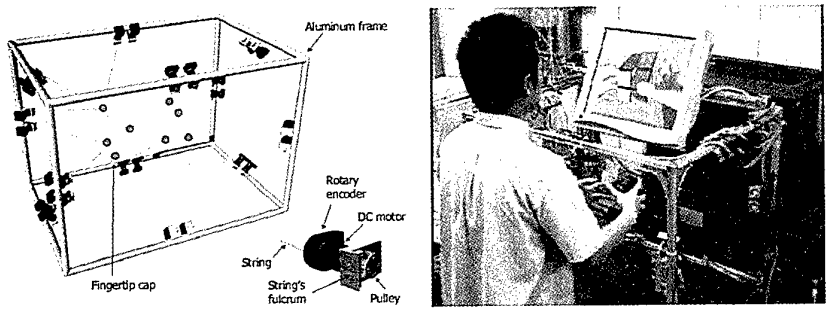
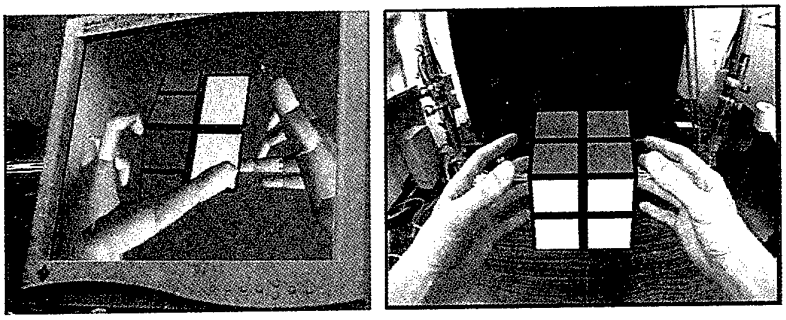


Fig 7 SPIDAR-8 and Virtual Rubik's Cube

Finally, I would like also to mention about the tentative plan of my works. Closely related with virtual reality (VR), mixed reality (MR) is become more and more attractive and challenging topic. Haptic display system, provided its user with realistic sense of touch, is believed to give enhanced performance by providing also the sense of immerse of the user into the virtual environment. Using SPIDAR-8, image sequences of user's real hands are to be used instead of computer graphic virtual hands manipulating the virtual objects in the virtual world. Such Visuo-Haptic system is now under implementation. It is believe to be a contribution of great deal in the both VR and MR system.



Virtual Reality Mixed-Reality

Fig. 8 VR and MR Environment

References

[1] Y. Hirata and M. Sato, "3-Dimensional Interface Device for Virtual Work Space", Proceedings of the 1992 IEEE/RSJ International Conference on IROS, 2, pp. 889-896, 1992.

[2] M. Ishii and M. Sato, "3D Spatial Interface Device Using Tensed Strings", PRESENCE-Teleoperators and Virtual Environments, Vol. 3 No. 1, MIT Press, Cambridge, MA, pp. 81-86, 1994.

[3] M. Ishii, P. Sukanya and M. Sato, "A Virtual Work Space for Both Hands Manipulation with Coherency Between Kinesthetic and Visual Sensation", Proceedings of the Forth International Symposium on Measurement and Control in Robotics, pp. 84-90, December 1994.

[4] M. Ishii, Masanori Nakata, and M. Sato, "Networked SPIDAR: A Networked Virtual Environment with Visual, Auditory, and Haptic Interactions", PRESENCE-Teleoperators and Virtual Environments, Vol. 3 No. 4, MIT Press, Cambridge, MA, pp. 351-359, 1994.

[5] Y. Cai, S. Wang, and M. Sato, "A Human-Scale Direct Motion Instruction System Device for Education Systems", the IEICE Transactions, Vol. E80-D, No. 2, pp. 212-217, 1997.

[6] S. Kim, M. Ishii, Y. Koike, and M. Sato, "Design of a Tension Based Haptic Interface: SPIDAR-G", Proceedings of World Multiconference on Systemics, Cybernetics, and Informatics: SCI 2000, pp. 422-427, July 2000.

[7] S. Walairacht and M. Sato, "4+4 Fingers Haptic Interface Device for Virtual Environment", Proceedings of World Multiconference on Systemics, Cybernetics, and Informatics: SCI 2000, pp. 427-433, July 2000.

Profile

Makoto SATO graduated in 1973 from Department of Physical Electronics, Faculty of Engineering, Tokyo Institute of Technology, where he obtained the Doctor of Engineering in 1978. He became an assistant in the same faculty, and now a Professor of Precision and Intelligence Laboratory, Tokyo Institute of Technology. He is engaged in researches on pattern recognition, image processing, and virtual reality.

Shape Forming by Cutting and Deforming Operations

Koichi Hirota*, Atsuko Tanaka**, Toyohisa Kaneko***, Michitaka Hirose*

*Research Center for Advanced Science and Technology, The University of Tokyo
**Department of Information and Computer Sciences, Toyohashi University of Technology
***Central Research and Development Laboratory, OMRON Corporation

Abstract

In this paper, an implementation of visual and haptic feedback of deforming and cutting operations is discussed. In the implementation of the deforming operation, the surface shape is represented by a geometric model while the physical reaction is simulated using a spring model. The deformation of the spring model is reflected onto the geometric model by using the interpolation technique. In the implementation of the cutting operation, we realize visual and haptic feedback of the cutting operation remarking on the geometric and physical aspects, respectively. Combining the deforming and the cutting environment, we successfully implemented a work space in which we can form and design shapes through operations similar to clay modeling.

Key words: cutting, deforming, designing shape, virtual environment, force feedback

1. Introduction

Shape forming is one of promising application area of virtual reality, and various studies on virtual clay-modeling has been carried out. Deforming and cutting operations have been typical means to create shapes, and many modeling softwares provide the deforming and cutting operations. However, most of them do not provide the direct manipulation interface for those operations.

When we are going to realize realistic cutting and deforming operations in environments, we need to implement object models that behave similarly to the real objects according to operations by the user. Since such behavior derive from the physical nature of objects, physically based modeling and simulation is desirable to increase reality in virtual environment. Also, if we are going to feedback the sensation of force in the interaction, computation of force based on the physical model is indispensable. However, it has been a problem that the physically based simulation of cutting and deforming operations generally requires more computation cost compared with the simulation only by geometrical models. Consequently, if we apply the physical model, the complexity of the shape with which we can interact in real-time is strictly limited. On the other hand, importance of presenting force sensation during operations came to be recognized[1], and the computation algorithm of haptic rendering came into an important topic of study[2].

In our study, we investigate methods to simulate cutting and deforming operations with force feedback. Also, by integrating these simulation methods, we implement a virtual modeling environment. Although various tools are used for cutting operations in the real world, cutting operations using a knife or a fret saw is intended in this study.

As we stated above, the complexity of the physical model is limited because of the computation cost. A problem of previous approaches to implementing physically base models is that both of physical and geometrical models are sharing a same structure (i.e., the model of same complexity).

We propose an idea to use two models of different complexity for physical and geometric simulations, respectively[3] (i.e., physical model and geometric model). Also, the geometric model is shared by both cutting and deforming operations. We employ a model in which the shape of objects are defined as a collection of tetrahedral elements (i.e., tetrahedron model).

2. Deforming Operation

These are several studies on the implementation of deformable objects as follows. As a geometric approach, the idea of Free Form Deformation [4] has been proposed in which smooth deformation of shape is realized by applying an interpolation technique to the computation of deformation. However, it is a control-point based approach, and we can not use this approach for direct operation in the virtual environment. Also, to solve the problem, the idea of Direct Deformation Method[5] is proposed. However, if we are going to feedback force, these models must be combined with other model that can compute the interaction force.

There are may studies to introduce physically based models to simulate deformation in computer graphics and virtual reality. In those studies typically two models have been applied: Spring Network model[6]

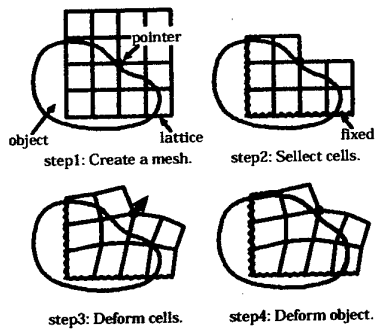


Fig.1: Process of Deforming Operation

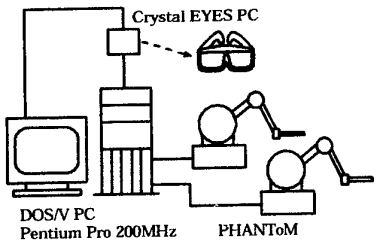


Fig.2: System Construction

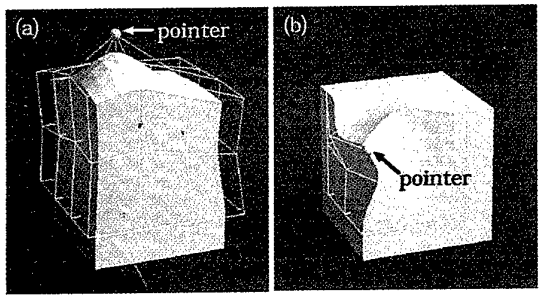


Fig.3: Examples of Deforming Operation

and Finite Element Method model. According to previous studies, the FEM model is not suitable for real-time simulation because it requires high computation cost. Although there is a study to investigate a fast computation method of linear FEM model[7], this approach is not applicable to the simulation of large deformation. In contrast, the Sprig-Network model requires less computation time, and consequently the higher update rate is attained. This is why most studies on the haptic interaction with deformable objects have applied the Spring Network model.

As we stated above, in previous studies, processes of physical simulation and geometric representation are sharing a same structure (e.g., the network of spring is constructed along edges of the polygon model). In this approach, as the resolution of geometric model becomes higher, the physical model also becomes complex.

In our approach, we use separate models for the deforming simulation and the representation of shape, respectively. We employ a spring network model for the physically based simulation of deformation and the result of the simulation is reflected on the precise geometry model. Also, by introducing the condition of breaking into the spring network model we realize the tearing operation.

2.1 Implementation of Deforming Operation

The spring network model consists of cubic cells that are connected with each other at vertices. In each cell, 28 springs are spanned between all of the combination of two nodes among eight nodes. Figure 1 shows the process of deforming computation schematically. Firstly, a spring network that covers a cubic area is created the stylus tip. Next, spring cells that is out of the object volume is deleted so that the shape of the spring network becomes more close to the object shape (i.e., so as to attain better approximation in deforming characteristic). Also, the boundary condition of the spring network is defined (e.g., the vertices at the stylus tip is fixed to the stylus).

During the operation, the position of stylus tip is updated according to the motion of the user, and the deformation of the spring network is simulated. Further more, the shape of the geometric model is changed by computing the position of each node of the tetrahedron model using the interpolation technique based on the algorithm of 3-D Coons Patch.

The block diagram of the system for the prototype implementation is shown in Figure 2. In the system, we use a PC (AT compatible, dual Pentium Pro 200MHz) with an accelerated graphics card (Fire GL 1000, Diamond Multimedia) for the simulation and visual rendering, two PHANToM devices (1.5A, SensAble Technologies)[8] for haptic interaction, and a LC shutter grasses (Crystal Eyes PC, Stereo Graphics) to provide stereoscopic image.

The beginning and the ending of the deforming operation are transmitted to the system by pressing and releasing the button switch of the stylus, respectively.

Figure 3 shows examples of the deforming operation. In figure (a), the user is pulling up about the center of top surface, where a small sphere indicates the position of stylus tip and the spring network for the deforming simulation is represented as a wire-frame mesh. We could obtain smooth deformation of polygon model from the deformation of the coarse spring network model. Figure (b) shows an example

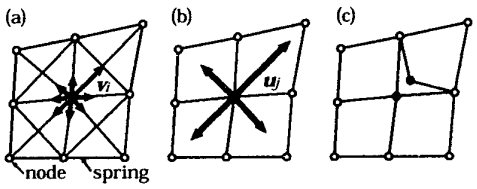


Fig.4: Dividing Spring Model

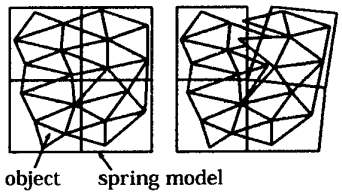


Fig.5: Dividing Tetrahedral Model

of the twisting operation. Since the device we used in this experiment is not capable of representing torque, the sensation of twisting moment is not fedback to the user.

2.2 Implementation of Tearing Operation

Tearing operation is often observed in the clay modeling especially to adjust the volume of clay during the modeling task. The material is torn when the internal stress caused by the external operation exceed the maximum stress that the material can bear. In our experiment, we realize the tearing operation by computing the internal stress of the spring network and locally dividing the network according to the stress.

As is described above, the spring net work consists of cubic cells, and neighboring cells are connected at vertices with each other. In our model, we assumed that the material is broken when the tensile stress exceeds a limit(see Figure 4). We computed the maximum tensile stress in a approximate way. Also, we assumed that the crack is caused perpendicular to the orientation of tensile stress. This assumption is introduced into the model as the algorithm of grouping cells sharing a vertex when the connection at the vertex is cut by the stress. Namely, the grouping is performed based on whether the contribution of each cell to the stress is positive or negative.

The division of the spring network is reflected on the geometry model (i.e., the tetrahedron model) by dividing the tetrahedron model and distributing tetrahedral that are crossing the surface of spring cells exposed by the division (see Figure 5).

Since the volume that is divided by the tearing

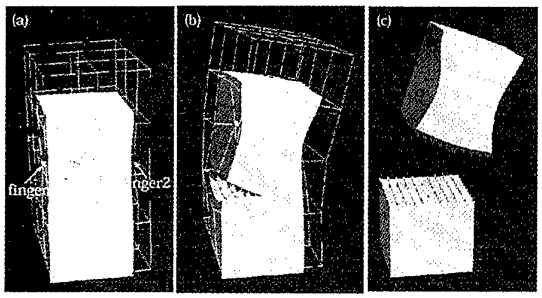


Fig.6: Process of Tearing Operation

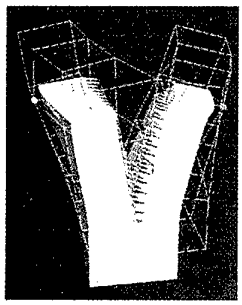


Fig.7: Tearing Operation with Both Hands

operation depends on how the user grasps the object. In our implementation, the user grasps the object by two stylus points corresponding to two PHANToM devices. Figure 6 shows the steps of this operation, where the stylus tips are represented by small spheres. Also, it is possible for the user to tear an object apart left and right (see Figure 7).

3. Cutting Operation

Cutting is one of the most basic operations among various tasks involving shape forming and surgical simulation. However, there are few studies that deal with the force applied during the cutting operation. There is an investigation that implemented a sculpturing operation in a virtual environment[9], where the voxel-based model was used to define the shape. There is also an investigation that introduced force feedback during the sculpturing operation[10]. In that investigation, the force fed back to the operator was determined only from the velocity of cutting the object.

One approach to implementing the cutting operation is to divide the objects geometrically based on the trajectory of the cutting tool. There is an investigation in which the geometrical cutting operation was implemented via a boolean operation on the polygon-based model[11]. However, in that investigation, the analysis on the cutting force was insufficient.

3.1 Computation of Cutting Force

We define the cutting edge as a finite set of discrete points (i.e., discrete edges). By computing the force on each discrete edge, we obtain the approximate distribution of force on the edge. We assume a line-type cutting edge. Namely, the cutting edge is omni-directional and the rotation of the cutting edge around its axis does not cause reacting torque. Also, we assume that the discrete edges are independent of each other. Namely, the status of a discrete edge does not affect the computation of other discrete edges.

The object deforms when the force from the cutting tool is applied. We assume that the force affecting on a discrete edge is proportional to the displacement at the point where the discrete edge collides with the object. To represent this relationship, we introduce the stiffness coefficient.

In the simulation, each discrete edge holds the position of two points. One is the present position of the edge. This is the same as the position where the cutting edge collides with the deformed object. The other is the position of the present colliding point when the deformation is relaxed. This is the same as the position where the cutting edge collides with the object in a nondeformed state. Consequently, the deformation of the object on each discrete edge is calculated as the disparity between those positions.

We modeled three kinds of typical forces that affect the cutting edge: fractional force , cutting resistance, and viscous drag[12]. The progress of the cutting operation is represented by moving the cutting edge in the object, namely, by updating the position of the colliding point based on the force affecting the discrete edge.

The fractional force is introduced to represent the friction between the cutting edge and the object. The force does not contribute to the destruction of the material (i.e., does not contribute to cutting).

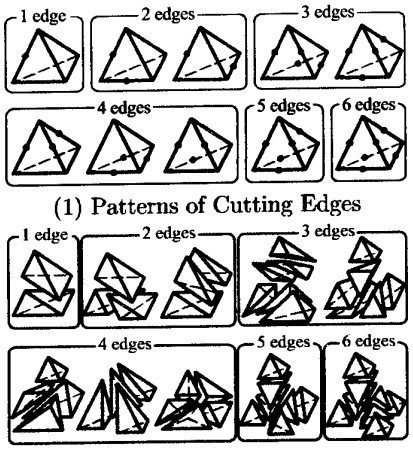
Mechanical cutting is an operation that destroys a part of the material due to the force applied from the cutting edge. This destruction is governed by the shearing force. In our model, the shearing force is approximately computed and the part of the material is destroyed when the shearing force exceed the maximum shearing force that the material can bear.

Viscous drag is the force that is caused as a function of the velocity of the cutting edge. In our model, we assume that the viscous friction is proportional to the velocity of the cutting edge.

3.2 Geometric Cutting

In our implementation, the geometric change caused by the cutting operation is represented by dividing tetrahedra colliding with the trajectory of the cutting edge. The dividing patterns of each tetrahedron is summarized in Figure 8.

Following is the computation flow of the cutting operation (see Figure 9): Firstly, the trajectory of the cutting edge is recorded, and the trajectory surface is defined as a set of triangular patches. Next, the cross points between those triangular patches and edges of tetrahedral cells in the object model are computed. Each cell is divided into parts on those cross points, and each part is re-divided into tetrahedral cells. Finally, the neighboring relation of cells is updated, and the whole object is divided into fragments. The proposed algorithm provides a fast method to compute intersection between the cutting edge and the object approximately.



(1) Patterns of Cutting Edges

(2) Patterns of Dividing Tetrahedral

Fig.8: Dividing Patterns

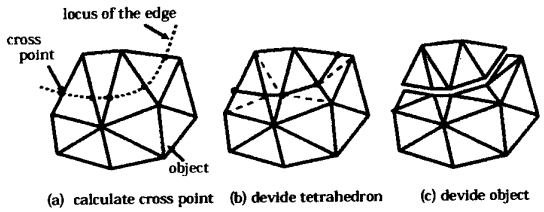
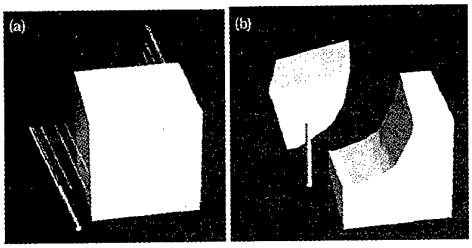


Fig.9: Process of Cutting Operation

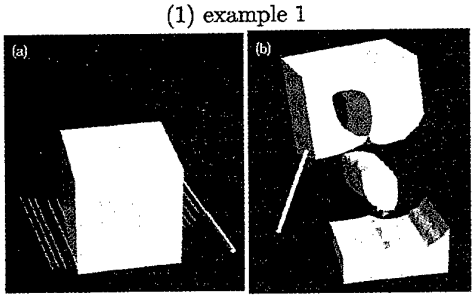
Figure 10 shows examples of cutting operations and resulting shapes. In the case of (1), the shape consisting of 6000 tetrahedral is colliding with the trajectory surface consisting of 18 polygons and took about 4 seconds for the geometric processing.

3.3 Representation of force while cutting

By combining the algorithm of haptic and geometric computations, we implemented a cutting environment. For the fast collision detection between objects and the cutting edge, we employ the voxel mesh surrounding the object. Voxels containing a part of an object are marked in advance to the operation, and we regard that the each discrete edge is colliding with the object when it is in a marked voxel.



(1) example 1



(2) example 2

Fig.10: Cutting Process

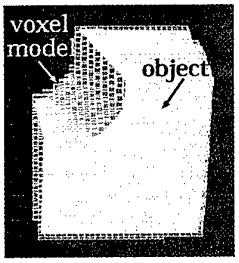


Fig.11: Voxel Model for Force Feedback

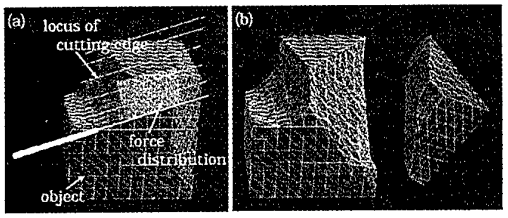


Fig.12: Representation of Cutting Force

An example of the voxel model is shown in Figure 11. Also, an example of cutting operation with force is shown in Figure 12. As is observed in this figure, the distribution of force on the cutting edge is restricted to the part that is colliding with the voxel model. After the operation, the voxel model is deleted.

4. Application to Shape Forming Task

We integrated the algorithms proposed in previous sections into a virtual environment, and experimentally applied the environment for a shape forming task. An example of the process and the result of a user's operation is shown in Figure 13 and 14. The shape of a petal is created from a square panel by deformation. Also, original shapes of the leaf were quarryed from a rectangular object, and they were stretched and flattened so that they look like leaves. The stalk was created in a similar way. Finally, all those elements created above are arranged in a space.

5. Conclusion

We proposed an approach to realize cutting and deforming operations with force feedback. To attain both of the fast update rate of physical simulation for force feedback and the precise representation of geometric shape, we defined coarse physical model and fine geometric model and combined them with each other. Also, by sharing a geometric model in both cutting and deforming operations, it became possible to switch these two operations without the transforming the internal representation of the object.

One of studies to be carried out in the future is to observe how the sensation of force is used in shape forming tasks and to evaluate the contribution of the sensation to the efficiency. The prototype system implemented in this study will provide an environment for this kind of future study.

References

1. Burdea G.: *Force & Touch Feedback for Virtual Reality*, A Wiley-Inter-Science Publication, New York, 1996.
2. Salisbury K., Brock D., Massie T., Swarup N., Zilles C.: Haptic Rendering: Programming Touch Interaction with Virtual Objects; *Proc. Symp. Interactive 3D Graphics*, pp.123-130, 1995.
3. Tanaka A., Hirota K., Kaneko T.: Deforming and Cutting Operation with Force Sensation; *J. Robotics and Mechatronics*, Vol.12, No.3, pp.292-303, 2000.
4. Sederberg T.W., Parry S.R.: Free-Form Deformation of Solid Geometric Models; *SIGGRAPH '86*, pp.151-161, 1986.
5. Yamashita J., Fukui Y.: A Direct Deformation Method; *Proc. VRAIS'93*, pp.499-504, 1993.
6. Norton A., Turk G., Bacon B., Gerth J., Sweeney P.: Animation of fracture by physical modeling; *Visual Computer*, Vol.7, pp.210-219, 1991.
7. Hirota K., Kaneko T.: Virtual Elasticity; *Proc. HCI'99*, pp.1040-1043, 1999.

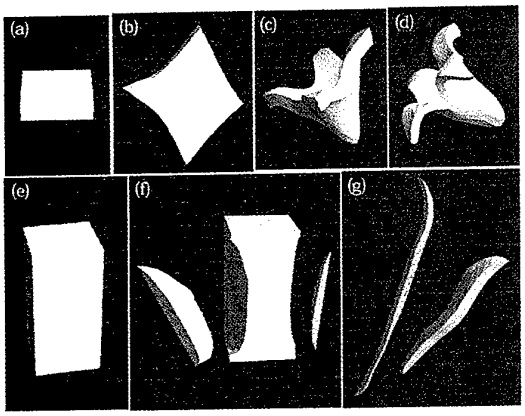


Fig.13: Steps of Forming Shapes

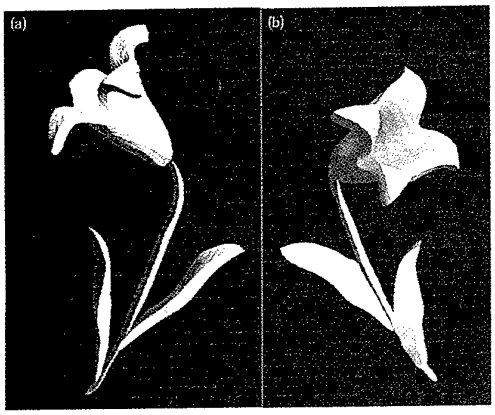


Fig.14: Example of Created Shape

8. Massic T.H.: Initial Haptic Explorations with the Phantom: Virtual Touch Through Point Interaction: Master's Thesis, M.I.T., 1996.

9. Galyean T.A.: Sculpting: An Interactive Volumetric Modeling Technique, *Computer Graphics*, vol.25, no.4, pp.267-274, 1991.

10. Yamamoto K., Ishiguro A., Uchikawa Y.: A Development of Dynamic Deforming Algorithms for 3D Shape Modeling with Generation of Interactive Force Sensation; *Proc. VRAIS '93*, pp.505-511, 1993.

11. Tanaka A.. Hirota K., Kaneko T.: Virtual Cutting with Force Feedback; *Proc. VRAIS '98*, pp.71-75, 1998.

12. Hirota K., Tanaka A., Kaneko T.: Representation of Force in Cutting Operation; *Proc. IEEE VR '99*, p.77, IEEE, 1999

Haptic texture presentation in a three-dimensional space

Yasushi Ikei, Masashi Shiratori, and Shuichi Fukuda
Tokyo Metropolitan Institute of Technology
6-6, Asahigaoka, Hino-shi, Tokyo, Japan
{ikei, fukuda}@tmit.ac.jp, shiratori@krmgiks5.tmit.ac.jp

Abstract

This paper describes a new haptic display which imparts surface texture information on a three-dimensional (3D) object to the user's fingertip. First, a pin array type display device, the Texture Display F10, equipped with ten vibratory pins is introduced. The discrimination of texture patterns in a 3D space is investigated using the F10 display. A force feedback device (the PHANToM) is attached to the F10 to provide a repulsive force from the surface during the exploration of a finger on a texture. The difference threshold of a wavelength was measured to investigate the basic performance of the new composite haptic display. The waveform discrimination among three different waves was successfully demonstrated by using the display, which indicated a partial display capability.

Key words: Haptic texture, Pin-array Display, Tactile and Force Feedback

1. Introduction

A haptic texture sensation is evoked through an interaction between a part of a human body, particularly at a finger, and an object's surface which has a relatively small variation in properties. The properties related to haptic sensation consist of micro geometry, stiffness, the coefficient of friction, thermal conductivity and capacity etc. We observe an intricate texture sensation integrated from all these properties. The texture sensation is not clearly elucidated yet, although Hollins et al. [1] addressed a three-dimensional perceptual space based on analysis of limited common objects. There are few researches discussing a texture sensation in a physical 3D space which include free hand motion.

Displays for texture sensation have been developed in a restricted manner since Minsky [1] demonstrated a two dimensional force feedback device for presenting virtual textures. As the device for textures needs to reflect dense and minute changes on a surface in addition to covering fast and broad hand motion, the construction of the device is extremely difficult. Thus far, haptic texture rendering has been implemented with two approaches; producing stimulus distribution directly on a skin surface, and conveying the force perturbed at a textured surface by a force-reflecting device, somewhat indirectly. This approach is discussed within the method to render the shape of 3D objects as producing local perturbation [3, 4]. However, the method is not demonstrated with a quantitative experiment. The former approach is related to the devices that convey information to a handicapped person. An array of vibratory elements has been used for the purpose of transmitting a symbolic code or characters to the back or the fingertip. The device for non-symbolic information in this course started only recently as a novel virtual reality interface.

We have investigated the pin array type display for presenting haptic textures [5]. The display is equipped with fifty pins concentrated within a fingertip area, however the display is too large and heavy to be attached to the finger. A new type device was produced by changing actuators and reducing the number of pins to shrink the size appropriate for finger mount in order to enable 3D exploration of surfaces. The new display was reinforced again by mounting it to a force feedback device to provide it with both capabilities of cutaneous and kinesthetic stimulations. The next two sections describe the pin display which can be attached to the user's finger and allows it free three-dimensional motion. The two succeeding sections state the display with force feedback and its evaluation results.

2. Texture Display F10

The Texture Display F10 (Figure 1) is a compact haptic display which can be attached to the user's fingertip allowing the user three-dimensional exploration of surfaces of a spatial object. The F10 has ten pins driven by bimorph-like piezoelectric actuators (LSD2665X, Megacera, Inc.). The pins are arranged in a matrix of two columns and five rows with a 3-mm spacing as illustrated in Figure 2. The frame and contact pins of the display are fabricated of photo-curing resin. The dimension was determined from the size of the actuator. The weight of the display except the wiring is about 30 grams. The amplitude of each pin is controlled in forty ways in the range up to about 22 microns. Sensation scaling over the amplitude range through the JND method revealed that the adept users could distinguish fifteen levels of sensation intensity. We formed forty levels of output intensity change on the display along with these fifteen levels of sensation intensity.

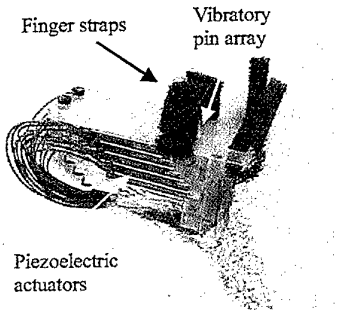


Fig. 1 Texture Display F10. Ten vibratory pins are driven individually by piezoelectric actuators. The F10 is mounted to the index fingertip with finger straps.

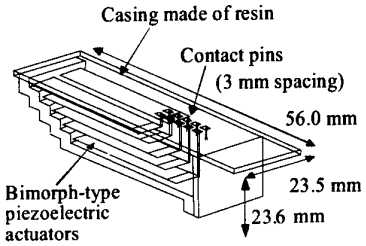


Fig. 2 The frame of Texture Display F10. The frame and contact pins were fabricated of photo-curing resin.

3. Performance test of the F10 display

A discrimination test regarding similar texture patterns was conducted to investigate the presentation quality of the F10 display. The textures provided for the test are shown in Figure 3. The textures have regular intensity distribution in normalized gray scale (ranging between 0.0 to 1.0), which were created by a sinusoidal function or its combination. This gray scale intensity was linearly mapped to the fifteen sensation intensity levels of the F10. The textures were grouped in three sets for three independent sessions. The size of every texture was 120 x 90 mm^2. The wavelengths of the sine functions are 40, 30, and 24 mm for the test set 1 (Figure 3a), and 30, 22.5, and 18 mm for the set 2 (Fig. 3b). For the sets 3 and 4, the wavelengths in the lateral (x-axis) direction were 60, 80, 48 mm, and 30, 40, 24 mm, respectively; for depth (z-axis) direction, 45, 36, 90 mm, and 22.5, 18, 45 mm, respectively.

In the session of the discrimination experiment, four test surfaces were placed in a virtual three-dimensional space as illustrated in Figure 4a. This scene was presented visually to the subject by a monocular 17' CRT screen. Each test surface was mapped by a single texture randomly selected from the same set. The mapped data

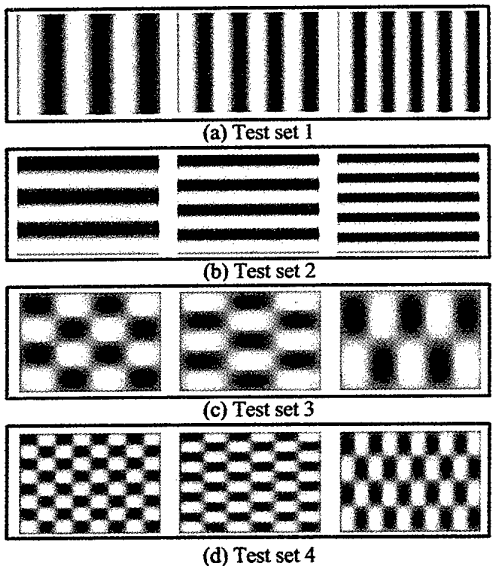


Fig. 3 Textures used in the discrimination test. Four sets were used individually in the session.

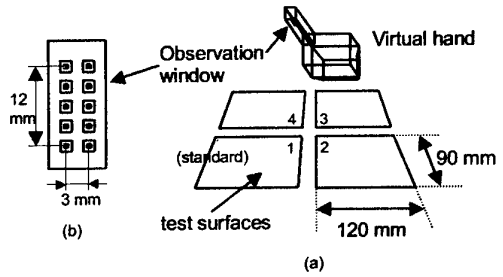


Fig. 4 Test surfaces in a virtual space (a), and pin layout of the virtual observation window at the fingertip (b).

was used only for haptic presentation; the surface was rendered in flat white on the screen.

The hand movement of a subject was measured by the FASTRAK (Polhemus Inc.) three-dimensional sensor. The intensity of pin vibration was determined according to two-dimensional position of the pin inside a test surface. Namely, when the tip of a pin intrudes under a test surface, the point projected orthogonally from the pin tip onto the test surface is located. Then the intensity of the point in the texture is calculated based on the sine function. The intensity data and the display command are transmitted to the device controller PC. The intensity data based on the hand position is updated at 30 Hz.

Two experienced subjects (ZJ, XH) and one inexperienced subject (MZ) performed the experiment putting on the F10 at the index finger and masking

Table 1 Correct answer ratio for texture discrimination

Subject	Set 1	Set 2	Set 3	Set 4
ZJ	100 %	100 %	100 %	100 %
XH	100 %	100 %	100 %	100 %
MZ	100 %	100 %	90 %	70 %

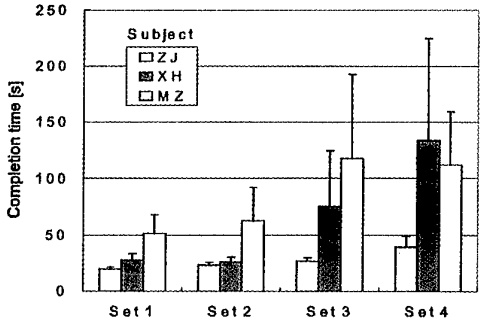


Fig. 5 Completion time for each texture set.

headphones. The subjects were asked to find whether the same texture(s) was on the test surfaces as a standard that was on the left-near surface. Ten judgements for the individual set were imposed to the subject.

Table 1 shows the correct answer ratio of the experiment. The subjects' answer were 100 percent correct except for the subject MZ who had little experience with the F10 display and missed the perfect discrimination for the sets 3 and 4. Two-dimensional discrimination requires an accurate voluntary trace motion and consequent pattern perception, which appears not necessarily easy for a novice user without doing some exercise.

Figure 5 shows the average completion time and SD for ten time trials. Tens of seconds were required inevitably to trace all of the four test surfaces; probably at least five seconds for each surface was necessary to capture the feature. No significant difference is observed between the set 1 and 2, however a remarkable increase of time occurred with sets 3 and 4 except for the subject ZJ. The pattern complexity normally added to the completion time, whereas it was observed only slightly with the subject ZJ since he was the primary system builder and had gained many experiences with the display output.

The interview with the subjects after the experiment collected the following observations. First, the trace movement on an unrestricted (without force feedback) plane did not evoke the parallel sense of exploration on a real physical surface. Since the finger penetrates the test surface, it was difficult to feel the exact position of the surface. Second, the bump shape of the texture which is normally perceived with a reference coordinate or a restricted motion was difficult to perceive with only a cutaneous sensation feedback. The recognition of a shape along the unclear trace path seemed to impose an increased perceptive load to the subjects.

4. Texture Display F10++

A force feedback device (PHANToM 151AG) was attached to the F10 display to provide it with force reflecting capability. Thus the Texture Display F10++ imparts haptic representations of both force and surface characteristics of a 3D object to the user's fingertip. The system in use is shown in Figure 6. The user holds a handle fixed to the F10 display to place his/her index

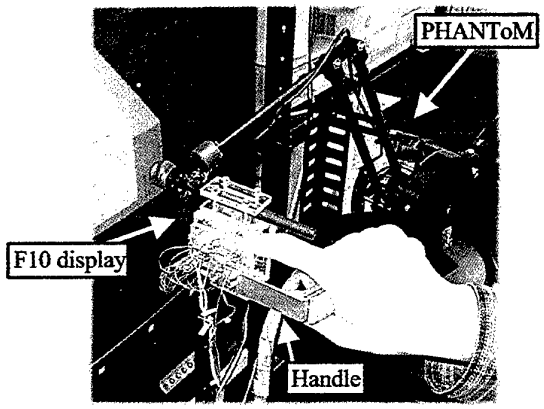


Fig. 6 Texture Display F10++. The F10 texture display is attached to the stylus of the PHANToM so that it can convey texture sensation of object's surface as well as touch reaction force from the virtual object to the user's finger.

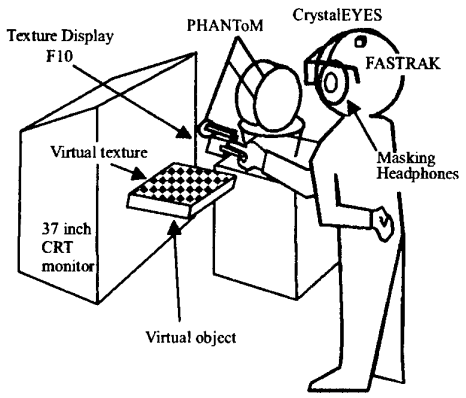


Fig. 7 Texture Display F10++ system setup. The F10 imparts texture information on a virtual object to the user along with force feedback provided by the PHANToM.

fingertip lightly on the pin array. Figure 7 shows the system setup. A virtual object with a texture on its surface is rendered three dimensionally within a workspace of the PHANToM carrying the F10 texture display.

This system is controlled by three PCs: the F10 controller, the PHANToM controller, and the rendering PC. The rendering PC calculates simulation loops that update both graphic and haptic information to be rendered. The rendering PC and F10 controller is connected by a serial communication line which enables data update at the F10 display at 76 Hz. The connection between the PHANToM controller and the rendering PC is established by a shared memory of 500 kilobyte/sec bandwidth. The position of the user's finger is reported from the PHANToM controller at 1 kHz, whereby the rendering PC updates texture information for the F10. The force feedback calculation is performed locally at the PHANToM controller that has a copy of object's data structure. Visual rendering at the rendering PC runs with a separated thread which depicts virtual objects at 18 Hz to the 37 inch CRT. (Stereo graphic images 800x600 dot are provided to each eye at 60 Hz through CrystalEYES PC.)

5. Evaluation of the F10++ system

5.1 Difference threshold of wavelength

The resolution of texture presentation was investigated by a psychophysical experiment. The differential threshold of wavelength was measured by using the constant method where five textures with different wavelengths were randomly presented to be compared with a standard stimulus. As the standard stimulus, a texture with a 1.2 mm interval, or wavelength, was used since it was around the minimum length as discussed later. Variable stimuli discriminated had wavelengths from 1.2, 1.6, 2.0, 2.4, and 2.8 mm. The standard stimulus and the variable stimulus were presented randomly on either the region A or B in Figure 8. The shape of wave used in the experiment was a clipped sinusoid indicated in Figure 9(a) where the intensity image and its cross section are depicted. At the peak of the intensity, the largest (level 15) stimulus was produced.

Five subjects (26 years old on average) performed the experiment. In order to control the condition, a velocity-index moving line was presented to indicate the trace velocity of 30 mm/sec. The subject mounted the Texture Display F10 to the right index finger, and traced on the both regions (standard/variable) following the velocity index. The both regions were painted in flat white with a separating central line and contour lines in black. One out of the five different wavelengths was randomly selected and presented paired with the standard.

The subject was asked to report within 60 seconds whether the pair had a same wavelength or not. Ten

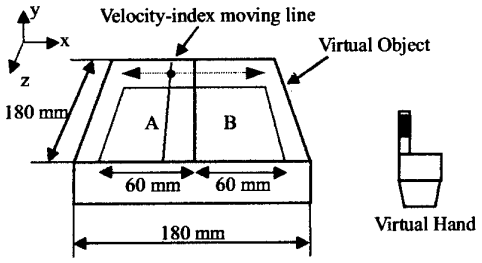


Fig. 8 Virtual surfaces provided for the discrimination of wave-lengths. Textures with different wavelengths were presented in the regions A and B 60 mm wide.

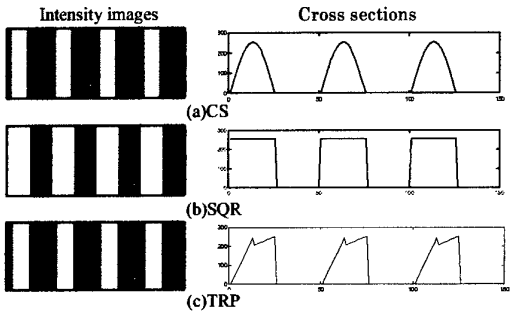


Fig. 9 Texture used in the experiment. (a) clipped sinusoidal, (b)square, and (c)trapezoidal wave forms.

trials form one session; each subject performed five sessions. As a reference, additional five sessions with no force feedback were performed as well. In this case, a repulsive force from the surface was not presented, whereas the weight of the F10 display and its handle was compensated to zero by adding a lifting force by the PHANToM. The force feedback limiting the finger from intruding into the object was added by 0.9 N/mm in proportion to the depth of intrusion at the center of the observation window, in the direction of a surface normal. A virtual hand was rendered with wire frames at the position shifted from the subject's own hand by about 100 mm to the screen. The orientation of the virtual hand was fixed to the z-axis (depth).

Figure 10 shows the upper difference threshold of the five subjects. The average among subjects was 0.48 mm in the case with force feedback, and 0.54 mm without force feedback. Regarding the sampling of the waveform at 76 Hz of the system update rate, the Nyquist wavelength is 0.79 mm when the subject's finger moves at 30 mm/sec. The standard wave length, 1.2 mm, is 1.52

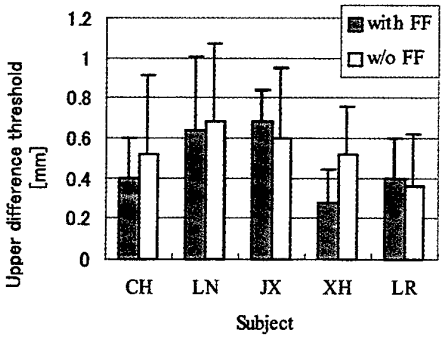


Fig. 10 Upper difference threshold (mean/std dev) calculated from the data by the summation method regarding the clipped sinusoidal waveform of 1.2 mm.

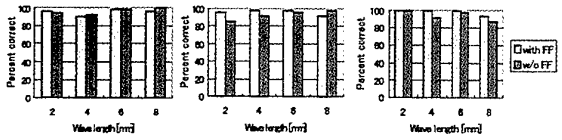


Fig. 11 Correct answer ratio for waveform discrimination.

times as long as the Nyquist wave length, and it produces a 25 Hz signal when it is traced at 30 mm/sec. If the wavelength of a variable stimulus is 1.68 mm, it produces a 17.9 Hz signal which is 7 Hz smaller than that of the standard. The result of the experiment shows that this difference was noticeable by 50 % rate on average.

The difference between subjects appears to be significant, although the difference between "with force feedback" and "without force feedback" is not significant. The reason the force feedback did not affect the difference threshold is considered to be the short and straight path required to complete this task. This means that fluctuation in the tracing trajectory did not act as a crucial hindrance to perception of the spatial frequency of the ridges.

5.2. Discrimination of waveforms

Discrimination of waveforms was investigated with respect to three waveform pairs: clipped sinusoid (CS)/square (SQR), CS/trapezoid (TRP), and SQR/TRP. The square and trapezoidal waveforms are depicted in Figure 9(b) and (c), respectively. The wavelength was varied from 8 mm to 2 mm with a 2 mm decrease. The same setup as the previous experiment was used except for the waveforms and the velocity-index line which was not indicated allowing the subject arbitrary comparison. Five subjects performed the experiment first without force feedback, then with force feedback. Paired identification test was used for analysis. The pair presented on the virtual object was randomly selected from CS/CS, CS/SQR, and SQR/SQR, and randomly placed on either of the regions in the case of CS/SQR discrimination. The duration before the decision whether the paired textures were identical or not was limited to 60 seconds. Ten decisions formed one session.

Figure 11 shows correct answer ratios averaged among subjects. No remarkable difference was observed over the three pairs, wavelengths, and force feedback modes. The overall average of correct answer ratio was 95.5 %. This figure indicates that the difference between three wave shapes was perceived clearly by the subjects. According to the interview with the subjects after these experiments, they could observe the difference even between the sensations occurred in tracing leftward and rightward in the case of the TRP waveform. Namely, the asymmetry of TRP's side inclinations was conveyed to the user's tactile sensation.

The force feedback restricting the finger on the object's surface provided an extremely natural feel of exploration as compared to the case lacks it. However, the correct answer ratio obtained here suggests that the force feedback did not work effectively in this experiment. Nevertheless, we believe there are reasons that helped the condition without a force feedback to achieve the correct discrimination. That is, the vibratory stimulation was presented regardless of the position as long as the finger penetrated under the surface. In addition, the trajectory of the subject's finger was stable because the weight of the F10 display was cancelled by the PHANToM; and the orientation angle of the virtual hand was fixed. Moreover, the patterns discriminated were simple for capturing. We consider that this good perception will not persist if the texture pattern does not exist on a flat pane and contains a more complicated variation.

6. Conclusion and future work

A three dimensional haptic texturing in a virtual space is a challenging issue since it requires both cutaneous and kinesthetic sensations being evoked. The Texture Display F10 permitted to produce a stimulus distribution on a fingerpad successfully, which is related only to cutaneous sensation. Although it allows the subject to discriminate patterns after he got accustomed to the device, the sense of feeling a surface was not natural without a constraint force. This mode of haptic stimulation would be more suited to the presentation of a

volume data which does not involve a rigid contact.

The subjective impression of a surface texture was greatly improved in the case of the F10++ display which presents both cutaneous and kinesthetic sensations. It was demonstrated that both the force feedback and the stimulus intensity distribution within a finger surface were crucial for three-dimensional haptic texturing. Although not discussed in the present study, the use of force perturbation in accordance with the texture profile will provide another control mode of interest on this display system. Further investigation of presentation accuracy with broader conditions would be involved in the course of clarifying the feature of this haptic texture display system.

References

1. Hollins, M., Faldowski, R., Rao, S., and Young, F., Perceptual dimensions of tactile surface texture: A multidimensional scaling analysis, Perception and Psychopysics,54(6), 697-705, (1993).
2. Minsky, M., Ouh-young, M., Steele, O., Brooks, F. P., Behensky, M., Feeling and Seeing: Issues in Force Display, Interactive 3-D Graphics, 235-243, (1990).
3. Ruspini, D. C., Kolarov, K., Khatib, O., The haptic display of complex graphical environments, Computer Graphics, (August, 1997), 345-352.
4. Srinivasan, M. A., Basdogan, C., Haptics in virtual environments: Taxonomy, research status, and challenges, Computers and Graphics, 21(4), (1997), 393-404.
5. Ikei, Y., Wakamatsu, K., and Fukuda, S. Vibratory Tactile Display of Image-based Textures. Computer Graphics and Applications, 17(6), 53-61, 1997

Interactive Two-Handed Terrain and Set Design in Immersive Environments

Falko Kuester, Bernd Hamann and Kenneth I. Joy

Center for Image Processing and Integrated Computing
Department of Computer Science
University of California, Davis, CA 95616-8562
{kuester, hamann, joy}@cs.ucdavis.edu

Abstract

This paper introduces *VScape*, a virtual environment for intuitive, hand-based terrain design. We present a design environment that provides an intuitive interface for the creation and manipulation of 3D scenes as required for terrain, game-level and set design. *VScape* was developed to provide the user with maximum design flexibility while providing a small, yet powerful set of easy-to-use tools and functions.

Keywords: Digital Design, Virtual Reality, Immersive Environments

1. Introduction

Virtual environments (VEs) are being used for industrial product design, analysis and verification tasks, medical imaging, architectural walkthroughs, geo-scientific exploration and sculpting. *VScape* combines data analysis, designs and verification capabilities of these environments and applies them to scene design suitable for urban planning, game-level design and set design.

Modeling environments traditionally rely on the use of polygonal, volumetric or mathematically defined primitives. Since primarily developed for the interactive design of terrains, *VScape* is currently surface-based and supports hand-based sculpting, painting and texturing on a polygonal level. For this type of application the goal was to "think visually" in terms of shapes, colors and textures instead of vertices, edges or curves and surfaces. As a consequence, the modeling concept is different from traditional keyboard and mouse centered computer based design and closer to traditional hands-on modeling. The user is equipped with a set of spatially tracked gloves and can employ a head-mounted display, immersive workbench or standard monitor-based stereo to create a 3D scene (Figure 1, 2). The core design criteria was to provide technical and non-technical users with an easy-to-use environment, for the creation of realistic environments. At the same time it was important to offer an unconstrained interface to the user that reduces or removes the pre-meditative design phase. This was accomplished by providing an environment that fosters the use of built-in verification tasks and the development of "game strategies" as part of the design cycle, resulting in a thoroughly developed and tested final product. Visibility, reachability and accessibility controls are built-in features that are automatically used throughout the design cycle. Relevant viewpoints or paths can be created as required and revisited throughout the design cycle for verification tasks.

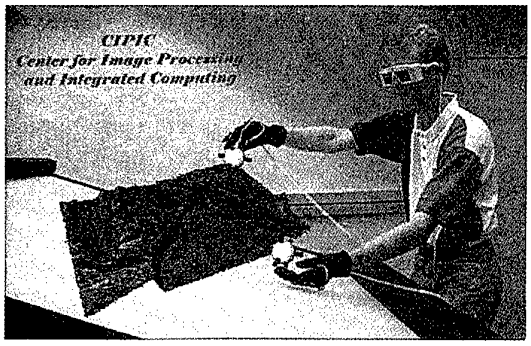


Figure 1: Terrain modeling and verification

2. Implementation

The observation that humans develop certain patterns on how to distribute tasks between their hands [14,15] has lead to the development of two-handed interfaces supporting this natural dexterity. Most of these interfaces are based on spatially tracked input devices, such as data gloves and pointers. *VScape* uses a set of spatially tracked pinch gloves, which can be used to navigate and manipulate the environment. This hand-based modeling approach provides access to efficient sculpting and painting metaphors that enable efficient and effortless expression of design ideas.

Furthermore, *VScape* is based on an object-oriented design approach, which treats every visual component within the VE as an object that can be freely positioned, manipulated, verified, analyzed and visualized. Once an object is created, its visual representation is added to a hierarchical scene graph. All visible objects contained in the scene graph can be selected and their properties visualized using a simple hand gesture. Special behavioral actions can be attached to any object and turn it into a tool for the manipulation of other objects. Any regular object within the scene graph can be directly accessed, scaled, translated, rotated, cloned and grouped. In order to allow intuitive object based modeling, the environment provides a basic set of controls, including:

- Object creation
- Object selection
- Object/scene manipulation
- Object/scene verification

In addition, accessibility controls for these operations are provided in the form of:

- Object/scene navigation
- Virtual menus

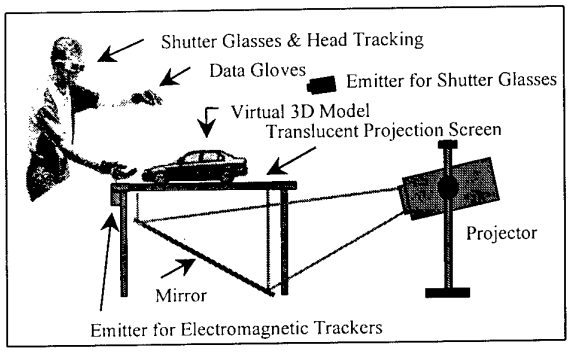


Figure 2: Hardware setup

Scene Navigation

Using head tracking, the user can study an entire model by simply moving his/her head or physically walking around the model. However, since the variety of possible application settings requires a less restrictive navigation paradigm, a two-handed interface is provided to freely translate, rotate and scale individual objects or the entire scene. In the object navigation mode the user can select an object through a particular one-handed gesture and then freely re-position and analyze it. If no particular object is selected while a navigation action occurs the system switches into scene-navigation mode and the action is applied to the entire scene, which by definition, is just another object composed of a group of objects. In this mode the user can use an imaginary rope to pull himself/herself through the scene using consecutive pinch-pull-release sequences. If both gloves are pinching at the same time, the imaginary segment between the pinching points is used as a five-degrees-of-freedom manipulator that allows to scale or rotate the scene or object. The relative position of the two points in regards to the original center point between the hands when the initial pinch event occurred determines the orientation of the scene and the distance between the two points defines its scale. This scheme supports a user-defined level of accuracy in which finer or coarser levels of precision can be defined by scaling the workspace. In other words, viewpoint movement supports an intuitive translation between working scales and provide direct access to different levels of modeling accuracy. This navigation scheme is intuitive and versatile, and new users are able to easily examine even complex scenes with minimal effort.

Virtual Menus

Menus are a vital component of all modeling systems since they provide access to the available system functions. With the transition from a 2D to a 3D environment, a new set of VR input devices and consequently new concepts must be implemented. Different solutions to this problem were proposed during recent years opting for either a direct port from the classical 2D menu to its 3D counterpart or new implementations designed specifically for 3D space [4]. Commonly observed problems are interference between the 3D menus and the scene and sub-menu access in highly cascading menus. We distinguish between gesture-based trigger and invocation events that allow the user to activate and select from various menus. A simple pinch gesture gives the user access to a base menu, which can be freely positioned in the VE. The menu is composed of 3D buttons assembled on a rectangular palette. All the sub-menus are opened within this original palette and can be traversed using simple hand gestures. Following our original design philosophy, all menus are implemented as objects that can be translated, rotated and scaled as desired. The menu items can apply associated functionality to other objects when

activated and be represented as text, a graphical presentation of the associated function or a combination thereof.

Terrain Creation
VScape provides a variety of mechanisms for the interactive creation and manipulation of terrain data. Terrain information can be either imported in polygonal form from a file or interactively created by using drawing primitives or freeform shapes and manipulated with a suite of virtual tools. Based on the desired terrain, game level or set, the design cycle can start at different levels from either a planar surface, artist sketch, a blueprint or even a satellite image mapped onto either a surface, 3D model or other types of geometry.

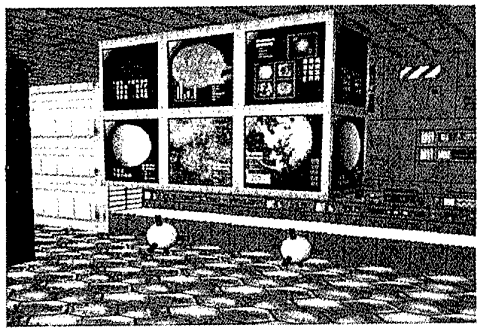
Object Creation
Arbitrary polygonal objects can be used to provide additional scene contents. Application-specific modeling libraries are supported and provide access to a wide range of primitives. These objects can be accurately positioned above the terrain using a rod level and then attached by simply dropping them onto the scene. User-definable objects and libraries including components such as houses and bridges are easily added through customizable menus, allowing the creation of additional scene contents. These objects can be accessed through the virtual toolbox and configured or extended to meet application specific demands. After invoking the virtual menu and selecting the appropriate object library, the chosen object can be simply grabbed and positioned on the terrain where desired. The application supports a "snapping mechanism", which enables accurate object placement onto the defined terrain. Objects automatically snap to the surface and can easily be cloned, moved, scaled, rotated or planted with a simple gesture. These libraries let the user design a scene, while immersion enables real-time verification. *VScape* reads and writes most of the standard file formats, including flt, wrl, 3ds and dxf among others.

Object Selection
This operation is the starting point for a variety of interaction tasks. The basic idea is to use a 3D input device to select the closest object to a spatial position. When a device-specific action is invoked in the form of a particular state event the absolute position of the tracker is mapped too world coordinates. The data gloves are visualized with virtual proxies. When the proxy intersects the bounding box of a particular object, the object is highlighted and ready for selection.

Object Manipulation
Once an object is selected it can be rotated, translated, scaled, cloned, re-shaped, grouped, deleted, or otherwise manipulated.

Terrain Manipulation
While constructing scenes it is important to observe specific boundary conditions such as construction time and cost. The available virtual tools discussed in the next section support user-definable design constraints, such as the amount of terrain movement per second.

Scene Verification
Design verification tasks such as visibility, reachability and accessibility, are frequently encountered during evaluation tasks. In our environment, they are built-in and are automatically



used throughout the design cycle. If required, relevant viewpoints can be stored and visited as desired.

3. Toolbox

The virtual toolbox merges the advantages of conventional physical tools and unconstrained virtual tools with the natural dexterity of a two-handed design environment. Instead of merely defining tools, we define actions and functionality, which can be associated with a set of geometrically defined modeling primitives provided as part of the toolbox or any object in the scene. Thus giving the user unlimited space for creativity and the means for the creation of new tools and design concepts. In our object-oriented framework, tools can be used to shape models, which subsequently can be turned into tools on their own. The virtual toolbox of *VScape* includes these types of tools:

- **Brushes** can apply color, material or texture to physical objects they come in touch with.
- **Filters** can be applied to an object or scene, and aid in smoothing, stitching or decimation tasks.
- **Guides** constrain the movement of an object and can be used in combination with any of the listed behaviors. Constraints could be movement in only a certain plane, around a certain axis, in a certain volume, etc.
- **Manipulators** allow high-precision positioning, rotation, and scaling of objects.

- **Magnets** can apply attractive or repulsive forces to objects. The "influence volume" of a magnet is determined by the scale of the environment. One can scale down the environment, with a simple hand movement, and the influence area gets smaller, or scale it up to increase it.
- **Paintbrush** paints color and applies texture to the surface. This tool can be used in brush-mode in combination with any other object in the scene.
- **Rulers** can be used to verify object dimensions or aid in the construction of objects.
- **Stamps** turn an object into a "3D printing stock" imprinting its information on another object.
- **Smoothers** turn an object into a "3D putty knife" or piece of sandpaper for surface smoothing and are primarily used for the removal of hard edges.
- **Snappers** aid in connecting objects within the scene.
- **Tesselators/Simplifiers** add detail at user-specifiable locations.

Magnets and stamps are very efficient tools for common modeling tasks, particularly when artistic creativity is emphasized.

4. Conclusions

VScape provides an intuitive environment for rapid prototyping of terrain, sets or game levels. It preserves the natural dexterity of physical modeling environments while providing the benefits of a digital design space. The current challenge lies in the development of more complex interaction and modeling schemes in the form of new virtual tools and input devices, using voice, gesture and pattern recognition. One of the most challenging tasks for the near future is to provide the necessary modeling precision required for engineering design tasks.

The object-oriented framework is easily extendable and provides a user-friendly prototyping environment. An enlarged feature set is currently under development. The generation of viewpoint-dependent adaptive meshes in real-time, subject to user-specified frame rates and/or error bounds, is targeted for performance reasons. Additionally, the rising number of programmable force-feedback devices and decreasing cost promises even more intuitive interaction potential. As for most new technologies, the initial investment of resources is substantial, but the rapid development of graphics hardware already shows good performance on high-end PC systems.

5. Acknowledgements

This work was supported by the National Science Foundation under contracts ACI 9624034 and ACI 9983641 (CAREER Awards), through the Large Scientific and Software Data Set Visualization (LSSDSV) program under contract ACI 9982251, and through the National Partnership for Advanced Computational Infrastructure (NPACI); the Office of naval Research under contract N00014-97-1-0222; the Army Research Office under contract ARO 36598-MA-RIP; the NASA Ames Research Center through an NRA award under contract NAG2-1216; the Lawrence Livermore National Laboratory under ASCI ASAP Level-2 Memorandum Agreement B347878 and under Memorandum Agreement B503159; the Lawrence Berkeley National Laboratory; the Los Alamos National Laboratory; and the North Atlantic Treaty Organization (NATO) under contract CRG.971628. We also acknowledge the support of ALSTOM Schilling Robotics, Chevron, General Atomics, Silicon Graphics, Inc. and ST Microelectronics, Inc. We thank the members of the Visualization Thrust at the Center for Image Processing and Integrated Computing (CIPIC) at the University of California, Davis.

6. References

1. Agrawala, M., Beers, A. C., Froehlich, B., Hanrahan, P., McDowall, I. And Bolas, M., The two-user responsive workbench: Support for collaboration through independent views of a shared space, in SIGGRAPH 97 Conference Proceedings, T. Whitted, ed., Annual Conference Series, pp. 327-332, ACM SIGGRAPH, Addison Wesley, Aug. 1997.
2. Beier, K. P., Virtual reality in automotive design and manufacturing. In: Convergence '94, International Congress on Transportation Electronics, Dearborn, Michigan, October 1994. SAE (Society of Automotive Engineers).
3. Cutler, L. D., Froehlich, B., Hanrahan, P., Two-handed direct manipulation on the responsive workbench, in Proceedings of the Symposium on Interactive 3D Graphics, pp. 107-114, ACM Press, (New York), Apr. 27-30 1997.
4. Deering, M. F., The HoloSketch VR sketching system, Communications of the ACM, 39(5):54-56, 1996.
5. Durlach I., Mavor, A.S., Committee on Virtual Reality Research, Commission on Behavioral Development, Social Science, Mathematics Education, Commission on Physical Sciences, and Applications, National Research Council. "Virtual Reality: Scientific and Technological Challenges," National Academy Press, 1994.
6. Ebert, D. S., Shaw, C. D., Zwa, A. and Starr, C., Two-handed interactive stereoscopic visualization, in: Proceedings of IEEE, R.Yagel and G. M. Nielson, eds., pp. 205-210, IEEE Press, Los Alamitos, Oct. 27-Nov. 1 1996.

7. Farin, Gerald E., Curves and Surfaces for Computer-Aided Geometric Design, 4th edition, Academic Press, San Diego, California, 1997
8. Froehlich, B., Fischer, M., Agrawala, M., Beers, A. and Hanrahan, P., Projects in VR: Collaborative production modeling and planning, IEEE Computer Graphics and Applications 17, pp. 13-15, July-Aug. 1997.
9. Froehlich, B., Barrass, S., Zehner, B., Plate, J. and Goebel, M., Exploring geo-scientific data in virtual environments, in: D. S. Ebert, M. Gross and B. Hamann, eds., Proceedings of IEEE Visualization '99, IEEE Computer Society Press, Los Alamitos, California, pp. 169--173, October 1999.
10. Galyean, T. A. and Hughes, J. F., Sculpting: An interactive volumentric modeling technique, Computer Graphics 25 (Proc. ACM SIGGRAPH '91), pp. 267-274, July 1991.
11. Green, M. and Halliday, S. , A geometric modeling and animation system for virtual reality, Communications of the ACM 39, pp. 46-53, May 1996.
12. Green, M., Shared virtual environments: The implications for tool builders, Computers and Graphics 20, pp. 185-189, Mar.-Apr. 1996.
13. Guiard, Y., Symmetric division of labor in human skilled bimanual action: The kinematic chain as a model, The Journal of Motor Behavior, 19(4):486-517, 1987.
14. Guiard, Y. and Ferrand, T., Asymmetry in bimanual skills, in: Manual asymmetries in motor performance, D. Elliott and E. A. Roy, eds., CRC Press, Boca Raton, FL., 1995.
15. Hui, K. C. and Ma, M. C., Deforming virtual objects with an instrumented glove, in: Proceedings of the Conference on Computer Graphics International 1998 (CGI-98), F.-E. Wolterand and N.M. Patrikalakis, eds., pp. 393-395, IEEE Computer Society Press, Los Alamitos, California, June 22-26 1998.
16. Krueger, W., Froehlich , B. Visualization blackboard: The responsive workbench, IEEE Computer Graphics and Applications 14(3): May 1994,12-15.
17. Kuester, F., Duchaineau, M. A., Hamann, B., Joy, K. I., and Uva, A. E. (1999), 3DIVS: 3-dimensional immersive virtual sculpting, in: Ebert, D. S. and Shaw, C. D., eds., Workshop on New Paradigms in Information Visualization and Manipulation (NPIV '99), ACM Press, New York, New York, Kansas City, Missouri, November 1999, pp. 132 –139.
18. Liverani, A., Kuester, F. and Hamann, B. (1999). Towards interactive finite element analysis of shell segments in virtual reality, in: Banissi, E., Khosrowshahi, F., Sarfraz, M., Tatham, E. and Ursyn, A., eds., in: Proceedings of 1999 IEEE International Conference on Information Visualization (IV'99) - Augmented and Virtual Reality Symposium, IEEE Computer Society Press, Los Alamitos, California, pp. 340-346.
19. Noble, R. A. and Clapworthy, G. J., Sculpting and animating in a desktop VR environment, in: Proceedings of the Conference on Computer Graphics International 1998 (CGI-98), F.-E. Wolterand and N. M. Patrikalakis, eds., pp. 187-197, IEEE Computer Society, Los Alamitos, California, June 22-26 1998.
20. Obeysekare, U., Williams, C., Durbin, J., Rosenblum, L., Rosenberg, R., Grinstein, F., Ramamurthi, F., Landsberg, A., and Sandberg, W., Virtual workbench: A non-immersive virtual environment for visualizing and interacting with 3D objects for scientific visualization, in: Proceedings of IEEE Visualization '96, R.Yagel and G. M. Nielson, eds., IEEE, Los Alamitos, Oct. 27-Nov. 1 1996, pp. 345-349.
21. Piegl, L. and Tiller, W., The NURBS Book, 2nd edition, Springer Verlag, New York, New York, 1996
22. Rosenblum, L., Durbin, J., Doyle and Tate, D., Projects in VR: Situational awareness using the responsive workbench. IEEE Computer Graphics and Applications 17(4):July/August 1997, pp.12-13
23. Sastry, L, Ashby, J. V., Boyd, D. R. S., Fowler, R. F., Greenough, C., Jones, J., Turner-Smith, E. A. and Weatherill, N. P., Virtual reality techniques for interactive grid repair, Numerical Grid Generation in Computational Field Simulations, Ed.
24. Shaw, C. D, Green, M., THRED: A two-handed design system, Multimedia System Journal 3(6), November 1995.
25. Shaw, C. D and Green, M., Two-handed polygonal surface design, in: Proceedings of ACM Symposium on User Interface Software and Technology, Two Hands and Three Dimensions, 1994, pp. 205-212.
26. Wong, J. P. Y., Lau, R. W. H. and Ma, L., Virtual 3D sculpturing with a parametric hand surface, in: Proceedings of Computer Graphics International 1998 (CGI-98), F.-E. Wolter and N. M. Patrikalakis, eds., IEEE Computer Society, Los Alamitos, California, June 22-26 1998, pp. 178-186.

ICAT 2000

Communication with VR Training System using Voice and Behavior

Kazuaki Tanaka, Tomoaki Ozaki and Norihiro Abe
Faculty of Computer Science and System Engineering,
Kyusyu Institute of Technology
Iizuka-shi, 820-8502 Japan
kazuaki@mse.kyutech.ac.jp
Hirokazu Taki
Faculty of System Engineering, Wakayama University, Japan

Abstract

In this research the training system using a virtual reality system was developed to instruct assembly/disassembly of mechanical parts to a user. A bidirectional interface system is realized that permits a user and the system to communicate each other using verbal and nonverbal information. When a user has questions in the process of operation, he can ask or give an order to the system that is an instructor using a spoken language and nonverbal behavior such as pointing action. A model of the instructor, an avatar is rendered in the virtual environment, he replies to questions or commands from a user. While an avatar uses a spoken language and can show instruction and operation of virtual parts with his behavior. Not only the synchronized recognition of voice and behavior of a user, but also the synchronization mechanism of the speech synthesis and the behavior generation of an avatar were stated clearly.

Keywords: Verbal/Non-verbal Communication, Training System, Assembly of Mechanical Parts

1. Introduction

We have reported several papers on the training system in mechanical assembly/ disassembly domain using a virtual reality system [1~3]. This training system is different from the traditional one using a mouse and a keyboard. It can watch the behavior of a user and instruct a right way when his action is wrong. But this system is not able to know the intention of the user definitely because no voice interaction facility is provided with the user. In other words, only by watching the human nonverbal behavior, the system can't completely detect the human intention or hesitation [2].

In communication between human beings, a spoken language becomes important besides the nonverbal behavior. So, in this research we propose a training system with verbal/nonverbal communication facility between human being and a computer system. In an assembly/disassembly training system, a user is permitted to get into a virtual environment in which a virtual machine is rendered and to perform a simulation of assembly/disassembly operation. When a user has questions in the process of operation, he can ask or give an order to the system which is an instructor in a spoken language. In this system, a model of the instructor, an avatar is rendered in the virtual environment, he replies to questions or commands from a user. While an avatar uses a spoken language and can show instruction and operation of virtual parts with his behavior.

In this system, both spoken language and nonverbal behavior can be input at the same time in order to realize verbal/ nonverbal communication. A virtual reality system can be brought closer to a real environment by using this interface. For example, we can ask a question or issue an order about an object to the system using a spoken language pointing at the object with a data glove. The system developed in this research permits an avatar to perform communication with a user using a spoken language information and a non-verbal information.

This time, as the field of the application of a bidirectional communication using verbal / non-verbal information, we selected the field of assembly / disassembly of mechanical parts. But we think this system applicable to various interface between human and machine

In this research, we proceed the research to attain a big aim to bring the communication between human and machine close to that between human beings.

2. System configuration

2.1. Hardware organization

The system consists of the computer which builds a virtual reality system, a microphone for a user to perform voice input, and 3-dimensional position sensors and data gloves for a user to input non-verbal behavior. General drawing is shown in Figure 1.

2.2. Continuous speech recognition parser (JULIAN)

This research used JULIAN which Prof. Doshita's research laboratory in Kyoto University developed as a speech recognition software. JULIAN is a recognition parser performing continuous speech recognition on the basis of a finite state grammar (DFA). It begins to look for the most plausible word list based on a given DFA for voice input from the microphone (continuous speech to make a pose with gap) and outputs it as a character string. DFA is made from vocabulary and the syntax rule that a user registered.

2.3. OpenInventor

To build a virtual reality system, three-dimensional surface models are used. A three-dimensional graphics library, OpenInventor [4] of SGI Company is used.

3. Assembly training system

3.1. System configuration

This system consists of 3 parts including avatar unit, spoken language processing unit, and non-verbal behavior analysis unit.

In a spoken language processing unit after voice input from a user is processed through a speech recognition and natural language processing sub-system, the result is given to an avatar unit.

In the nonverbal behavior analysis unit, hand position and attitude of the user are analyzed and the result is transmitted to the avatar unit. The avatar unit estimates the information sent and takes the factual knowledge of the virtual machine described in the system to make an appropriate response to a user.

We summarize the main facility of each part in Figure 2. We describe each function in detail later.

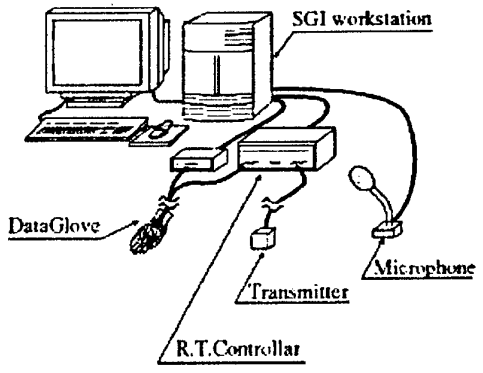


Figure1. Hardware organization

3.2. Verbal/ non-verbal interface

In this system, we used the interface that a user could input spoken language and non-verbal behavior simultaneously. Consequently, a user has only to utter toward a microphone in case issuing voice input without any keyboard action. The operation method peculiar to this interface is shown in the following.

i. With a traditional interface, the unique name must be used in order to distinguish the object from others. But when there are many same objects like mechanical parts, it is difficult to designate one of them using the name. This is, however, easily realized simply by pointing or grasping the object. A user can speak to the system by inputting the spoken language such as "Install this part on that." while pointing at the two objects with a data glove. A user will need not memorize the identifier of object parts by admitting the use of the directive. How to make correspondence between terms meaning instruction such as 'this', 'that' and the behavior like a pointing action will be fully described in 6.2.

ii. A user is able to order the avatar to do assembly operation. If we should want to interrupt the operation while the avatar is executing an assembly operation, we could have the avatar suspend the operation by issuing a phrase or sentence that means the suspension of the operation.

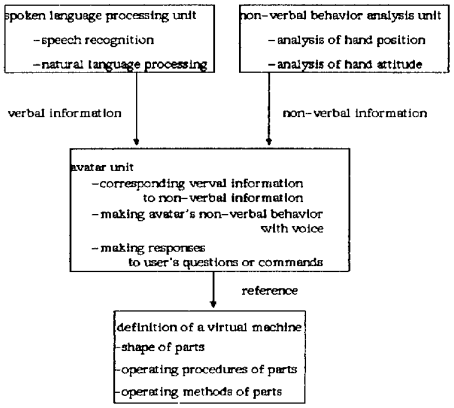


Figure 2. System configuration

3.3. Definitions of operating procedure

In this system, the operating procedure (AND/OR procedure) is defined with an AND/OR graph as shown in Figure 3. Hereafter, we call the part a mvobject which has a component to be moved after a user has selected it with a data glove or voice input, and call the partner part a basic part into which the mvobject is installed. Each node in the AND/OR graph shown in Figure 3, for examples START, END and points from 1 to 8, expresses an assembly status of the give assembly.

Assembly operation along an arc of the graph (operation) is necessary in order to change the state of the assembly. In operation, operating instruction and the object parts (mvobject, basic part) are described.

All nodes of the AND/OR procedure shown in Figure 3 consists of OR nodes. In other words, a user has only to sequentially follow the graph from the upper part toward the lower part. For example, assembly procedures such as [START - 1-5 - END], [START - 3-8 - END] are right procedures.

3.4. Definitions of mechanical part

In this system mechanical parts are defined with a Scene Graph [4] as shown in Figure 4. A **MyParts** is data node. A part name and a part number are described in the **MyParts**. A part name corresponds to the voice input from a user. A part number is used for describing the object part in the AND/OR procedure.

3.5. Assembly method

A basic assembly method of virtual part should be explained here. To make clear an assembly method and to help the operation of a virtual part, an arrow is attached to each portion of the part to be mated with another part as shown in the Figure 5. The direction coincides with that of the assembling operation. If the following conditions are met, then the operation is automatically finished at the final state; A user is moving each part to the direction of an arrow. The roots of arrows attached to the parts get closer each other.

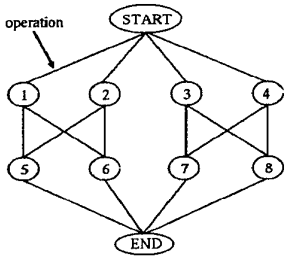
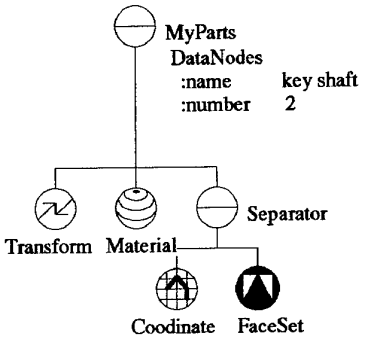


Figure 3. Operating procedure based on AND/OR graph



MyParts : definition of a part name and a part number
Transform : definition of a part position
Coordinate, FaceSet : definition of a part shape

Figure 4. Definition of mechanical part

And in this system, the operating procedure (AND/OR procedure) is defined with an AND/OR graph [5].

4. Non-verbal behavior analysis unit

4.1. Selection of parts with data glove

In this system, the analysis of spoken language and that of behavior are performed in parallel. This makes it possible for a user to specify the mechanical part to be manipulated or selected by pointing action, or to grasp and move it by hand using spoken language.

At present, the analysis of user's behavior is limited to only the hand movement. Using three-dimensional position sensor added to the user's writs, the quantity of the translation and rotation are measured from the wrist. The attitude of the hand is detected using a data glove.

The system permits the user to specify an object by pointing with a forefinger. The state of the hand is judged referring to the values of joint angles.

When a data glove is pointing at some objects as shown in the Figure 6, the system judges that the parts are to be selected that are included in a cone emanating from the finger-tip and that are intersecting the conic beam

When the palm of the data glove is going to be closed to grasp the part as shown in the Figure 7, if a user has no objects in the data glove and if the bounding box of the data glove and the bounding box of the parts interfere each other, the system decides that the user has grasped the part and changes the color again.

4.2. The analysis of the hand movement using a three-dimensional position sensor

When an object is selected using a data glove, the user may move his/her hand with the forefinger pointing out. Of

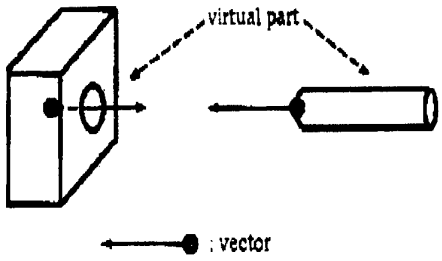


Figure5. A virtual part given an arrow

course a forefinger may be bent when the user intends to point at nothing. In the former case, the cone emanating from the forefinger may interfere in several objects during the hand displacement. It is, however, difficult to find the object that the user aimed at from the interfered objects.

As a result of analyzing the behavior of a man, when the man points at an object with a forefinger, the hand generally stops with the forefinger pointing at the object for a while.

The movement of the hand measured with a three-dimensional position sensor when a man is going to point at an object is shown in the Figure 8.

It is understood that the pointing action corresponds to the portion (a) in the Figure, and that the data from the three-dimensional position sensor are comparatively stable for the moment. So on finding that the movement of the hand stops, the procedure mentioned in 4.1 is made active.

5. Spoken language processing unit

5.1. Natural language processing

A spoken language input from a user (Japanese) is converted into a character string by JULIAN. Next, a natural language processing program will analyze the character

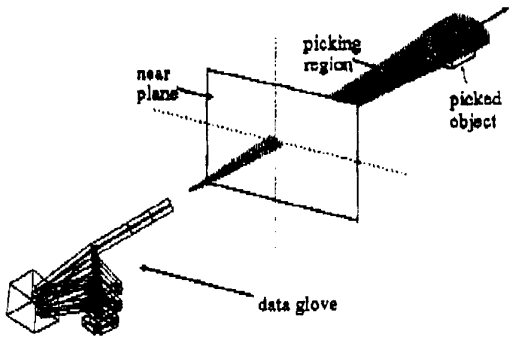


Figure 6. Selection of part by pointing action

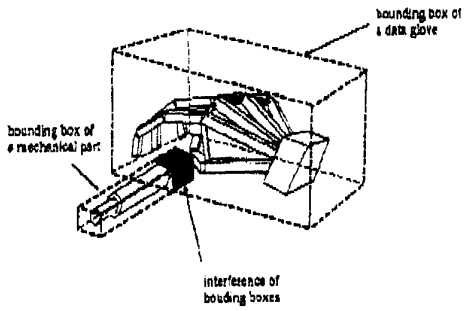


Figure 7. Selection of parts by grasping operation

string through the speech recognition and the semantics of the voice is extracted. A user must register into JULLIAN the words and syntax rules used in the speech recognition as described in 2.2. The dictionary made at that time is also available to the language processing.

We show an example of the word dictionary and syntax dictionary in Figure 9 and Figure 10.

The syntax rule is registered assuming the categories registered in the dictionary to be non-terminal symbols.

Semantic analysis is done in top-down fashion. When a sentence "Assemble the worm shaft.(ウォームシャフトを組み立てる。)" is input, the input sentence is matched to the syntax "OBJ WO OPV_A AUX_A" in the syntax dictionary, and a category shown in the Figure 11 is obtained.

Because a category is registered corresponding to a function of a word, semantics of the word becomes possible.

At this time whether the content of the sentence can be handled with the system or not is judged. To increase a number of sentences to be understood, you have only to add words belonging to a category or categories and syntax rules. Inversion expression and more than one expression can be also accepted. The second rule in syntax rules shown in the Figure 10 is the inversion form of the first syntax rule. The flow of the process is shown in Figure 11.

5.2. Constructing the contents of dialog

An analysis result provided with the natural language processing exploits the knowledge of assembly, and is stored in a list called a contents list. When "Assemble the worm shaft.(ウォームシャフトを組み立てる。)" is a recognized character string, a contents list as shown in the Figure 12 is made.

Key words corresponding to the contents of the sentence

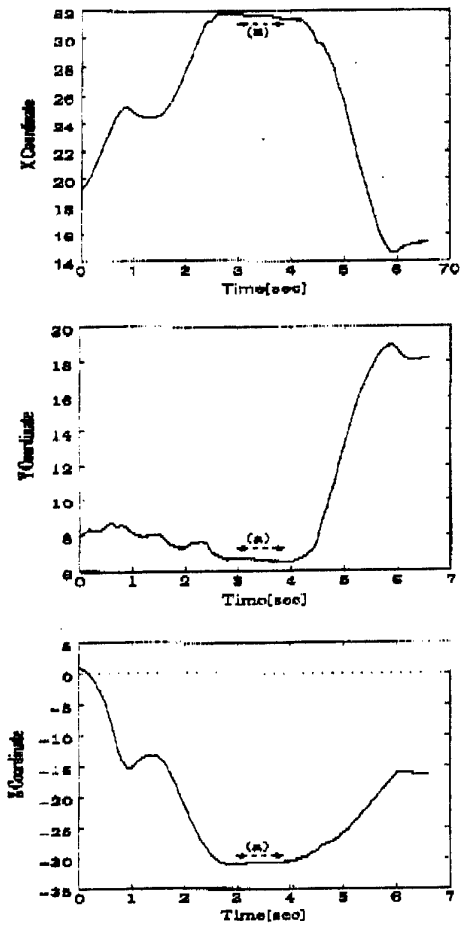


Figure 8. Analysis of the hand movement using a 3-dimensional position sensor

are stored in the first line of the contents list. The system distinguishes the contents of the utterance with the key words. If some assembly operation is necessary, a method realizing the operation is put in the second line, and the object is entered in the 3rd and the 4th line. Because two parts are mainly selected as objects of one operation, the 3rd and the 4th line are prepared. Voice information from a user is transmitted to the avatar unit in the form of the content list.

5.3. Flow of process

We describe the flow of process of spoken language in the following. At first, a user issues an inquiry or command to the system using a spoken language. Next, natural language processing analyzes the spoken language, and if the system is able to accept the contents, a contents list is made. Otherwise, the user must repeat the voice input.

The content list of the conversation is estimated after being communicated to the avatar unit.

```
##An object of operation → Semantics category
%OBJ → ory
name
グリース          The word to recognized
ウォームシャフト
  :

##Method of operation (～ます)
%OPV_A
組み立て
取り付け
  :

## The end of a verb (imperative)
%AUX_A
ろ
なさい

##Particle (を)
%WO
を
```

Figure 9. A part of dictionary

```
OBJ      WO      OPV_A
OPV_A    AUX_A   OBJ
```

Figure 10. A part of syntax dictionary

6. Instructor (avatar) unit

In this chapter, we explain the process performed in an instructor unit. The nonverbal information and the spoken language information from a user are respectively processed in the nonverbal behavior processing unit and the spoken language processing unit and their results are communicated to an instructor unit.

An instructor unit evaluates the information and makes the appropriate response to a user based on the factual knowledge of virtual parts described in a system.

On the instruction of parts operation, it is important to have a user operate a mechanical part with a data glove, but we believe that he will understand how to manipulate the part if he sees someone operating the part. So in this system, we prepare the following mode for responding to a question or a command from a user. In the mode, an avatar shows a user how to operate a virtual part with his hands explaining the operation in a spoken language.

This chapter explains the process of behavior generation of an avatar in detail.

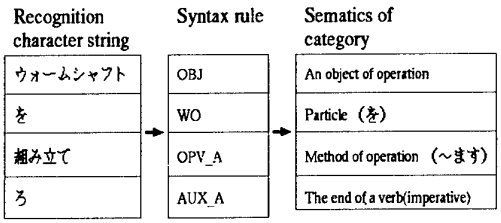


Figure 11. A result of language processing

Order	Operation
Operation	assemble
Object1	worm shaft
Object2	Nothing

Figure 12. Contents list

6.1. Dialog engine

A dialog engine shown below is installed into the instructor unit to make response for a user.

When an operation command is given from a user, the system matches the content list (5.2) to the operation prescribed in the AND/OR graph (3.3), then a response is made. If the operation command fits the operation in the AND/OR graph, the operation and explanation are performed by an avatar. When the instruction is wrong, a warning is issued. On the contrary, an avatar refers to the AND/OR graph to generate a command or question and the current state of the world. As the system (an avatar) knows the current state of the world, it is able to generate both the relevant command and erroneous command referring to the AND/OR graphs. In almost cases, a relevant command will be entrusted to a user.

When he cannot show a correct answer, an avatar will show the answer to him by moving parts. If he hesitates without starting action, an avatar asks if a user understand what he should operate. At first, he is asked to tell two-part names and to indicate them. If he cannot respond, the avatar will show the answer in place of him. If he is going to grasp a wrong part, the avatar will show the right one. If he is going to move the part toward wrong direction, it is prohibit and the avatar will show the correct movement by his hand in the same way as the case a user takes a leadership.

Voice output is generated with the ViaVoice of IBM, but it simply translates a sentence generated with the answer generating routine into Japanese emotional voice output.

6.2. Correspondence between verbal information and a nonverbal one

We have already described that in this system we prepared a mode to designate mechanical parts by combining a directive with a pointing action.

Correspondence between a directive and a pointing action is necessary for a system to understand the contents which the directive shows. This process is performed in the avatar unit in which a verbal information and nonverbal one are collected from a user.

For example, when "install the worm shaft here" is input, the object or the place corresponding to the phrase "the worm shaft" and the word "here" are found from the pointing actions, respectively. In this case, even if there were several parts corresponding to "the worm shaft", the one belonging to the class of a worm shaft is put into the candidate set. Nonetheless, when two or more candidates are left, the one with the size or the structural characteristic making the operation specified possible is selected. Nevertheless, if a unique object cannot be determined, the system must ask a question to the user to make clear an object to be selected.

Note here that there is a problem. The analysis of user's action is taken place in real time because it is not measured with a vision system but with magnetic sensors. On the other hand, as the analysis of utterance is prolonged until it will terminate, it is hard for the system to know the word uttered as soon as corresponding action is analyzed. When actions that are recognized as pointing action are observed several times and several demonstrative pronouns or the definite names appear in the corresponding utterance, to find the correspondence between the actions and pronouns/nouns is difficult. If the numbers of their appearance are equal, then they correspond in the order of appearance. At first, the system solved the problem based on the order of appearance, but at present the correspondence is solved based on the time of appearance.

6.3. Behavior generation of an avatar

An avatar has three joints in his arm and 14 joints in his hand same as shown in the Figure 13.

The behavior of an avatar is decided by assigning respective values to the position and rotation of each joint. The values given to the position and rotation of each joint are constrained as the attitude of an avatar cannot deviate from the human attitude.

In this system, the avatar is permitted to do nonverbal behavior such as grasping and pointing action in a virtual environment. As it is difficulty to compute all values of position and rotation of 14 joints of his hand in case of both behavior, they are acquired with the motion capture

method using a data glove beforehand.

Next we explain how to determine the attitude of his arm. Here presume that the position of his shoulder is fixed in the pointing or grasping action. Then everything to be done is to determine the values of the position and rotation of remaining wrist, elbow and shoulder.

In case of the pointing behavior, the center of gravity of the part an avatar is going to point is first retrieved. Next, the values of the position and rotation of a wrist are determined to enable him to point to this centroid position. Values of the position and rotation of the remaining elbow and shoulder can be obtained by the inverse kinematics.

As how to grasp depends on the shape of the object to be grasped, it is difficult to decide values of the position and rotation of a wrist by computation. Consequently relative position between the part to be grasped and a wrist must be registered beforehand. Values of the position and rotation of the remaining elbow and shoulder can be obtained in the same way as in the pointing action.

A series of attitudes of an avatar from the initial state to the end state can be got by a linear interpolation of both situations. When an avatar can neither point to nor grasp a part from his current position, he has to moves to the new position that makes him do the behavior.

Figures 14 and 15 show the situation an avatar performed the installation operation of the worm shaft.

6.4. Synchronization of behavior and spoken language by avatar

Problem is the synchronization of voice and behavior. When an avatar mates a part A and B, the way of operation must be explained with voice while he is performing the operation.

As an example, consider the following case in which an avatar shows a sequence of operations saying that you should insert this part A into the hole of the part B. After first the avatar extends his arm to an object and grasps it, he will move it to an approach point of an operation. At that occasion, it is assumed that the reference of the part name is finished at the same time as he grasps the part.

When a necessary preparation is successfully done, the avatar explains how to mate them while operating them from approach points. Here, it is assumed that the approach points are set at the positions shown in the Figure 14. After all, the following will be his behavior. He will grasp a part uttering a phrase like "the part A", he will move it to an approach point afterwards. The similar operation is performed on the part B. Note that he will move a part to an approach point saying nothing. He will say that "in this way a part A is inserted into B" operating the part. When an operation consists of several secondary operations, each secondary operation and a corresponding explanation are synchronized in the similar method as that mentioned above.

The ViaVoice of IBM is used to have an avatar speak the content of explanation. The software gives us the time needed to utter a phrase consisting of N characters. Of course the time needed for the utterance is also controllable, and the start and the completion time of an utterance are also controllable.

On the other hand the time needed to draw each frame of an avatar's behavior is depended on the complexity of background (Context) in drawing and a power of a computer used.

Assume here that it takes T to utter a series of phrases and that a graphic generation of M (M=M1+M2) frames must be finished at the end of the utterance. M1 is a frame number to be drawn by directly before the time he grasps the object and it is determined in the following way. M2 means images from the grasping point to the approach point. Interpolation of images are performed using the inverse kinematics so that his hand moves smoothly along a line connecting an initial point and an end point. Let assume the background or context does not change suddenly while an avatar utters a given phrase.

And if the time needed for drawing an initial frame is t, then the time necessary for drawing of M frames becomes tM. (Of course this graphic generation is invisible to users).

In the case of T>tM1, give the sleep time(T/t-M1) after every frame generation. In the case of T<tM1, start the speech synthesis at the time(tM1-T) after graphic generation started. As the utterance is finished almost simultaneously at the time he grasps an object, the system has

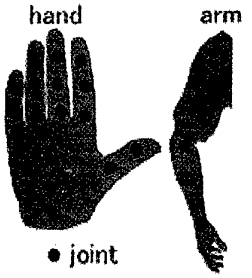


Figure13. Joint of hand and arm

Figure14. Grasp of virtual parts

Figure15. Installation of a worm

only to drawn the remaining M2 image in the same rate as M1 afterwards. And the speech synthesizer is notified of the completion of a graphic generation.

In this way, by dividing one operation into two steps, a grasp and operation itself, voice and a graphic generation are synchronized. There are some cases that an operator does not require another part to be operated. In such cases the object of the operation is considered to be just one.

7. CONCLUSION

In this research the training system, which instructs assembly / disassembly of mechanical parts to a user was developed. A bi-directional interface system is realized that permits a user and the system to communicate each other using verbal and nonverbal information.

Not only the synchronized recognition of voice and behavior of a user, but also the synchronization mechanism of the speech synthesis and the behavior generation of an avatar were stated clearly.

You may feel a few differences between the impression from the avatar and the sense received from a human being.

When an avatar utters a same word again and again, the tone should be changed. For example, even if a user repeated the same mistake, an avatar just utters the same warning in the same tone. Functions that make his tone and expression more strictly are necessary in order to give many better effects to a user.

The current system cannot prohibit any erroneous operation of a user physically. By replacing the hand of a user with PHANToM which is the haptic interface and restricting the movement of PHANToM, operation errors can be prohibited.

It is evaluated with various situations, and technique proposed with a research now is improved.

We will continue our effort aiming at the construction of more natural man machine interface by introducing new frames and evaluating them in various situations.

References

[1] Norihiro Abe and Saburo Tsuji "A consulting system which detects and undoes erroneous operations by novices" Proc. of SPIE, pp.352358, (10 1986)

[2] Norihiro Abe, Tomohiro Amano, Kazuaki Tanaka, J.Y.Zheng, Shoujie He, and Hirokazu Taki "A Training System for Detecting Novice's Erroneous Operation in Repairing Virtual Machines" International Conference on Virtual Reality and Tele-Existence(ICAT), pp.224229,(1997)

[3] Norihiro Abe, J.Y.Zheng, Kazuaki Tanaka and Hirokazu Taki "A training System using Virtual Machines for Teaching Assembling/ Disassembling Operations to Novices" International Conference on System, Man and Cybernetics,pp.20962101 (1996)

[4] J, Wernecke "The Inventor Mentor", Addison Wesley Publishing Company (1994)

[5] Tomoaki Ozaki, Kazuaki Tnaka, Norihiro Abe, Hirokazu Taki"Verbal/Nonverbal communication in Virtual Environment", International Conference on Systems, Man, and Cybernetics, 1999,to appear

Development of R-Cubed Manipulation Language
The design of an RCML 2.0 system

Dairoku Sekiguchi, Wei-Chung Teng, Yasuyuki Yanagida, Naoki Kawakami, and Susumu Tachi
School of Engineering, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, JAPAN
{dairoku, waldo, yanagida, kawakami, tachi}@star.t.u-tokyo.ac.jp

Abstract

The concept of R-Cubed (Real-time Remote Robotics: R^3) aims to provide a way to telexist anywhere in the world by controlling remote robots over the network. RCML (R-Cubed Manipulation Language) is considered to be a language for describing the interface for controlling remote robots in an R-Cubed concept. RCML 1.0 is an extension of VRML97 and uses a PROTO node, which is an extension node of VRML97. Through the experimental implementation of an RCML 1.0 system, two design problems were revealed. One is a limitation on implementation, and the other is a separation of user interface and control information definition. To overcome these problems, we designed a new version of the RCML system. This paper proposes a new design of the RCML system called RCML 2.0. In RCML 2.0, we introduced a language RXID 2.0 for defining Graphical User Interface (GUI), which is used for controlling the remote robot into the system. Both RCML 2.0 and RXID 2.0 are XML- based languages. By using XML, expandability and flexibility in implementation are introduced to the RCML system. RXID 2.0 has mechanism for a one-way link to an RCML data structure, and this mechanism provides for the complete separation of the control of the robot and the user interface. We also show the reference implementation of the RCML 2.0 system.

Key words: R-Cubed, RCML, RCTP, RXID, XML

1. Introduction

R-Cubed (Real-time Remote Robotics: R^3)[1] is a concept that enables a user to telexist anywhere in the world with a sensation of actually being there. This is accomplished by controlling remote robots over the network. Users of an R-Cubed system feel and act as if they really existed in a remote environment, regardless of the physical limitations of time and space [2].

RCML (R-Cubed Manipulation Language) is considered to be a bottom-up approach of the R-Cubed concept. The design of an RCML system utilizes existing infrastructures and devices such as the Internet and PC and, users of the system will be able to use it easily and intuitively. In a manner similar to the way a VRML browser provides a standard method for accessing the virtual world, we intend to provide a standard method for accessing the remote real environment with an RCML system.

2. Related Work

Recently, network robotics is active research area. Many implementation methods have been examined. The simplest implementation is the combination of CGI and HTML [3]. A CGI and HTML based system generates a new web page whenever a user requests a command to a robot. Hence this implementation does not allow a user to control a robot continuously and is not suitable for a system such as RCML that requires continuous control of a remote robot.

An implementation that uses web browser and Java applet [4] is widely used method [5][6][7]. By using Java applet, a user can control a remote robot without installing any special software and, continuous control of a remote robot is achieved at the same time. However it is difficult to build a system such as a high end RCML system that requires real-time processing, because Java has limitation on its performance.

To become more general and sophisticated method for controlling a remote robot, an approach that uses an ORB (Object Request Broker), such as CORBA [8] and DCOM [9], has been also examined. Hirukawa et al. [10] use CORBA to implement their teleoperation system. ORiN [11] that is developed by JARA (Japan Robot Association) [12] uses DCOM. These ORB are mechanism for handling distributed objects and do not define interfaces between each object. Hence it is necessary to define a standard method (API) that can adapt to various robots, but it is very difficult to define such general interface in advance of actual system implementation. Until now several implementations that use an ORB have been proposed, but a standard method for controlling a remote robot is not established yet.

3. Previous Implementation

As previously stated, our goal is to provide a standard method to access the real world. Our first step toward this goal was to design the first RCML system called RCML 1.0 in 1997 [13]. RCML 1.0 consists of RCML 1.0 and RCTP/1.0. RCML 1.0 is a description language for controlling a remote robot, and RCTP/1.0 is an HTTP/1.1-based protocol for transferring control data. The design of RCML 1.0 is based on VRML97 [14]. By adding a method to describe the real world to VRML97, we aimed to merge access to the real world and access to the virtual world seamlessly. To maintain upper-compatibility with VRML97, RCML 1.0 uses a PROTO node, which is an extension node of VRML97. By retaining upper-compatibility with VRML97, we can make good use of the existing VRML browser to develop a client program (RCML browser). Furthermore, users who only have a VRML browser will still be able to access an RCML 1.0 file and browse the virtual worlds.

We also developed an experimental implementation to examine and verify the design of the RCML 1.0 system and demonstrated the remote control of the omnidirectional mobile robot [15][16]. Compared to approaches using CGI and HTTP, our system has a short response time that enables us to control the remote robot in a continuous operation and not in a one-by-one command-based operation. In addition, by combining a VRML view, the seamless integration of the two access methods to the virtual and real worlds and intuitive operation were achieved.

However, some design problems within the RCML 1.0 system were revealed at the same time.

The first problem is limitation on implementation. Because the design of RCML 1.0 is an extension of VRML97, it is most efficient to implement the client side program by extending the existing VRML browser. Hence, the development of a client program will always be restricted by limitations of the VRML browser. For instance, there is no choice other than Java to extend the VRML browser, and Java is not a very suitable development environment for a system such as RCML that requires real-time processing.

The second problem is the necessity for the separation of the user interface and the control information definitions. In RCML 1.0, user interface definition such as choice of input GUI and the control information definition such as definition of value are mixed and described in one file. Since control information is specific to each robot, once it is written, it will not be modified so frequently. However, user interface is sometimes modified more frequently than control information. Because various configurations of user interface can be considered, two or more user interfaces may be prepared for one robot. In fact, our RCML 1.0 experimental system has several user interfaces, and each user interface has a different type of input interface, such as a scroll bar and a button. When several user interfaces are prepared for one robot in the RCML 1.0 design, the same control information will exist simultaneously in each file. Such a situation is inefficient and difficult for file management. Hence, it is important to separate user interface definition and control information.

4. The Design of the RCML 2.0 System

We tried to make the design of the new RCML 2.0 system as simple as possible. To simplify the system, a target robot is described as a set of variables that are necessary for controlling a robot, and the control of the target robot is considered to be equivalent to accessing variables. The following figure shows a simple example by two degrees of freedom with a pan/tilt camera.

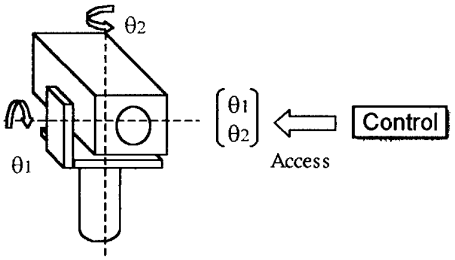


Fig. 1 An example by pan/tilt camera

In the example above, there are two variables that correspond to each pan/tilt axis, and the camera is controlled by accessing these two variables. At this point, we show a very simple example that has only two variables. When we have to handle more variables, it is better to manage variables in a tree structure than in a flat structure. Thus, the RCML 2.0 system manages variables in a tree structure, and this tree structure is called an RCML data structure.

In the RCML 2.0 system, RCML 2.0 is a language for describing an RCML data structure. As described in the previous section, RCML 1.0 inherits not only the advantage of VRML 97 but also the disadvantage of VRML 97. Hence, we decided to base the design of the RCML 2.0 on Extensible Markup Language (XML) [17]. By using XML, the following advantages are introduced into RCML:

- Expandability
- Clear syntax
- Flexibility in implementation

In the RCML 2.0 system, RCTP/2.0 defines a method for accessing the RCML data structure via a network.

Upon defining the specification of RCTP/2.0, we considered the following things:

- Ease of implementation
- Expandable design
- Providing the mechanism for real-time control by minimizing the overhead of a data stream

We used the syntax and the sequence of the well-known HTTP/1.1, instead of creating a new protocol entirely, thus making it more understandable. Moreover, because RCTP/2.0 is based on HTTP/1.1, the user can learn it easily, and expandable design is also satisfied because it uses the expansion mechanism of HTTP/1.1. In order to minimize the overhead, we designed a special format for the data stream. RCTP/2.0 also has a mechanism for real-time control by synchronizing time between a server and a client.

We introduced a language for defining GUI, which is used for controlling the remote robot into the system. This language is called RXID (RCML Extensible Interface Definition) 2.0. RXID 2.0 supports well-known common GUI elements such as window, scroll bar, button, and text input and can define property for each element, such as position, size, and caption. Hence, a user can easily design various kinds of user interfaces for controlling the remote robot. RXID 2.0 is also an XML-based language and has mechanism for a one-way link to an RCML data structure, which is illustrated in the following section.

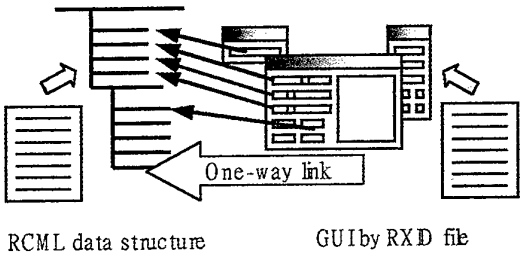


Fig. 2 One-way link in RXID 2.0

This one-way link defines the relationship between a GUI element described in an RXID file and a variable in an RCML data structure defined by an RCML file. By linking these two elements, the input from the GUI side is transferred to an RCML data structure, and the change of variables in the RCML data structure is transferred to the GUI side. Thus, a user can control remote robots by GUI and know the status of the remote robot. Because RXID 2.0's one-way link starts from the RXID file side, it is not necessary to modify the RCML file when describing the RXID file. This provides for the complete separation of the control of the robot and user interface. Hence, multiple user interfaces for one RCML file (Fig. 3) can be defined. Or, one integrated user interface for multiple RCML files (Fig. 4) can be defined.

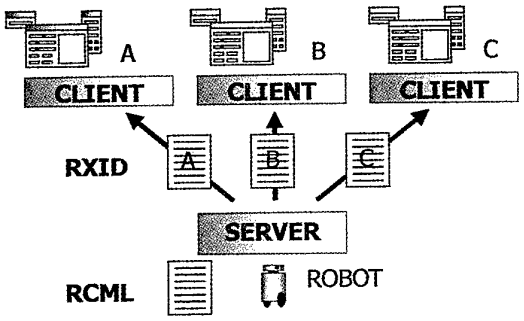


Fig. 3 Multiple interfaces for one robot

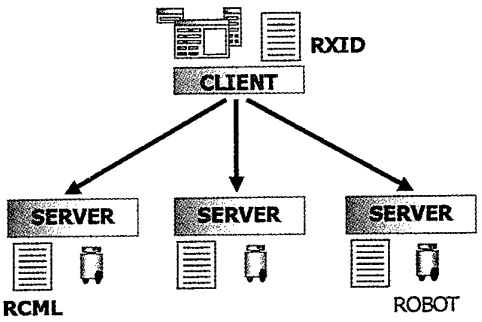


Fig. 4 One user interface for multiple robots

5. Outline of the RCML 2.0 system

The next diagram shows an outline of the RCML 2.0 system.

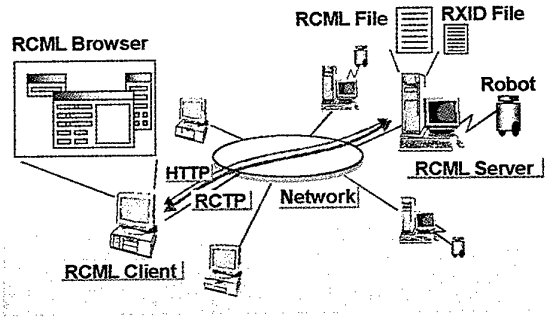


Fig. 5 An outline of the RCML 2.0 system

The RCML 2.0 system consists of an RCML server and an RCML client. A robot is connected to an RCML server. Each RCML server has an RCML 2.0 file, which contains the information of the robot connected to the server and an RXID 2.0 file, which defines the user interface for controlling the robot. An RCML client program specially designed for controlling a remote robot by a human operator is called an RCML browser. An RCML browser downloads the RCML 2.0 file and the RXID 2.0 file by using a standard protocol such as HTTP. An RCML browser then displays a GUI

panel based on the RXID 2.0 file and connects to the server using RCTP/2.0 based on the information described in the RCML 2.0 file. Once an RCTP/2.0 connection is established, a user can freely control the remote robot with the RCML browser.

6. RCML 2.0

The specification of RCML 2.0 is very simple. RCML 2.0 has only six nodes as follows:

Table 1. Elements of RCML 2.0

Elements	Explanation
<rcml>	The root element of RCML. This element is used to describe the information about an RCML site.
<group>	This element declares a group of data.
<access>	This element declares a method to access the <data> node.
<data>	This element declares the <data> node in an RCML data structure.
<link>	This element declares a link for its parent element.
<meta>	This element declares a metadata for its parent element.

In the above list, four elements from <rcml> to <data> elements are used to describe the RCML data structure. The <link> and <meta> elements are elements for describing additional information (metadata) for a data node.

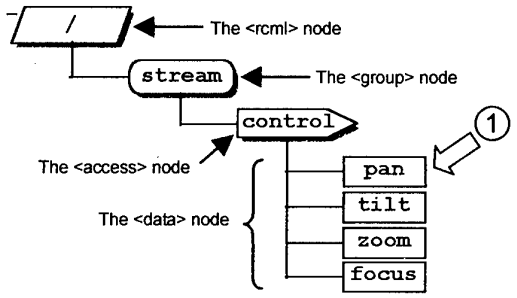


Fig. 6 RCML data structure by sample RCML (Listed in Appendix A)

In an RCML data structure, to indicate a specific node path expression that can be commonly seen at file system is used. For instance, the path to the <data> node located at (1) in Fig. 6 is described as follows:

```
/stream/control/pan
```

In an RCML data structure, the name of a node must satisfy the following rules:

- The same rule that is defined as "Name" in an XML syntax applies to a node name.
- Nodes in the same level must have different names.
- The order of nodes does not have a specific meaning, unlike an XML document.

7. RCTP/2.0

RCML 2.0 only defines interface for controlling remote robots. Hence, to make an actual system, some sort of communication method is required. RCTP/2.0 is used as a communication method in the system. RCTP/2.0 defines the protocol for reading and writing data that are described by RCML 2.0. RCTP/2.0 has the following functions:

- Access for data - read and write
- Controls of access privilege

7.1 Access methods in RCTP/2.0

RCTP/2.0 has some data access methods. In RCML 2.0, these access methods can be specified for each <data> node. Each access method is briefly described in the next section.

7.1.1 Normal access

When no access type is specified in an RCML file, the normal access method is used. The normal access method uses connection-oriented connection. An access occurs to each <data> node. This is the simplest access method.

7. 1.2 Event-type access

When an event-type access method is specified in an RCML file, this access method is used. The same as a normal access method, an event-type access method uses connection-oriented connection. The difference from the normal method is simultaneous access for set of data and the occurrence of a Fdata change event" from a server. It places the importance of the assurance of changing variables between a server and a client. So, an event-type access is suitable to set the parameter for the robots or to send a sequence of commands.

7.1.3 Stream-type access

When a stream-type access methods are specified in an RCML file, this access method is used. Different from other methods, a stream-type access uses a connection-less data stream. By sending data as a stream, a stream-type access can change data continuously. To send the newest data without delay, a lost packet is not sent again in a stream-type access. A stream-type access attaches more importance to real-time access of data than event-type access. So, when a bandwidth of network is very broad and time delay is short, it is very useful.

7.2 Connections of RCTP/2.0

RCTP/2.0 defines two types of connections: control and data stream.

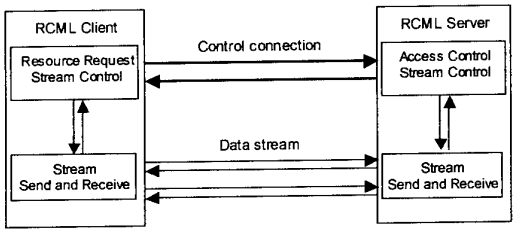


Fig. 7 Connections of RCTP/2.0

Control connection mainly obtains access control and controls data stream. Control connection uses a connection-oriented method. A client establishes a control connection to a server. A session is a period starting when the client establishes a control connection and ending with disconnection. Normal access and event-type access use control connection.

On the other hand, data stream continuously transfers the control data that is needed to control remote robots. Data stream is used to transfer control data for the robot that requires real-time control. Thus, it uses a connection-less method that does not handle the re-transmission of packets. A stream-type access uses this data stream.

7.2.1 Control connection

As in HTTP/1.1, control connection is a protocol based on a request and response pair. The structure of the message is also the same, where a start-line includes a method and an RCTP version in a request, and it includes a status code and a Reason-Phrase in a response. A message header follows the status line and the message body comes in last. RCTP/2.0 defines the following 10 methods:

Table 2. Methods of RCTP/2.0

Method name	Explanation	C→S	C←S
CONNECT	Starts an RCTP session	O	×
ACQUIRE	Acquires an access permission	O	×
RELEASE	Releases an access permission that was obtained	O	×
READ	Obtains the value of <data> node	O	×
WRITE	Sets the value of <data> node	O	O
SETUP	Sets the parameters for access method	O	×
GO	Instructs the beginning of access	O	×
PAUSE	Instructs the pause of access	O	×
STOP	Instructs the end of access	O	O
BYE	Ends the session	O	O

RCTP/2.0 allows a server to issue a request on a client in the method WRITE, STOP. and BYE, which is quite different from HTTP/1.1. In the above table, C→S represents the request from a client to a server while C←S represents the request from a server to a client.

7.2.2 Data stream

Data stream uses a connection-less method that uses packets to communicate. HTTP/1.1 does not have data stream connection. Data stream is used for stream-type access. To ensure real-time communication, it does not re-transmit data when packets are lost. A data stream packet can include several "payloads," which ar Rpayl minimum units of data transmission. By making several payloads that are generated at the same time into one packet, it is possible to decrease the number of packets in a data stream. A payload also has a field that shows the type of information it contains. Thus, it is possible to overlap several types of information in one data stream. When transmitting real data in a data stream, a binary format is used as in READ and WRITE methods in control connection.

In addition to data-stream payload for real data transmission, RCTP/2.0 also defines flow-control payload. The protocol for flow control is very simple: a request for operation and the acknowledgement of the request and the negative acknowledgement. Operation provides heartbeat operation for synchronizing local time and reading and setting operation of flow-control parameters. As flow-control parameters, RCTP/2.0 defines the transmission interval of payloads and the timeout value of receiving payloads.

7.3 Two aspects of RCTP/2.0

The control of a data stream and the management of the right to control the robot must take place at the same time. Because, when controlling remote robots, to give permission for sending and receiving a data stream for a client is equivalent to giving the right to control the robot to the client. Thus, it is inefficient and complicated to implement when they are managed by different protocols. Therefore, RCTP/2.0 has two aspects: management of server resources and transmission and control of a data stream.

8. RXID 2.0

RXID 2.0 defines the following elements:

Table 3. Elements of RXID 2.0

Name	Explanation
<rxid>	The root element of RXID.
<window>	This element creates a window. A window can be used as a placeholder for all other RXID widgets.
<session >	This element declares an RCTP session.
<access>	This element declares an access method to data nodes in an RCML data structure.

widget elements	This kind of element creates user interface elements (RXID widgets).

The <window> element is always the child element of the <rxid> root element. One <window> element corresponds to one window displayed by the RCML browser. Attributes of the <window> element represent the property of a window such as position, size, title, and background image. Each <window> element must have at least one <session> element to specify the URL of a target RCML file. The <access> element can be used to declare the access method to the specific node in an RCML file. The <window> element also has widget elements as child elements. Widget elements are used to place various user interface elements (RXID widgets) inside the window. The current version of an RCML browser supports the following widget elements:

Table 4. Currently supported widget elements

Name	Explanation	R	W
<box>	This element creates a box.	×	×
<label>	This element creates a label. A label is used to display static text.	×	×
<text>	This element creates a text. A text is used to show values that can be updated in real time.	O	×
<button>	This element creates a button.	×	O
<checkbox>	This element creates a checkbox.	O	O
<radioGroup>	This element creates a group of radio buttons.	O	O
<scroll>	This element creates a scroll bar.	O	O
<slider>	This element creates a slider.	O	O
<edit>	This element creates an edit box.	O	O
<popUpMenu>	This element creates a pop-up menu.	O	O
<netmeeting>	This element creates a live video viewer component (NetMeeting).	×	×
<html>	This element creates an html viewer component.	×	×
<actionButton>	This element creates an action button.	×	×

In the list above, 'R' indicates that the widget can read data from an RCML data structure. For instance, the <slider> element reads a current position of the slider knob from an RCML data structure and updates the position of the slider knob. On the other hand, 'W' indicates that the widget can write data to an RCML data structure. The element, which supports 'write' action, such as a button, checkbox, and scroll, can write the change of value inputted from a user to an RCML data structure. There are also elements that support neither read nor write action. Boxes and labels, for example, represent static widgets and are not related to an RCML data structure.

The widget that can do read or write action has a "dataPath" attribute to declare a one-way link to an RCML data structure. Here is brief example of a scroll bar:

```
<scroll dataPath="/stream/control/pan" ... />
```

The scroll bar above is linked to the node specified by "/stream/control/pan" in the RCML data structure (Fig. 6 (1)).

9. Reference System

We also implemented an actual system based on the design of RCML 2.0. The main purpose of this system is to show the reference implementation of the RCML 2.0 system. Hence, we tried to fullfill the specifications of an RCML 2.0 system as much as possible, and we also tried to keep the system simple and easy to understand and to extend.

The target platform of our system is Windows (Windows 98, NT 4.0, 2000) and Unix (FreeBSD, LINUX, etc.). Currently, the RCML client supports Windows platforms only. The main development language is C++, and "XML for C++ (Version 2.3.1)" [18] is used as an XML processor.

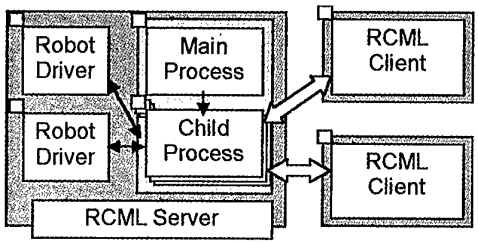


Fig. 8 Processes in RCML 2.0 system

The RCML server consists of the main process, the child processes, which handle each session to an RCML client, and the robot driver processes, which handle each robot. The RCML client is one independent application and connects to the desired RCML server by typing URL as would be done in an ordinal web browser.

The following image is the screen-shot of the RCML client:

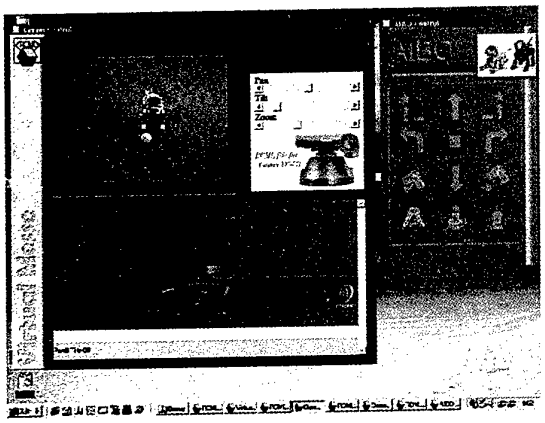


Fig. 9 The screen-shot of the RCML client

10. Conclusion

In this paper, we showed a new design for an RCML system (RCML 2.0) [19]. By using XML in the system design, the new design provides expandability and flexibility to the RCML system. In RCML 2.0, a language RXID 2.0, which is used for defining user interface, is introduced. By introducing RXID 2.0 into the system, complete separation of the control of the robot and user interface is achieved. We also developed the reference implementation of RCML 2.0 system. Our reference implementation fulfills almost all the specifications defined by the specifications of the RCML 2.0 system.

References

1. MITI of Japan, R-Cubed WG ed.: "R-Cubed", *Nikkan Kogyo Shinbun*, (1996).
2. S. Tachi: "Real-time Remote Robotics – Toward Networked Telexistence", *IEEE Computer Graphics and Applications*, pp. 6-9, (1998).
3. R. Simmons: "Xavier: An Autonomous Mobile Robot on The Web", *Preprints IROS'98 Workshop 'Robots on the Web'*, pp. 43-47, (1998).
4. http://java.sun.com
5. M. R. Stein: "Painting on the World Wide Web: The PumaPaint Project", *Preprints IROS'98 Workshop 'Robots on the Web'*, pp. 37-42, (1998).
6. Roland Siegwart, et al.: "Guiding Mobile Robots through the Web", *Preprints IROS'98 Workshop 'Robots on the Web'*, pp. 1-6, (1998).
7. P. Saucy, F. Mondada: "KhepOnTheWeb: One Year of Access to a Mobile Robot on the Internet", *Preprints IROS'98 Workshop 'Robots on the Web'*, pp. 23-29, (1998).
8. http://www.omg.org
9. http://www.microsoft.com
10. H. Hirukawa and I. Hara: "The Web Top Robotics", *Preprints IROS'98 Workshop 'Robots on the Web'*, pp. 49-54, (1998).
11. Mizukawa,M., Matsuka,H., Koyama,T., Matsumoto,A.: & Dfacto standard API for Open and Networked Industrial Robots", *Proc. 30th Int. Symp. on Robotics*, pp.455-462, Oct. 1999
12. http://www.jade.dti.ne.jp/~jara/
13. Y. Yanagida, N. Kawakami, S. Tachi: "Development of R-Cubed Manipulation Language - Access Real Worlds Over the Network", *Proc. of the 7th International Conference on Artificial Reality and Tele-existence*, pp159-167, 1997
14. http://www.web3d.org
15. W. C. Teng, A. Nukuzuma, N. Kawakami, Y. Yanagida, S. Tachi: "Development of R-Cubed Manipulation Language -The specification of RCML and RCTP-", *Proc. of the 8th International Conference on Artificial Reality and Tele-existence*, pp152-162, 1998
16. W. C. Teng, D. Sekiguchi, A. Nukuzuma, N. Kawakami, Y. Yanagida, S. Tachi: "Development of R-Cubed Manipulation Language –Implementation and Evaluation of RCML System-", *Proc. of the 9th International Conference on Artificial Reality and Tele-existence*, pp79-83, 1999
17. http://www.w3c.org
18. http://alphaworks.ibm.com/
19. http://www.rcml.org

Appendix A: RCML 2.0 Sample

```
<?xml version="1.0"
encoding="Shift_JIS"?>
<!DOCTYPE rcml SYSTEM "rcml.dtd">

<!-- RCML Version 2.0 sample -->

<rcml
    version="2.0"
    site="rctp://rrr.rcml.org"
    timeSource="GPS"
    timePrecision="1E-4"
    title="RCML sample"
    author="D.Sekiguchi"
    info="RCML test site."
    contact="mailto:dairoku@rcml.org"
>

  <group name="stream" permission="rw">
    <access name="control"
        type="stream"readInterval="16e-
3"
        writeInterval="16e-3"
        readTimeout="10"
        writeTimeout="10">
```

```
      <data name="pan" type="int"/>
      <data name="tilt" type="int"/>
      <data name="zoom" type="int"/>
      <data name="focus" type="int"/>
    </access>
  </group>

  <link kind="UserInterface"
    href="sample.rxid">

</rcml>
```

ICAT 2000

Hair Shape Modeling from Video Captured Images and CT Data

Ali, Md. Haider and Toyohisa Kaneko
Toyohashi University of Technology
1-1, Hibarigaoka, Tempaku-Cho, Toyohashi City 441-8580, Japan
kaneko or haider@mmip.tutics.tut.ac.jp

Abstract

We propose a three-dimensional (3D) reconstruction method of human hair-shape from rotating head multiple video captured images and CT data. It is well known that no hair is present on the polygonal skin surface of the human head (*3D-head*) reconstructed from CT or MRI data. Our task is to reconstruct and add the hair-shape on the 3D-head to create a realistic human head model for simulating post-surgical facial expressions. Using a sculpturing technique based upon rotating head images we propose a method of reconstructing the hair-shape with the help of 3D-head. We have utilized binarized voxel data of the 3D-head (*solid-head*) in this regard. The sculpturing object in our definition is the solid-head surrounded by assumed thick hair-voxels. We sculpture the surrounding hair-voxels according to the extracted hair-region from the video captured images while keeping the internal solid-head intact. We reconstruct the concave and semi-occluded regions by digging up to the visible skin surface of the solid-head in/near the hair region. We define *complete-head* as the 3D polygonal surface obtained from solid-head including the residue-sculptured hair-voxels on it. Experimentally we have shown that our method can successfully reconstruct the concave and semi-occluded regions in the skin-hair junction regions, which is not easy to reconstruct by the conventional way.

Key words: realistic modeling, 3D head modeling, 3D hair modeling, visualization.

1. Introduction

The 3D reconstruction techniques build real world into computational models, which is urgently required in virtual reality, CAD/CAM, and other related fields[1]. In medicine, 3D modeling is very essential especially in the field of computer-integrated surgery (e.g., surgery simulation and/or image guided surgery). Our task is to reconstruct a realistic complete head model for simulating post-surgical expressive faces. We have already proposed methods for making a realistic model face by precisely pasting blended colors from three photographs to the 3D facial skin surface derived from CT data[2,3]. In those proposals we emphasized on the 3D-2D projective registration and also our interest was limited to the facial part only.

Absence of hair actually fails to provide a complete head model, especially in the simulated post-surgical expressive faces. Without hair, a human face does not look realistic and moreover is not sufficient to create a vision-convincing animation. It is known that no hair is present on the 3D-head reconstructed from CT or MRI data. Even a Cyberware Digitizer™ scanner cannot reconstruct hair well (which is usually black). That is to say, there is no commercially available instrument to reconstruct the hair shape. In this paper we propose a method of reconstructing and adding 3D hair-shape on the CT/MRI reconstructed 3D-head to make a realistic complete-model.

In this paper we use four different names of 3D data of human head. The definitions are given here for further clarification.

- **3D-Head:** The hairless polygonal-data (skin surface) of the patient's head derived by the marching cubes algorithm from the original 12 bit gray level CT slices using a skin-air threshold.
- **Solid-Head:** Binarized voxel-data of the 3D-head. This can be obtained by any of the two following ways: (1)by filling the 3D head, or (2)from the same set of CT slices using identical threshold value as used in 3D-face reconstruction. In the former case there is a possibility to yield shape-error in the multiple layer regions, e.g., ears and nostril as it is not like a simple polygon filling. The latter way on the other hand easy to implement and there is no possibility of yielding shape-error.
- **Complete-Head:** The polygonal-data obtained from the solid-head covered with residue-sculptured hair-voxels by the marching cubes algorithm.
- **Final-Head:** This is actually the complete-head but to get better surface quality, the uncovered skin surface portion is replaced by the 3D-head surface

1.1 Literature Review

There are a number of works reported on reconstructing 3D shape from a sequence of 2D views and/or silhouettes[1,4,5,6,7,8]. In almost all cases the target is to reconstruct 3D shape correctly, especially the concave or un-exposed parts. L. Zhau and W. Gu[1] used a laser range sensor in conjunction with a sequence of images in this regard. J. Zheng and F. Kishino[4] proposed a technique of detecting un-exposed regions while reconstructing 3D shape from sequential image silhouettes. They employed a filter for detecting non-smooth points in the silhouette distribution. S. Sugimoto and M. Okutomi[5] proposed a technique of estimating radii of rotating points on the object surface using spacio-temporal images. To determine the missing radius data they fitted the obtained data with a suitable sine curve. There are also some proposals of reconstructing a 3D face by modifying generic facial geometry according to the photographs[10,11].

1.2 Our Method

Our method of reconstructing 3D hair-shape is simple but different from the works mentioned in the above. Insertion of solid-head into the sculpturing object and selection of hair-region silhouettes instead of complete head from the 2D image is the distinction from all other related works. Keeping the solid-head intact while sculpturing surrounded voxels to reconstruct the concave and semi-occluded regions in the resulting complete-head is a new way of 3D reconstruction.

1.3 Paper Organization

The remainder of this paper is organized as follows: Section 2 gives the outline of the proposed reconstruction algorithm. Section 3 describes the basic requirements for making a correct hair-shape. The sculpturing procedure is given in section 4. Experiments in section 5 and the conclusion and future work plan are in section 6. Finally the acknowledgement is in section 7.

2 Outline of the Reconstruction Algorithm

Fig.1 shows a flow-chart of the proposed reconstruction algorithm. A brief description of Fig.1 is given below.

The CT data provides two basic input data: (1)3D-head (polygonal-data), and (2)solid-head (voxel-data). The solid-head is the main part of the sculpturing-object. The sculpturing object is a 3D rectangular box filled with assumed hair voxels and the solid-head is placed at the center of the box. The shape and orientation of the 3D-head and the solid-head are almost identical. 3D-head can be said to be more accurate since it has subvoxel accuracy. The 3D-head provides necessary information in the form of 3D-edge to determine the camera parameters for each video captured image. Each video captured image also provides two input data for the reconstruction: (1)2D-edge, which is required for 3D-2D registration to determine the camera parameters of the video captured image, and (2)hair-region i.e., the extracted 2D hair-shape. Edge based registration in Fig.1 helps to determine the camera parameters of the video captured images by matching the projected 3D-edge with the corresponding 2D-edge.

At the hair-sculpturing stage, for each video captured image the result of edge based registration helps to position the virtual camera, which focuses towards the sculpturing-object. The technique of 2D hair-shape

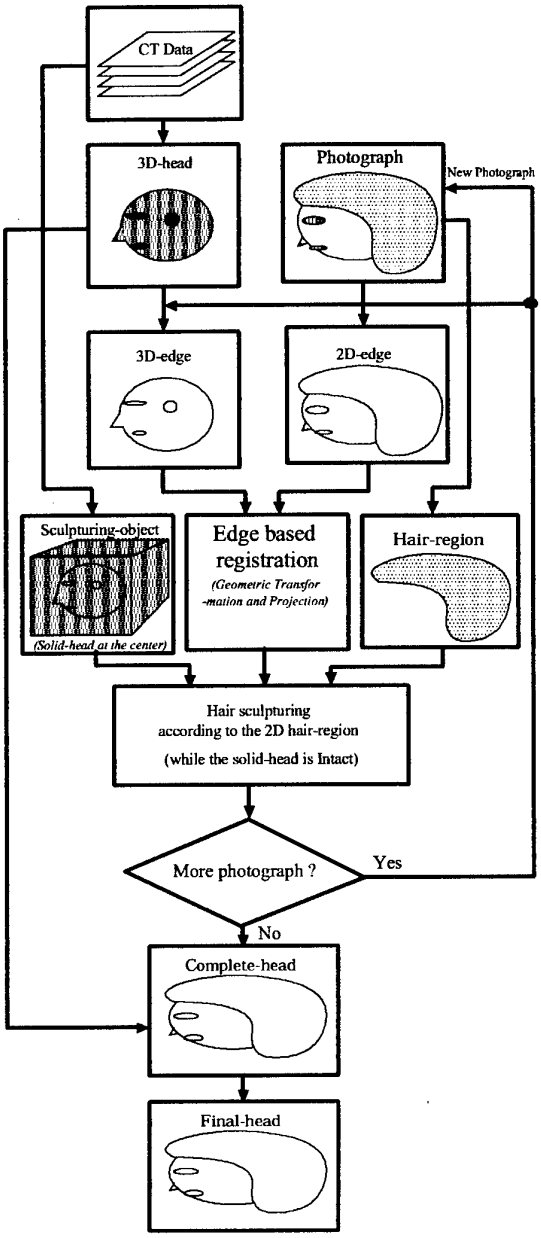


Fig.1 Flow-chart of the reconstruction method

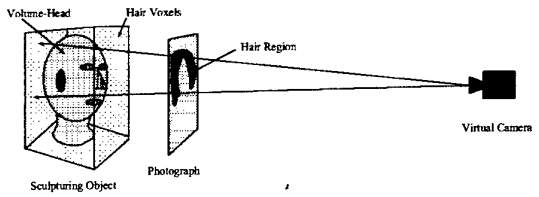


Fig.2 The technique of selecting hair-voxels in the sculpturing object.

re-projection on the sculpturing-object is shown in Fig.2.

Fig.3 The 2D virtual camera image of the solid-head (the white part and the gray region visible under hair-region). The superimposed black portion is the extracted hair-region from the original video captured image. The hair-voxels in the sculpturing-object has been divided in different groups according to this virtual camera image.

All the hair-voxels outside the re-projected ray-lines of the hair-region are cutout or removed. The hair-voxels, which are outside the hair-region but on the ray-lines to the solid-head (white region in Fig.3), are removed up to the solid-head surface. Fig.4 shows the sculpturing stage as a trans-axial view. The removed hair-voxels on the ray-lines to the uncovered solid-head actually leads to create concave and semi-occluded regions.

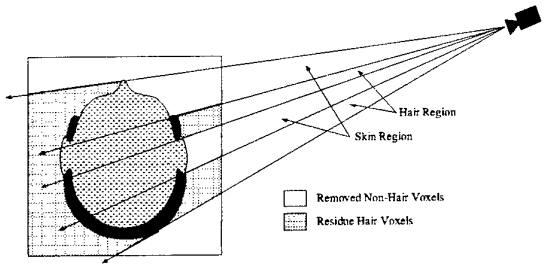


Fig. 4 Sculpturing technique from the top

The resulting hair-shape on the solid-head can be obtained by repeating the whole procedure for the rest of the video captured images. The complete-model is the polygonal surface derived from the solid-head including sculptured residue hair-voxels on the solid head, and the final-head is obtained by replacing the uncovered skin with that of the 3D-head.

3 Basic Requirements for Reconstruction

It should be noted that in this paper we emphasis on reconstructing the hair region as accurately as possible instead of the entire head. In our definition the target head (i.e., complete-head) is the combination of the hair-shape and the 3D-head (which is already available). To perform this task it needs two basic things: (1) to know the position and direction of the camera for each image, and (2) to extract the hair region correctly from each video captured image.

3.1 Camera Position Estimation

For each video captured image, we estimate the camera position by determining seven unknown parameters (six transformations and a projection function) of the virtual camera. The virtual camera is modeled as a simple pinhole camera. In the experiment we register a video captured image with the computer-generated image of the 3D-head in order to determine the virtual camera parameters. We perform the registration task automatically by our already reported *edge featured based 3D-2D projective registration technique*[3]. To obtain fast registration for the in-between images (i.e., images taken from the positions more than ten degree (10^0) far from the front, left or right), we assume an initial angle of rotation based upon the total number of in-between images and the angular span between left-to-front or front-to-right images. The rest of the camera parameters from the current image are assumed as the initial value for the next image.

3.2 Hair Region Segmentation

A semi-automatic tool called *intelligent scissors*[9] segments the hair-region from each video captured image. Fully automatic segmentation of 2D image is an unsolved problem, while intelligent scissors allow hair region to be extracted quickly and correctly using simple gesture motion with a mouse. When the mouse pointer comes in proximity to an edge, a dynamic programming based live-wire boundary wraps around the region. Finally the hair boundary is extracted by the using our already reported filling algorithm[3]. Fig.4 shows the hair-boundary and extracted hair-region, respectively, for a video captured image.

(a) Original photograph. (b) Hair-boundary. (c)Extracted hair-region

Fig. 5 Hair region segmentation

4 Hair Sculpturing

Our target in this work is to reconstruct and add 3D hair shape on the CT/MRI reconstructed 3D-head only. To do this, we need a good sculpturing object, which results a complete-head model after cutting out the non-hear-regions. We assume 3D sculpturing object as follows:

Initially we prepare a hairless solid-head filled with binarized voxels. As we mentioned before, there are two ways of making this solid-head: (1)by filling the 3D-head, and (2)by accumulating the binary converted CT slices. The latter way is preferable as it is easy to implement and there is no possibility of obtaining shape-error. Then a rectangular 3D box covers it. The size of the box is assumed as 30% more than that of the solid-head in each side. Except solid-head voxels rest of the box is filled with (which we call) hair-voxels. This 3D box as a whole is called the sculpturing-object. Both the solid-head-vexel and hair-voxel size in the sculpturing object are assumed as approximately one cubic *mm* ($0.908mm \times 0.908mm \times 1mm$).

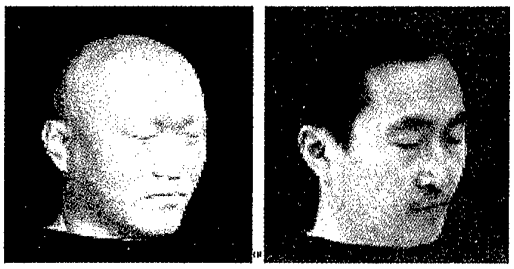
Let us consider a camera position as shown in Fig.2. Suppose we see the head and the hair as in Fig.3. After identifying the hair-region shown in Fig.5(c), we sculpture the hair-voxels by removing all the voxels outside the hair-region up to the solid-head, as shown in Fig.4. After a number of camera positions around the sculpturing-object are tried, the remaining voxels on the solid-head is the resulting hair-shape.

It is true that we are able to obtain hair-shape close to the actual as the number of camera positions increases. However, from a practical viewpoint, we consider a dozen of camera positions from the left, front and right sides, while maintaining the camera height at the ear and nose level.

The sculpturing method has an special characteristic that the concave and semi-occluded parts near the skin-hair junction (especially in the forehead region) can be reconstructed reasonably well. Usual sculpturing methods using silhouettes can deal only with convex shape but fails to provide concave parts. Primarily due to the presence of hairless solid-head (3D-head), our method can deal with the concave parts in/near the forehead.

5 Experiments

The CT data we employed was a size of $512 \times 512 \times 225$ with a resolution of $0.454mm \times 0.454mm \times 1mm$. The 480×480 pixels video captured images were taken with a SONY digital video camera of 640×480 resolution. A $1m \times 1m$ blue sheet was used as the background. The person sat on a normal revolving chair. The video image was taken by keeping the camera at a fixed position while he himself rotates the chair by his leg. His head level waved slightly. Because of our superior registration scheme, there is no need to hold the head position very tightly.



(a) 3D-head (b) Textured mapped 3D-head

Fig.6 Hair-less 3D-head and its textured mapped images

Fig.6 shows the hair-less 3D-face and its textured mapped image. All the hairstyles shown in this paper are added on this 3D-head. So far we performed the reconstruction task on the same individual with three different hairstyles. One of those is the original hairstyle and the two others are wigged. One of the video captured images of original hairstyle is shown in Fig.7a, the reconstructed complete-head and final-head are in Fig.7b and Fig.7c respectively, and the textured mapped final-head in Fig.7e. Fig.8 shows the same type of images for one of the wigged hairstyles. Artificial color is added on the final-head as shown in Fig.7d and Fig.8d.

The final-heads in Fig.7 and Fig.8 were reconstructed from ten images. In this paper we emphasis only on the reconstruction of the concave and semi-occluded regions. The surface quality of the hair-region can be improved by increasing the number of images. Whereas the rest of the 3D-head (uncovered) remains unchanged as this is from the CT data.

In Fig.7e and Fig.8e it is seen that for texture mapped image hair-shapes from ten images are acceptable.

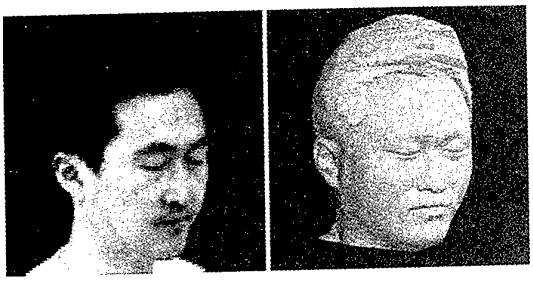
6 Conclusion and Future Works

Our method is to wrap-up the facial image obtained from CT with hair-voxels and to remove the non-hair regions obtained from a sequence of images. The novelty of this research is to deal with the concave parts in/near the forehead (hair-skin junction regions). Whereas the usual sculpturing methods using silhouettes can deal with the convex shapes only. The reconstruction method is simple and easy to implement on hospital environment where a CT scanner is readily available. The additional requirements are only a computer and a digital video camera.

To obtain fully automatic registration, we discourage to use hair-shape, which covers the ears completely. This is because the ear edge is one of the landmarks for our edge-based registration.

Our next target is to simulate dynamically the post-surgical facial expressions (e.g., laughing, jaw-movement etc.) for the cancer patient having facial

tumor, especially after replacing facial soft-tissue and/or removing a part of facial bone.

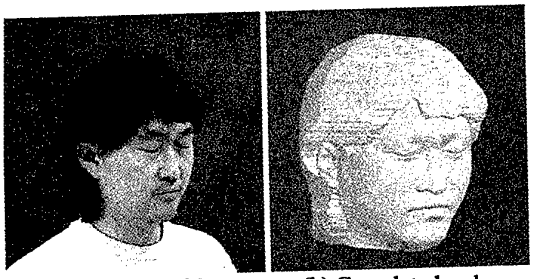


(a) Video Captured Image (b) Complete-head

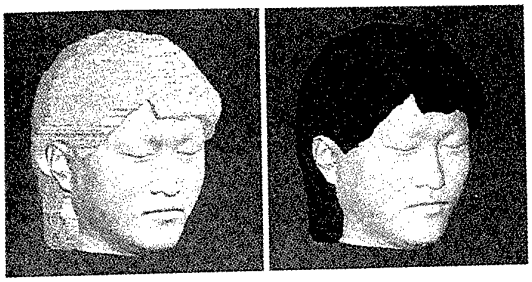
(c) Final-head (d) Final Head with artificial color

(e) Textured mapped Final-head

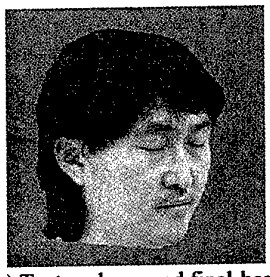
Fig. 7 Reconstructed head with original hairstyle



(a) Video Captured Image (b) Complete-head



(c) Final-head (d) Final Head with artificial color



(e) Textured mapped final-head

Fig.8 Reconstructed head with a wigged hairstyle

7 Acknowledgment

The authors would like to thank Dr. R. Sekiguchi and other staffs of the National Cancer Center, East Hospital, Kashiwa, Chiba, Japan for providing medical guidance and necessary CT data.

References

1. Ling-Xiang Zhau and Wei-Kang Gu: "3D Model Reconstruction by Fusing Multiple Visual Cues," *Proceedings of ICPR'98*, Brisbane, Australia, August 16-20, pp.640-642 (1998).
2. Ali, Md. Haider, Eiji Takahashi and Toyohisa Kaneko: "A 3D Face Reconstruction Method from CT Image and Color Photographs," *IEICE Trans. on Information and Systems*, E81-D(10), pp1095-1102 (1998).
3. Ali, Md. Haider and Toyohisa Kaneko: "Automatic Reconstruction of 3D Human Face from CT and Color Photographs," *IEICE Trans. on Information and Systems,* E82-D(9) pp1287-1293 (1999).
4. Jiang Yu Zheng and Fumino Kishino: "Recovering 3D Models from Silhouette Sequence and Detecting Unexposed Regions," *IEICE Trans. on Information and Systems*, J76-D-II(6), pp1114-1122. (1993). (in Japanese)
5. Shigeki Sugimoto and Masatoshi Okutomi: "Shape Estimation of Rotating Object Using Spacio-temporal Images," *Transaction of Information Processing Society of Japan,* 40(6) pp2717-2723. (1999). (in Japanese)
6. Richard Szeliski and Richard Weiss: "Robust Shape Recovery from Occluding Contours Using a Linear Smoother," *Proc. ARPA Image Understanding Workshop,* Washington, D.C., pp939-948 (1993).

7. P. Giblin and Richard Weiss: "Reconstruction of surfaces from profiles," *First International Conference of Computer Vision (ICCV'87)*, London, England, pp136-144 (1987).

8. Y. Matsumoto, H. Terasaki, K. Sugimoto and T. Arakawa: "A Portable Three-dimensional Digitizer," *Inter-national Conference on Advances in 3d Digital Imaging and Modeling*, pp197-204 (1997).

9. Eric N. Mortensen and William A. Barrett: "Intelligent Scissors for Image Composition," *Computer Graphics Proceedings, SIGGRAPH 95,* Los Angeles, California, August 6-11, pp191-198 (1995).

10. T. Akimoto and Y. Suenaga: "Automatic Creation of 3D Facial Model," *IEEE Computer Graphics and Applications*, 13(4) pp16-22 (1993).

11. Ferderic Pighin, Jamie Hecker, Dani Lischinski, Richard Szeliski and David H. Salesin: "Synthesizing Realistic Facial Expressions from Photographs," *Computer Graphics Proceedings SIGGRAPH'98*, Orlando, Florida, pp75-84 (1998).

ICAT 2000

Terrain Rendering With View-Dependent LOD Caching

Ming Fan Ueng Jung Hong Chuang
Department of Computer Science and Information Engineering
National Chiao Tung University
Hsinchu, Taiwan, ROC
{mfueng, jhchuang}@csie.nctu.edu.tw

Abstract

Real-time and smooth rendering of a large-scale terrain data has been a challenging problem. In this paper, we propose a geometrically continuous view-dependent level-of-detail (LOD) modeling aiming to speed up the generation of terrain mesh and in the meantime achieve a satisfied image quality. The terrain data is subdivided into blocks and each of which will possesses its own LOD mesh that is dynamically determined according to the viewing parameters. Between two adjacent blocks, a dike structure is proposed that aims to provide a smooth blending between two meshes of different levels of detail, and hence remove cracks that usually occur in previous methods. We also propose a mechanism in LOD modeling that caches the LOD of a block for the possible reuse in the following frames after it is generated. Since LOD selection and generation in general requires computation on each node level, such a LOD caching can potentially contribute a considerable saving of computation time.

Key words: LOD, Terrain rendering, Caching

1. Introduction

A rendering system is a kernel for visual simulation and virtual reality applications. In such applications, we are very much concerned about the high-performance and real-time visual capability. This leads to the quest of high resolution, low latency, and high but constant frame rate in the visual display. In the past years, many techniques have been proposed. Among them, we mention fast view and back-facing culling, visibility culling, level-of-detail modeling, hybrid rendering, and image-based rendering.

As a special case of the general rendering system, a terrain rendering system usually takes a terrain grid with high-field values as input, and has found applications in flight simulations, tank simulations, and other GIS applications. Most applications usually cover a very large area, and hence require a large-sized terrain grid. This results in too many polygons to be efficiently rendered by the current hardware. Level-of-detail (LOD) modeling has been proven to be a very effective technique for reducing the number of polygons.

This paper describes techniques for removing cracks that occur between two adjacent blocks of different LOD, and for caching LOD of a block and possible reuse in the following frames. In the following sections, we review previous work, and we describe the dike structure for blending two different LOD models and the cache mechanism of LOD model, and finally we show several experimental results.

2. Related Work

LOD modeling techniques for terrain grid can be classified into two major mesh structures: regular square grid (RSG)[2,3,6,8] and triangulated irregular network (TIN) [4,7,9].

In RSG approach, terrain grid is usually subdivided into blocks to avoid global propagation in dependency checking during LOD construction [2,3,6,8]. Such a block subdivision also provides a good support in the view culling and paging mechanism. In [6], a quadtree structure is used for each block. The quadtree structure is explicitly and hierarchically constructed based on a regular and symmetric triangulation of the grid vertices.

This hierarchical structure allows efficient derivation of a LOD model for new viewing parameters. During view dependent navigation, the delta segment projection for a node will be tested to see if the node should be simplified or refined. A block-LOD-reduction scheme is also used to reduce the LOD construction time by alleviating the testing at a huge number of nodes and allowing LOD be determined on the block basis. RSG approach has several advantages. For examples, Delaunay trianglation can be easily maintained for view-dependent selective refinement, switching between levels can be efficient and simple, and fast triangle strip can be easily constructed. It, however, produces for each block a mesh that is usually not optimal, and has cracks between two adjacent blocks of different LOD resolutions..

In TIN approach, vertices can be added and removed, or connection can be modified in order to obtain a mesh that is better approximating the original shape [4,7,9]. As a result, a reduced mesh with better approximation; but less polygons is generally possible. Comparing to RSG approach, this approach usually requires more computation time, is more troublesome to locally modify a terrain model , and less efficient in performing collision detection.

3. The Proposed Terrain Rendering System

3.1 Overview

The terrain rendering system we implemented takes the RSG approach; that is, we take the terrain triangulation as in [6]. As a preprocessing, we divide the terrain grid into blocks with a dike between each pair of adjacent blocks. In run-time, blocks are first tested for view-volume culling for each new frame, and for each of those blocks intersecting the view volume, we check to see if its cached LOD can be reused in the new frame. That is, the cached LOD can be reused if its projected error with respect to the new viewpoint is within a pre-specified error, or a new LOD should be re-generated if the test fails. The test is block-based: rather than vertex-based, and thus can be very efficient. After the LOD of all blocks within the view volume are ready, we triangulate the dikes such that the LOD models of different resolutions can be smoothly blended.

3.2 Hierarchical Structures

Two hierarchical structures are proposed. A dependency hierarchy is used to facilitate the run-time selective refinement. Moreover, we construct a triangle tree in such a way that triangle mesh can be efficiently derived once terrain vertices are selected for the current LOD without traversing terrain vertices one more time.

According to the triangulation rule in RSG approach, a terrain of $(2^n+1)\times(2^n+1)$ can be simplified to 2n+1 levels; as shown in Fig. 1 for n=2. A triangle tree is a binary tree in which each node represents a triangle in the RSG triangulation. The refinement on each triangle results in two triangles, representing the children of the corresponding parent node. Fig. 2 shows the triangle trees for a 3x3 terrain grid.

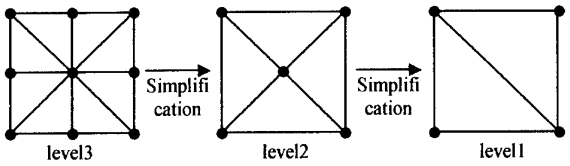


Fig. 1. Levels of LOD model.

While performing the refinement, a triangle in the RSG triangulation is subdivided into two triangles by adding a vertex on the bottom edge of the triangle. We call the top vertex of the original triangle is the mother or father vertex of the newly added vertex. The order that a vertex is selected for and added to the LOD model determines a hierarchy among terrain vertices. Vertices that are new in level l constitute the first level of the dependency hierarchy, and vertices that are newly added to level l form the l-th level of the dependency hierarchy. In the hierarchy, each vertex is associated with a father and a mother pointer pointing to its mother and father vertices. Fig. 3(c) is the dependency hierarchy for a 5 x 5 terrain grid shown in Fig. 3(a).

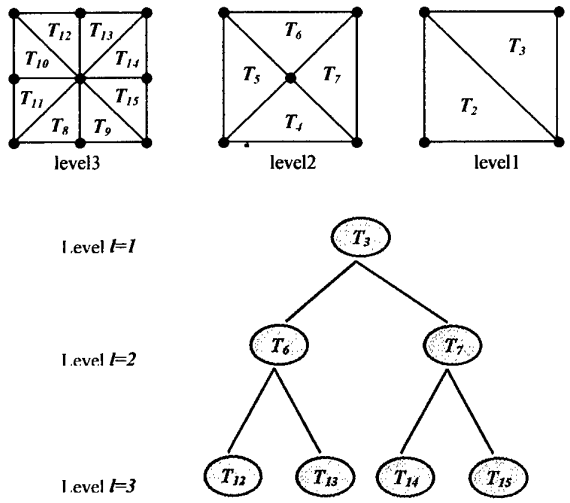
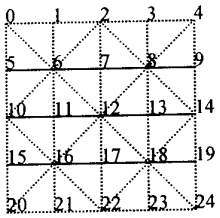
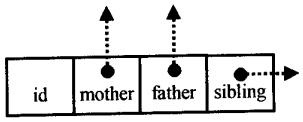


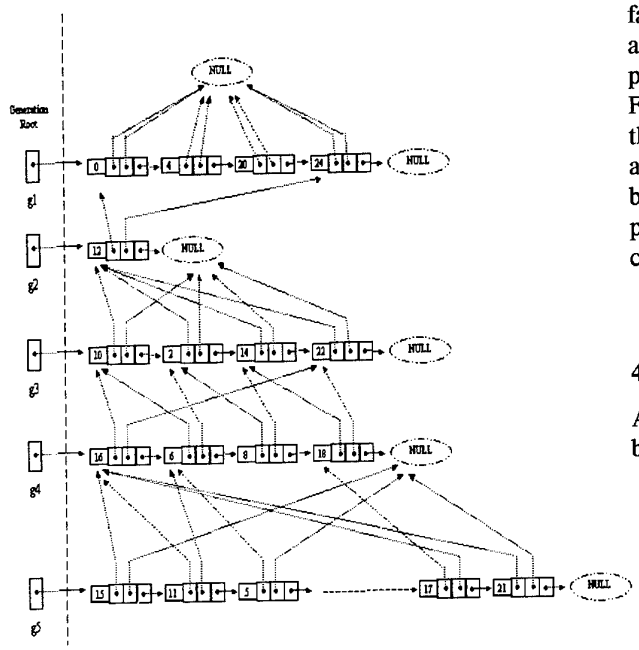
Fig. 2. Triangle binary tree.



(a) Vertex index



(b) Data structure for vertex



(c) dependency hierarchy

Fig. 3. Dependency hierarchy.

3.3 Dynamic Selective-Refinement

In navigation phase, the dependency hierarchy is traversed to derive the mesh of a desired resolution. When a node is visited, we do screen-error test to see if the projection of its height difference exceeds a pre-specified tolerance. If so, the vertex is selected, and in the meantime, its parent vertices are locked and selected without the screen-error test.

Each vertex is associated with two more variables, namely *active* and *lock*. The variable *active* is a Boolean recoding the selection state of the vertex. The variable *active* is TRUE when the vertex is selected, and FALSE otherwise. The variable *lock* for a vertex v is an integer recording the number of vertices that are children of v and are either selected or locked. A nonnegative *lock* means that v is locked, and a zero *lock* represents that v is not locked.

Two operations are involved in selecting the vertex v. *Dependency* operation switches the *active* variable of v from FALSE to TRUE while *unlocking* operation does oppositely. In dependency operation, the variable *lock* of parent vertices of v must be increased by 1. In case v has *lock* value 0, parent vertices of v must repeatedly perform dependency operation. In unlocking operation, the variable *lock* of parent vertices of v must be decreased by 1. In case v has *lock* value 1, parent vertices of v must repeatedly perform unlocking operation.

The dependency hierarchy is traversed in a bottom-up fashion. If a vertex is locked, its corresponding triangles are put into the display list. If a vertex is not locked and passes the screen-error test, the *active* variable becomes FALSE and unlocking operation is performed, provided that its *active* variable is TRUE. If a vertex is not locked and fails to pass the screen-error test, its *active* variable becomes TRUE and dependency operation is performed, provided that its *active* variable is FALSE, and its corresponding triangles are put into the display list.

4. Removing Cracks

A dike structure is proposed to remove cracks occurring between two adjacent blocks of different LOD resolutions. See Fig. 4 for illustration.

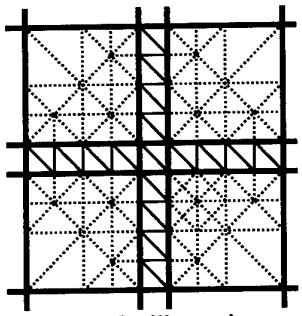


Fig. 4. Dike structure.

After LOD models are obtained for all blocks, we begin to triangulate the dike area one by one without altering the selection state of block's boundary vertices. In our implementation, each dike area is first completely triangulated and then simplified based on edge collapsing guided by the selection status of block's boundary vertices.

5. LOD Caching

The screen-error test mentioned in previous section takes the projection of vertex's height difference into account. As shown in Fig. 5, the height difference of B, denoted as δ_B, is defined as the deviation in z-direction from B to ΔAEC.

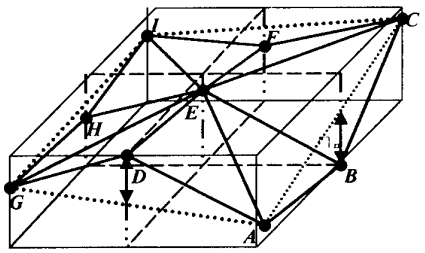


Fig. 5. Height difference on a vertex.

Following the formula in [6], vertex v will pass the screen-error test if

$$\delta^2_{screen}(\mathbf{e},v)=\frac{d^2\lambda^2\delta^2\left((e_x-v_x)^2+(e_y-v_y)^2\right)}{\left((e_x-v_x)^2+(e_y-v_y)^2+(e_z-v_z)^2\right)^2}\le\tau^2$$

where **e** is the eye point, d is the view plane distance, λ is the ratio of the unit length in world coordinate system over the pixel size in the screen coordinate system, δ is the height difference on vertex v, and τ is a user-specified error tolerance. The above formula can be viewed differently to define a s called *allowable height difference* of v as follow:

$$\delta^2_{allowable}(\mathbf{e},v)=\frac{\tau^2\cdot\left((e_x-v_x)^2+(e_y-v_y)^2+(e_z-v_z)^2\right)^2}{d^2\lambda^2\left((e_x-v_x)^2+(e_y-v_y)^2\right)}$$

As a result, the screen-error test is equivalent to testing if $\delta_v \le \delta_{allowable}(\mathbf{e},v)$.

The LOD caching mechanism aims to cache the LOD of a block for possible reuse in the following frames with the requirement that the screen error is within a user-specified tolerance. We first denote the *projection bound of the delta allowable height difference* as s (in pixel unit), and suppose that a cached LOD model can be reused if, for a new viewpoint, the projected delta allowable height difference of the LOD model is less than or equal to s.

Consider a vertex v_i, we have $\delta_{allowable}(\mathbf{e}_0,v_i)$ and $\delta_{allowable}(\mathbf{e}_1,v_i)$, respectively for viewpoints $\mathbf{e}_0$ and $\mathbf{e}_1$. By replacing τ with s, we obtain the bound on delta allowable height difference of v_i with respect to $\mathbf{e}_0$ as follows:

$$\varepsilon(\mathbf{e}_0,v_i)=\frac{s\cdot\left((e_{0x}-v_{ix})^2+(e_{0y}-v_{iy})^2+(e_{0z}-v_{iz})^2\right)}{d\lambda\sqrt{\left((e_{0x}-v_{ix})^2+(e_{0y}-v_{iy})^2\right)}}$$

We then claim that the selection state of v_i with respect to $\mathbf{e}_0$ can be preserved while viewing from $\mathbf{e}_1$ is the delta allowable height difference $\Delta\delta_{allowable}(\mathbf{e}_0,\mathbf{e}_1,v_i)$ is less than $\varepsilon(\mathbf{e}_0,v_i)$, where $\Delta\delta_{allowable}(\mathbf{e}_0,\mathbf{e}_1,v_i)$ is $\left\|\delta_{allowable}(\mathbf{e}_1,v_i)-\delta_{allowable}(\mathbf{e}_0,v_i)\right\|$. In such case, we can show that preserving the selection state of v_i results in a projected height difference bounded by $\tau+s$.

Next, we extend the preserving of vertex's selection state to the LOD caching of a block. For the LOD caching of a block, we, in principle, need to check if $\Delta\delta_{allowable}(\mathbf{e}_0,\mathbf{e}_1,v_i)<\varepsilon(\mathbf{e}_0,v_i)$ for all v_i in the block. This is, however, very time consuming. Since $\varepsilon(\mathbf{e}_0,v_i)$ becomes smaller when the distance between v_i and $\mathbf{e}_0$ gets smaller, it is reasonable to say that $\varepsilon(\mathbf{e}_0,v_i)$ is larger than or equal to the minimum of $\varepsilon(\mathbf{e}_0,v_{lt})$, $\varepsilon(\mathbf{e}_0,v_{rt})$, $\varepsilon(\mathbf{e}_0,v_{lb})$, and $\varepsilon(\mathbf{e}_0,v_{rb})$, provided that $\mathbf{e}_0$ and $\mathbf{e}_1$ are outside the block; see Fig. 6.

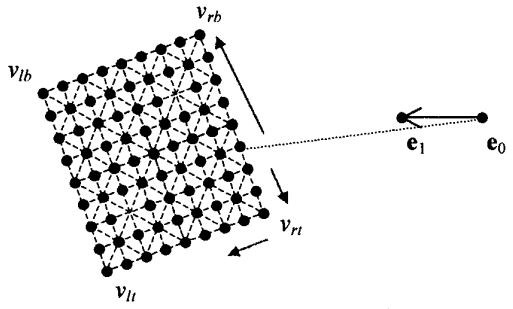


Fig. 6. Variation on tolerable height deviation.

6. Experimental Results

We have implemented the proposed scheme using C language, OpenGL, and GLUT library. Experiments have been performed using terrain data of Dan-Shoei River. Results are obtained on a PC with Pentium III 660Mhz CPU, 128MB Ram, and GeForce 256 3D graphics card.

The terrain data includes an area of 26,400mx26,400m, and is divided into 20x20 blocks, each of which has 33x 33 grid vertices. A complete triangulation of this terrain data has 868,488 triangles. We set up a navigation path with height about 1,000m, 40 degrees for field of view, and a display window of 800x800 pixels.

Table 1 depicts the performance of LOD caching mechanism based on several different τ and different s for each τ. More detail analysis is shown in Table 2. Using LOD caching, we have seen a 25% to 46% speed-up in frame time and 92% to 98% speed-up in LOD construction time. Note that the LOD models obtained using LOD caching have less number in triangles, ranging from 2.2% to 8.7% in our experiment. Our experience shows that the change ranges from 3% to 4% when s = 0.1 τ. Figures 7, 8, and 9 show the

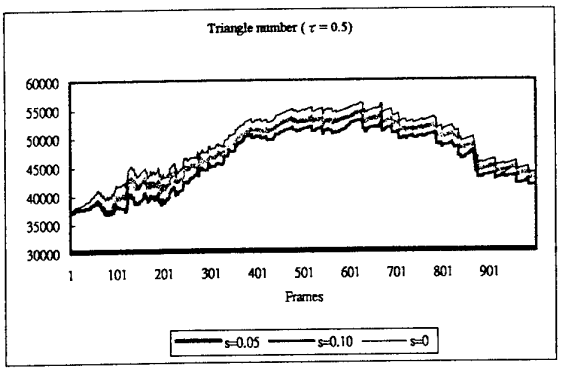
performance plots for various τ .

τ (pixel)	S (pixel)	Average triangle number	Average LOD generation time (ms/frame)	Average rendering FPS
0.5	0.05	47,487	1.2	6.9
	0.10	45,972	0.7	7.1
	-	49,097	34.8	5.5
1.0	0.05	24,595	1.4	13.7
	0.10	24,025	0.8	14.0
	0.20	22,950	0.6	14.9
	-	25,158	23.7	10.2
2.0	0.10	11,028	0.9	32.1
	0.20	10,770	0.7	32.4
	-	11,277	13.3	22.1

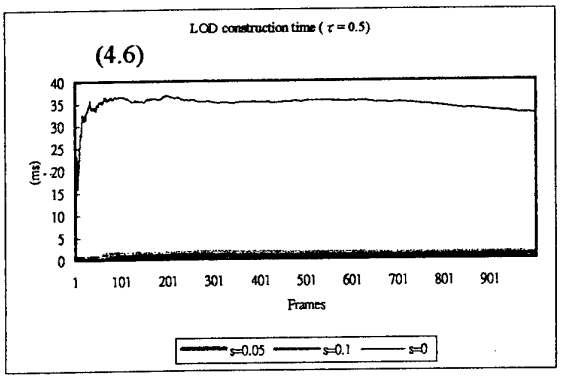
Table 1. Performance of LOD caching - 1.

τ (pixel)	S (pixel)	Change in average triangle number	Average LOD generation time		(c) Average gain factor on frame time
			(a) Gain factor on LOD generation	(b) Gain factor on frame time (due to LOD caching)	
0.5	0.05	-3.3%	96.7%	90.1%	25.4%
	0.10	-6.4%	98.1%	83.2%	30.3%
1.0	0.05	-2.2%	94.3%	89.2%	34.1%
	0.10	-4.5%	96.5%	84.8%	37.6%
	0.20	-8.8%	97.3%	74.5%	46.0%
2.0	0.10	-2.2%	92.9%	88.6%	45.5%
	0.20	-4.5%	94.5%	90.0%	46.7%

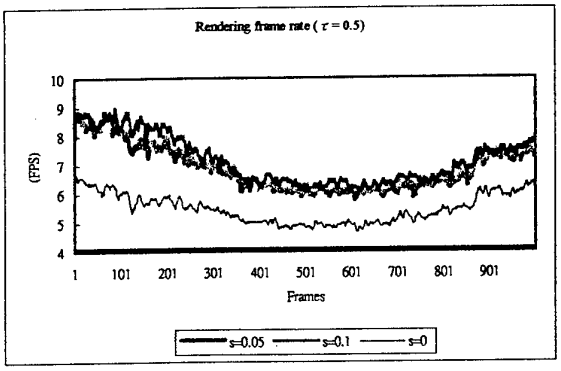
Table 2. Detail analysis on performance of LOD caching.



(a)

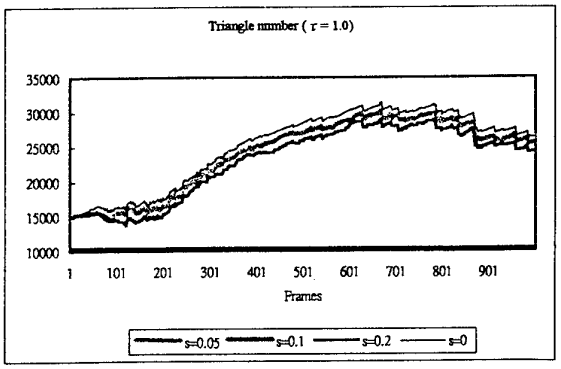


(b)

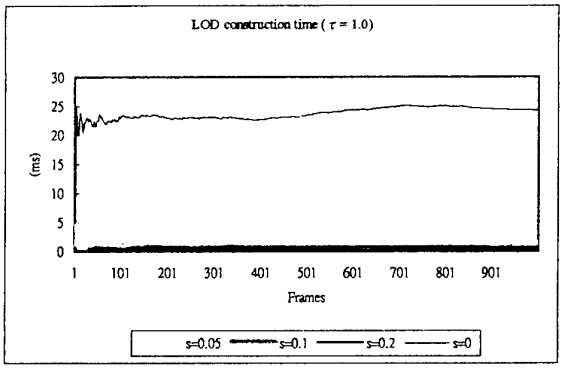


(c)

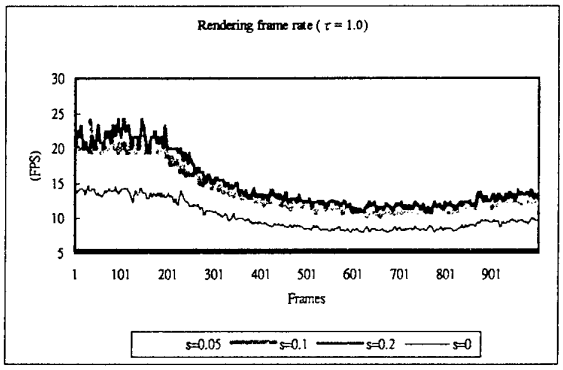
Fig. 7. Performance plot for $\tau = 0.5$.



(a)

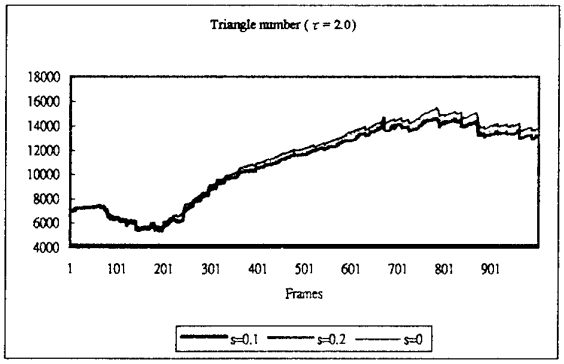


(b)

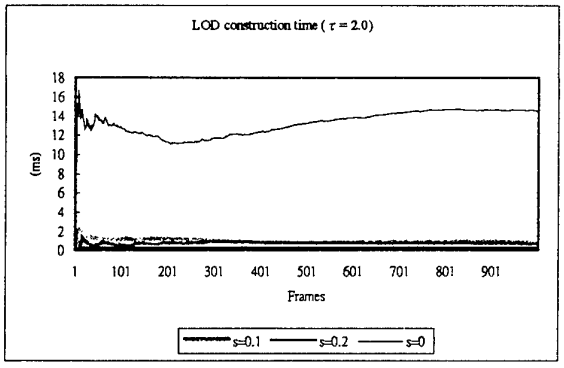


(c)

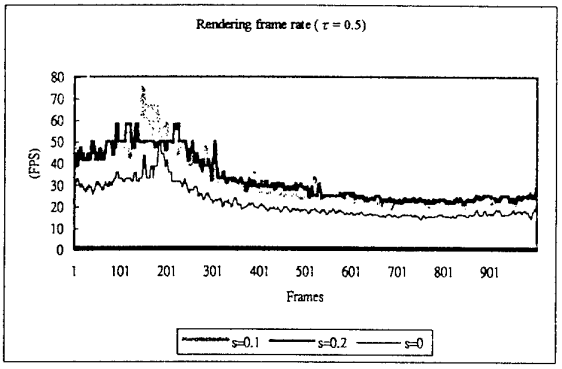
Fig. 8. Performance plot for $\tau = 1.0$.



(a)



(b)



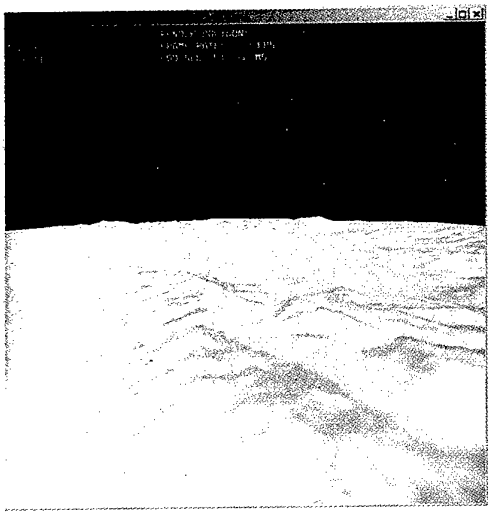
(c)

Fig. 9. Performance plot for $\tau = 2.0$

To do a better examination on quality performance of the proposed LOD caching mechanism, we count the number of vertices that should be selected but are not selected due to LOD caching; that is, those vertices that have projected height difference exceeding τ ; but are not selected. Table 3 depicts that, when $s = 0.1\ \tau$, the percentage of those vertices is bounded by 2%. Figures 10 and 11 are two images obtained in navigating the Dan-Shoei River.

τ (pixel)	S (pixel)	Average number of selected vertices	Vertices: should be selected; but not selected	
			Average number	%
0.5	0.05	23,731	374	1.5%
	0.10	22,971	782	3.4%
1.0	0.05	12,239	94	0.7%
	0.10	11,953	205	1.7%
	0.20	11,414	434	3.8%
2.0	0.10	5,425	35	0.6%
	0.20	5,295	74	1.3%

Table 3. Quality performance of LOD caching.



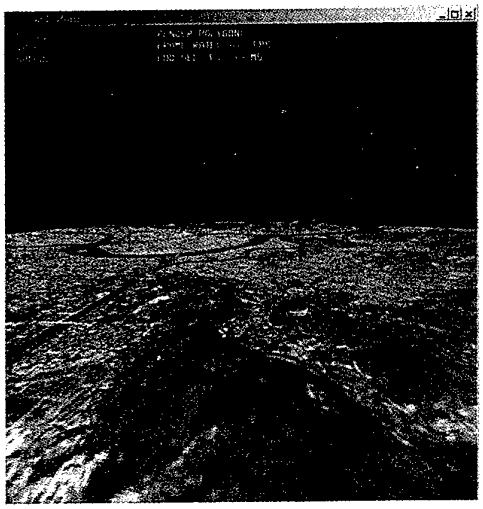
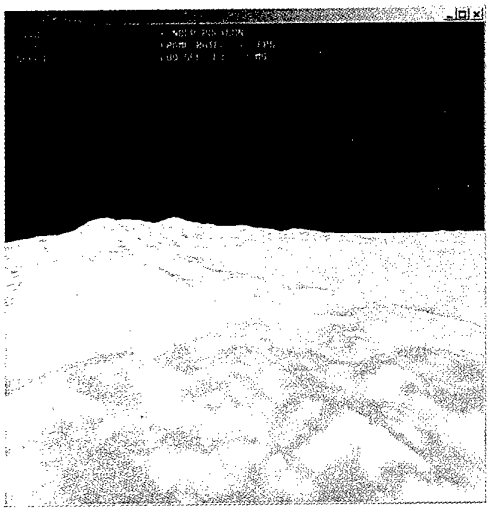


Fig. 10. Terrain image of Dan-Shoei River.



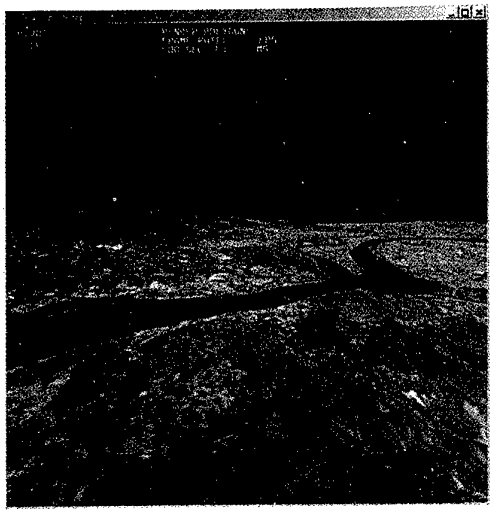


Fig. 11. Another terrain image of Dan-Shoei River.

4. Concluding Remarks

We have presented a terrain rendering system in which a dike structure and a LOD caching mechanism have been proposed to, respectively, remove cracks usually occurring in the boundary of adjacent blocks and speed up the LOD selection by reusing previously constructed LOD models. The experiments we have done revealed that dike structure successfully blends two LOD models of different resolution, and the LOD caching mechanism is able to speed up the LOD construction by 92-98%, and the frame time by 25-46%. Among future study plans, we will focus on frame-time control and the integration of hybrid rendering techniques into terrain rendering systems.

References

1. M. D. Berg, and K. T. G. Dobrindt, *1 On Levels of Detail in Terrains"* Tech. Rep. UU-CS-1995-12, Department of Computer Science, Utrecht University, April 1995.

2. M. Duchaineau, M. Wolinsky, D. E. Sigeti, M. C. Miller, C. Aldrich, and M. Mineev. *"ROAMing Terrain (Real-Time Optimally Adapting Meshes)."* Proceeding of IEEE Visualization '97.

3. C. S. Fahn and S. T. Wu. *"The View-Dependent Real-Time Rendering of Large-Scale Terrain in Continuous Level of Detail."* In Proceeding II of NCS99, Pages B374-B381.

4. R. J. Fowler and J. J. Little. *#Automatic Extraction of Irregular Network Digital Terrain Models."* Computer Graphics (Proceeding of SIGGRAPH '79), Vol. 13, No. 2, pages 199-207.

5. R. Klein and D. Cohen-Or. *#Incremental View-dependent Multiresolution Triangulation Terrain."* Proceeding of Pacific Graphics, 1997.

6. P. Lindstrom, D. Koller, W. Ribarsky, W. Hodges, L. Faust, and G. Turner. *"Real-Time, Continuous Level of Detail Rendering of Height Fields."* In Proceeding of ACM SIGGRAPH `96.

7. H. Hoppe. *"Smooth view-dependent level-of-detail control and its application to terrain rendering."* Proceeding of IEEE Visualization `98, October 1998, pages 35-42.

8. R. Pajarola, *"Large Scale Terrain Visualization Using The Restricted Quadtree triangulation."* Proceeding of IEEE Visualization `98.

9. M. F. Polis, and D. M. Mckeown. *"Iterative TIN Generation from Digital Elevation Models."* In Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, June 16-18, 1992, pages 787-790.

10. J. Shade, D. Lischinski, D. H. Salesin, T. DeRose and J. Snyder. *"Hierarchical Image Caching for Accelerated Walkthroughs of Complex Environments."* Proceeding of ACM SIGGRAPH `96, pages 75-82.

ICAT 2000

View-Dependent Focal Blur for Immersive Virtual Environments

Naoki Hashimoto , Masayuki Nakajima
Graduate School of Information Science & Engineering,
Tokyo Institute of Technology
2-12-1, Ookayama, Meguro-ku, Tokyo, Japan
{ *naoki, nakajima* }*@img.cs.titech.ac.jp*

ABSTRACT

Recently, researches using immersive virtual environments are widely carried out. While computers and projection devices become highly efficient, image distortion and perception errors, etc. become a problem in virtual environments. Therefore, the technique for more accurately transmitting the contents of virtual environments to the user has been required. Based on a such background, purposes of this research are to provide a view-dependent focal blur in immersive virtual environments and to consider that effects on depth perception. Focal blur enables us to perceive depth informations accurately in 3-D computer graphics. Therefore, it can provide better reality and presence of virtual environments.

In this research, we realized the view-dependent focal blur by the method for not depending on a screen position and our view direction in real time. Then, the effectiveness of this technique on depth perception was shown through some experiments.

Key words : Virtual Reality, Immersive Virtual Environment, Focal Blur, Depth Perception

1 Introduction

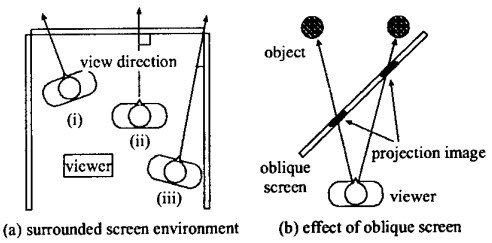
In recent years, immersive projection displays have been attracting the attention of researchers interested in VR (Virtual Reality). A CAVE system [1], which had been developed at the University of Illinois at Chicago in 1993 is typical system of an immersive projection display. By the present, many clone systems of CAVE [2] are made. These systems can generate highly immersive virtual environments. The application to various fields, therefore, is expected.

Immersive projection displays, however, has some problems in the practical use [3]. As one of the large problems, the special sensory property in the virtual environment generated with immersive projection displays is mentioned. Because of this property, we are confused when we use virtual environments. The main cause of this property is 1) measurement errors of a 3-D position tracker, and 2) effects of oblique screens used for immersive projection displays.

In some virtual environments, user's viewpoint and its direction are measured with 3-D position trackers. The 3-D scene is generated with these user's viewpoint information got by the 3-D sensors. The sensors are, however, very sensitive for its installation environment, and it is very difficult to reduce measurement errors. The generated 3-D scene gives the discomfort to the users when the measurement errors are included for user's viewpoint information. There is a object in front of a user in a virtual environment, for example. The object's position is not changed when the user moves toward the object. That position is, however, changed when the viewpoint information of the user has some measurement errors. The user is, therefore, greatly confused by this phenomenon. Therefore, the measurement error of the viewpoint exerts an enormous influence on the 3-D scene when we use an immersive projection display. The precision of the 3-D position tracker is, therefore, very important.

On the other hand, the effect of oblique screens is also a serious problem. It is possible that users freely move in the region surrounded with screens, when immersive projection displays are used. The positional relation of screens and user's viewpoint, therefore, dynamically change as shown in Figure 1(a). In the situation shown in figure 1(a)-iii, the user has to extremely view the screen from a oblique direction. In this case, depth perception errors which is peculiar to immersive projection displays occur. It is generally considered that the cause of this error is the effect by the focus adjustment increasing further than the parallax information for realizing the stereoscopic image [4]. This tendency strengthens, when the screen is more viewed from a oblique direction.

This situation is explained in detail using Figure 1(b). In this figure, there are two object placed



(a) surrounded screen environment (b) effect of oblique screen

Figure 1: Positional relation between viewpoint and screen

at the equal distance from a viewpoint, and these objects are completely identical. The distances to the projection images on the screen from the viewpoint are, however, greatly different because of the oblique screen. In such situation, it is not possible to focus on those two projection images simultaneously. In immersive projection displays, 3-D images with parallax in order to realize stereoscopic images are generated, and users can accurately perceive the depth. With information got from the focus adjustment, the user, however, perceives that the object which projection image is more close to the viewpoint than the other is placed more close to them when the effect of the oblique screen becomes more strong as shown in Figure 1(b).

Especially in this study, the effect of the oblique screen is noticed and focal blur effects are introduced as new information for reducing the depth perception error. In the daily life, our view is blurred depending on the focus point, and the focal blur is very important to perceive depth information. With recent immersive projection displays, we can realize binocular stereoscopic vision and changes of our viewpoint with high resolution images. A few VR systems, however, consider the focal blur [5]. In order to construct more natural and more realistic VR systems, it is necessary to consider focal blur depending on our viewpoint.

In this paper, we realized view-dependent focal blur in immersive virtual environments generated with immersive projection displays like a CAVE system. Then we carried out two experiments in order to examine the relationship between view-dependent focal blur and depth perception in virtual environments.

The remainder of this paper is structured as follows. Section 2 reviews previous related work. Section 3 describes the view-dependent focal blur. Section 4 illustrates a variety of results of some experiments. Section 5 discusses advantage and defect of our approach. Finally, section 6 provides conclusions.

2 Related Work

On a focal blur effect, many examinations have been carried out. In this section, some of them are introduced briefly.

Matter [6] observed that depth perception was produced only with focal blur. He placed the region with focal blur in a natural image, and the depth feel got from that region was evaluated. The evaluation of the case that blur reaches the edge of the region is also carried out. As the result, it was proven that the depth feel could be intentionally controlled by selectively adding the blurred region.

Shipley et al. [7] investigated the independent effects of three aspects of aerial perspective: blur, contrast and color change. They prepared many natural images applied these effects. Each image contained a pair of similar objects with a natural background. A subject's task was to indicate which object appeared closer. This experiment showed that focal blur assisted the depth perception.

The research on atmospheric effects which consider the view direction of the user though it has no direct relationship on focal blur is also carried out. "Fog" is famous as a effect for showing the image more naturally. Fog makes objects fade into the distance. It can be used to simulate haze, mist, smoke, or pollution. Fog, however, generally functions in the front-back direction of a display because the distance used for generating fog is the eye-coordinate distance between the viewpoint and the object. In order to solve this problem, Heidrich [8] proposed "Euclidean distance fog". In this method, the true Euclidean distance from the viewer to the object is used to compute more accurate fog. Euclidean distance fog ,therefore, effects with the dependence to the view direction. This is most useful in visual simulation application where realism is a top requirement.

The researches introduced in previous paragraphs had made a non-stereoscopic image to be an object. The study which used focal blur effect for stereoscopic image is shown next. Okajima et al. [9] developed a rendering system that can simulate focal blur of the human lens in real time. The system provides focal blur information in 3-D computer graphics images while the observer's eyes are moving around naturally. In their research, focal blur was used in stereoscopic images. Three environment, 1) focal blur effect, 2) stereo effect, 3) focal blur and stereo effect, was presented to the user in a experiment. In each environment, two objects are presented and the one perceived more closer was selected by the user. From this experiment, it was found that a depth is most correctly perceived when focal blur effect and stereo effect were simultaneously used. It was also proven that focal blur was more effective for depth perception than stereo effect. Matter et al. [10] re-

ported same results about a relationship between focal blur effect and stereo effect.

In a special example, focal blur is used not to assist depth perception, but to obstruct user's view. Hirose et al. [11] used focal blur to simulate visual field of visually handicapped person in virtual environment. This experiment is useful for barrier-free town planning.

By summarizing all of this section, it was proven that focal blur is effective for depth perception by many previous works. In this paper, we try to create more realistic and immersive virtual environment by using focal blur on immersive projection displays in which depth perception error frequently arises.

3 View-dependent focal blur

In the real world, our vision is in perfect focus only for objects left in a certain distance from the viewpoint. The farther the object is from this focused point, the more out of focus it is. It is called focal blur effect. In general 3-D computer graphics, the focal blur effect is not used, and everything we draw is in focus. Not only the lose of reality, but also it results in the lose of accurate perception of the 3-D scene.

In order to solve this problem and offer new information for the accurate depth perception described in a previous section, we introduce view-dependent focal blur which modified DOF (Depth-of-Field) effect [12] in immersive virtual environments.

3.1 Algorithm

In many VR systems, 3-D scenes are rendered with OpenGL. A method to realize DOF effect generally uses the accumulation buffer which is a part of OpenGL. This method is briefly shown in Figure 2(a).

In this method, we have to choose some pseudo viewpoints so that positions of them vary slightly around a true position and each viewing volume shares a common rectangle that lies in a perfectly focused plane. The images generated from these pseudo viewpoints are synthesized with the accumulation buffer. After this process, images which include focal blur effect are generated.

However, the relationship between our view direction and a position of a screen changes dynamically in immersive projection displays surrounded by large screens like a CAVE system. In the conventional technique, the focus plane and the screen must be parallel. Focal blur, therefore, functioned only for the front-back direction of the screen as shown in Figure 2(b). It is clear that the conventional technique is insufficient in immersive projection displays. To resolve this problem, we modified the traditional method as shown in Figure 2(c). In this new method,

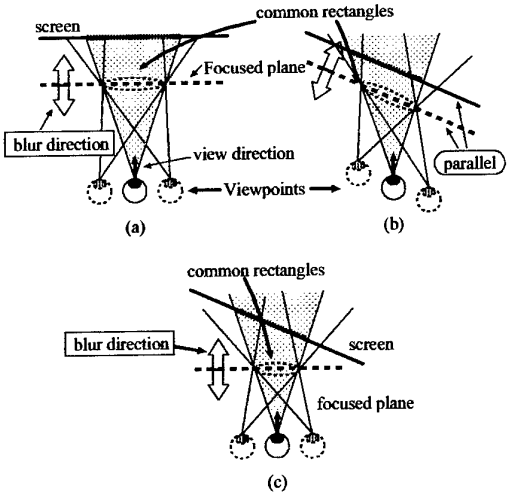


Figure 2: Viewing volume for view-dependent focal blur

the direction of the focus plane follows our view direction obtained with a 3-D position sensor. Focal blur, therefore, effects without dependence on the positional relationship between our view direction and a position of a screen.

With this method, the view-dependent focal blur is realized in immersive virtual environments. By using high-end graphics workstations, this method can be processed in real-time.

3.2 Application to actual 3-D scene

In this section, we introduce a sample 3-D scene rendered with the view-dependent focal blur effect. Positions of virtual objects drawn in that scene are illustrated in Figure 4. In this figure, a user views a screen from a oblique direction, and virtual objects arranging at 3 rows are placed in front of the user.

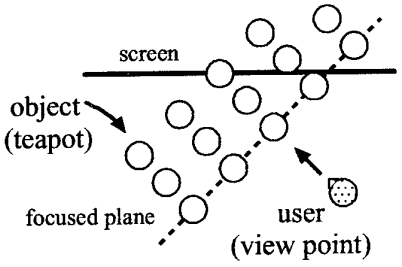


Figure 3: Structure of example scene

The rendered image of this scene is shown in Fig-

ure 4. Teapot images are used as virtual objects. In this image, objects are more and more blurred as their distance from a perfectly focused plane increases. It is very important that the focal blur effect is depend on the distance from not the screen but the user's focused plane.

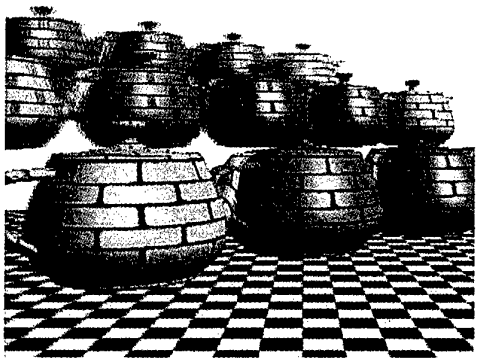


Figure 4: 3-D scene with view-dependent focal blur

3.3 Effect on visual acuity

Our proposed method gives focal blur effect on the region which is out of focus on the screen. In other words, the region is hard to be observed according to this blur effect. In this section, we measured the degree of focal blur effect with visual acuity as a criterion for evaluation.

The environment for the evaluation is shown in Figure 5. Subjects focused their eyes on a object which is d1 distant from their viewpoint. Next, we presented an eye examination chart at the distance of d2 from the subject's viewpoint, and measured their visual acuity. The eye examination chart contains "Randolt ring" generated by computer. The Randolt ring was displayed for 150 msec. This period is shorter than the time until the eye adjusted the focus on the examination chart.

We carried out this evaluation in three situations listed in Table 1. In the situation A, a user focuses on a object and measures visual acuity by using a eye examination chart (i). In this measurement, the user's eyes are not focused on the chart. This situation simulates unfocused conditions in the real world. On the other hand, in the situation B, chart (ii) is used instead of chart (i). The chart (ii) is projected image of the chart (i) on the screen shown in Figure 5. This situation is normal virtual environments without focal blur. The user's eyes, therefore, equally focus on both the object and the chart (ii). In the situation C, focal blur is added to the situation B. It realizes

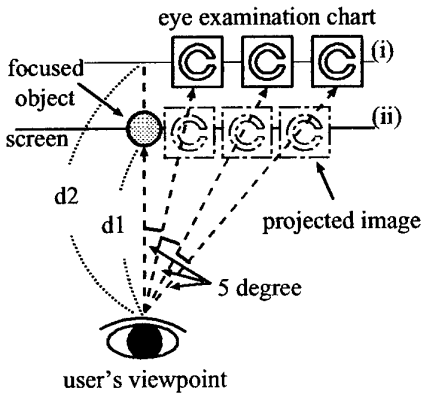


Figure 5: Evaluation with visual acuity

the virtual environment using focal blur effect. In all situation, distant parameters are d1 = 1500 *mm* and d2 = 2500 *mm*. These values are decided on the assumption of a CAVE system.

Table 1: Experimental condition

situation	eye examination chart	focal blur
A	(i)	OFF
B	(ii)	OFF
C	(ii)	ON

The result of this evaluation is shown in Figure 6. In this figure, a horizontal axis indicates a angle from a view direction focused on a object, and a vertical axis indicates visual acuity. The result of situation B is different from that of situation A. Situation B, which indicates a general virtual environment, realizes high visual acuity when the angle is within 20 degree. On the other hand, the result of situation C is similar to that of situation A. From these results, it is found that focal blur effect can realize a visual characteristic which is similar to the one in the real world.

4 Experiments and Results

In order to illustrate the relationship between the view-dependent focal blur and its effect on the depth perception in virtual environments, we performed two experiments.

The first one examined how the view-dependent focal blur contributed for accuracy of the depth perception in immersive virtual environments. In the situation in which the effect of the oblique screen

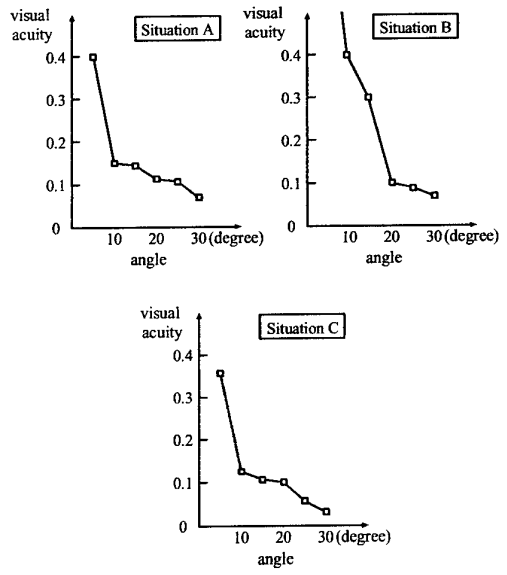


Figure 6: Visual acuity of unfocused region

was strong, the depth perception accuracy of the examinee was measured. In this time, the change of the accuracy by adding the view-dependent focal blur effect was observed.

The other one confirm reality reinforced with the view-dependent focal blur. In addition to conventional parallax information, the focal blur effect was added to the 3-D scene in this experiment. The change of the reality by adding the focal blur effect was measured.

In the following sections, we first explain our experiment equipment. Next, we describe each experiment in detail.

4.1 Experiment environment

We constructed a system shown in Figure 7 for following experiments. This is a simple immersive projection display. A large screen was prepared in order to cover a subject's field of view. The size of the screen is 120 inches. 3-D images generated with SGI Onyx is projected on the screen with a CRT projector.

We can also use a CAVE system. In the following experiments, however, precision of position measurement and flatness of the screen are extremely important. The 3-D position tracker which is used in our CAVE system contains some measurement error. The screens of our CAVE system is not completely flat, because these are soft screen stretched on a frame with some wires.

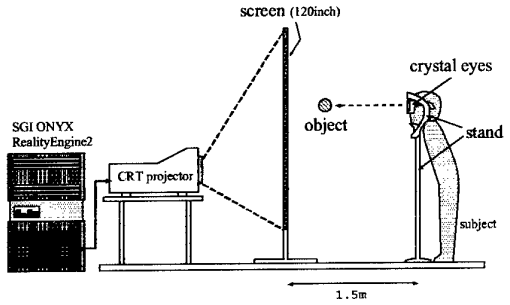


Figure 7: Experimental system

In order to resolve these problems, some contrivances are done in our experimental system illustrated in Figure 7. To begin with, a hard screen is used in the system. It is possible to remove distortion of projected images because the hard screen is perfectly flat. In addition, the head of the user is fixed by the stand. The position of the stand is precisely measured. It is possible to remove distortion of the images by the measurement error of the user's viewpoint because the head tracking is carried out without depending only on the 3-D position sensor.

In this system, parallax information is fundamentally contained. Subjects can experience a stereoscopic images by wearing a liquid crystal shuttering glasses.

4.2 Experiment 1: Evaluation of Accuracy

A purpose of this experiment is to study accuracy of the depth perception in immersive virtual environments with the view-dependent focal blur. For this experiment, the situation illustrated in Figure 8 was prepared. In this situation, we first placed two teapot objects at the same distance from a subject's viewpoint. The subject observe a oblique screen leaned toward thirty degrees from their view direction, on the assumption of immersive projection displays like a CAVE system. It is known that depth perception errors are occurred frequently and the users tend to perceive the object of the right side nearer because of the effect of the oblique screen as mentioned in the previous section.

In this experiment, the object of the right side is moved before and behind for the viewpoint in each trial, and the subjects are made to judge which one seems to be more close within two teapot objects. From this result, how the depth perception accuracy changed by adding the focal blur effect, was examined.

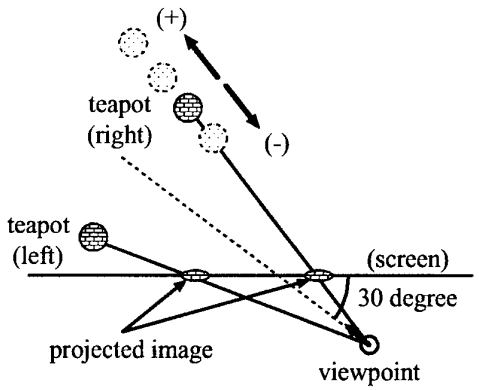


Figure 8: The accuracy of depth perception

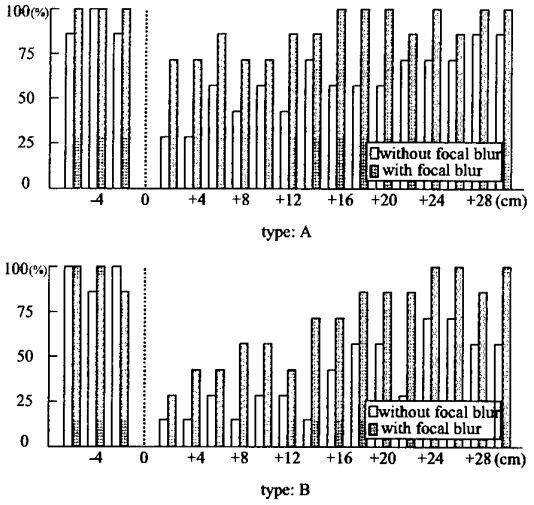


Figure 9: The accuracy of depth perception

4.3 Experiment 1: Results

A example of this result is shown in Figure 9. In this figure, the horizontal axis shows the distance at which we moved the right object before and behind, and the vertical axis shows the proportion of correct depth perception. This experiment was carried out for four subjects. These results are classified into two types as shown in Figure 9.

In case of users of type-A in Figure 9, The accuracy rate without the focal blur effect is around 50%, when the moved distance of the object is within 20 cm. With the focal blur effect, the accuracy rate is improved around 75%. The accuracy rate reaches 100% by the case. On the other hand, the accuracy rate of the type B is worse than type A as shown in Figure 9. There are many cases in which the accuracy rate is around 25% without the focal blur effect. The accuracy rate is 50% or less by the case even if the focal blur effect is added.

The cause of these differences is regarded as mainly user's individual difference. However, It was proven that the more accurate depth perception can be carried out in all subjects by using the view-dependent focal blur. This result shows that the focal blur effect reduces the depth perception error by the oblique screen.

4.4 Experiment 2: Evaluation of Reality

In this experiment, we examined the reality of an immersive virtual environment with view-dependent focal blur effect. We prepared three kinds of immersive virtual environments shown in Table 2 . The first one is a virtual environment with the view-dependent focal blur. The second one is with binocular stereoscopic effect and the last one is with both of view-dependent focal blur effect and binocular stereoscopic effect. Five subjects experienced above three kinds of environments, and compared two inside of them from a viewpoint of the reality and presence. The more realistic environment is scored, and the score is accumulated as shown in Figure 10. The examination was carried out at each eight times every each.

Table 2: Virtual environment for experiment 2

Env. Name	*Visual Effect*
A	Focal Blur
B	Stereo
C	Focal Blur + Stereo

4.5 Experiment 2: Results

In Figure 10, the result is classified in two groups. Three persons in five inside are in Type-I, and the remainder is in Type-II. In the Type-I, focal blur functions more efficiently than stereo effect. Conversely, stereo effect functions more better in the Type-II. In both case, a combination of focal blur and stereo is most effective. Because the individual difference is mainly included for this result, it is difficult to decide merits and demerits of focal blur and stereo effect. However, we can realize that the focal blur enhances the reality of the immersive virtual environment.

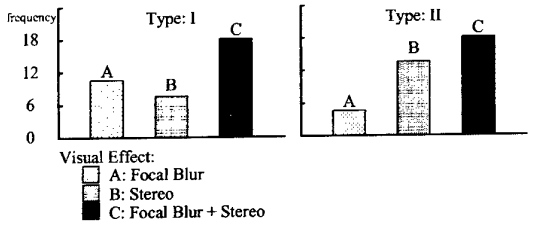


Figure 10: The reality of the virtual environment

5 Discussion

We illustrated the advantage of the view-dependent focal blur in previous sections. It can easily reduce the depth perception error and enhance the reality and presence of the virtual environments. This method, however, still contains some problems. In this section, we discuss about two subjects within the problems which is important when the method is used practically.

5.1 Optimization to individual

The results of the experiments in section 4 indicate the effectiveness of the view-dependent focal blur. This method, however, is unable to completely improve abovementioned problems like a depth perception error. It is considered that the main cause of this result is a individual variation. In the view-dependent focal blur, a degree of blur effect is controlled with some parameters. In the experiments of section 4, the parameters are dedicated by using heuristics. In order to function the method more effectively without individual variation, an optimization to individual users is needed.

There are many approach of the optimization. In this section, we introduce an approach using visual acuity. A visual acuity is also used in section 3.3. The experimental environment shown in Figure 5 is prerequisite. In this experiment, the characteristic of visual acuity in the real world and virtual environments with focal blur is measured as shown in Figure 6. By using this information, parameters which control a degree of blur are adjusted in order to simulate the characteristic of focal blur in the real world.

In a practical use of the view-dependent focal blur, the optimization is very important as a next step of this research. Therefore, we plan to wrestle this problem in the future.

5.2 Speed up

The view-dependent focal blur is very easy to implement if the accumulation buffer of OpenGL is available on conventional systems. It is suitable to enhance the reality and presence of traditional applications at a little cost.

The operations using the accumulation buffer is, however, very slow when inexpensive graphics hardwares are used. In this case, we must consider the speed up technique of the view-dependent focal blur. For example, the restriction of the blurred region is one of the approach. It is not necessary to use focal blur for the whole of the screen because our field of view is limited. If the blurred region becomes small, the view-dependent focal blur can be used without expensive graphics workstations. In recent years, VR systems based on PC (Personal Computer) have been a sudden increase. If the view-dependent focal blur is implemented on these systems, many virtual environments and its applications can take advantage of this method.

6 Conclusions

In this paper, we realized the view-dependent focal blur, and illustrated that focal blur is effective for natural and accurate depth perception in immersive virtual environments. In immersive projection displays like a CAVE system surrounded with screens, a depth perception error caused by the oblique screen is a big problem. The view-dependent focal blur is also used to reduce the effect of the oblique screen and to realize more correct depth perception.

In the future, we plan to optimize a degree of focal blur effect in order to effectively function for all users, and simulate more natural and realistic user's view in immersive virtual environments.

References

[1] Carolina Cruz-Neira, Daniel J.Sandin, Thomas A.DeFanti : "Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE", *proceedings of ACM SIGGRAPH'93*, pp.135-142(1993).

[2] Masayuki NAKAJIMA, Hiroki TAKAHASHI: "Multi-Screen Virtual Reality System: VROOM - Hi-Resolution and four-screen Stereo Image Projection System-", Proceedings of of International workshop on New Video Media Technology, pp.95-100(1997).

[3] Tetsuro Ogi: "Characteristics of Immersive Projection Displays and their Applications", *Journal of Human Interface Society*, Vol.1, No.4, pp.43-49(1999).

[4] Hirose, Ogi, Ishiwata, Yamada: "Development and Evaluation of Immersive Multiscreen Display CABIN", *The Transactions of the Institute of Electronics, Information and Communication Engineers D-II*, Vol.J81-D-II, No.5, pp.888-896(1998).

[5] Hashimoto, Takahashi, Nakajima: "A study of Focal Blur on Immersive Projection Displays", *proceedings of the 2000 IEICE general conference*, A-16-26, p.330(2000).

[6] G.Mather: "Image blur as a pictorial depth cue", *proceedings of the Royal Society of London*, Series B, 263, pp.162-172(1996).

[7] T.F.Shipley and M.L.Meyer: "Blur, contrast, and color as components of aerial perspective", *Investigate opthalmology & visual science*, Vol.40, S802(1999).

[8] Heidrich Wolfgang, Rudifer Westermann, Hans-Peter Seidel and Thomas ertl, "Applications of Pixel Textures in Visualization and Realistic Image Synthesis", *proceedings of 1999 Symposium on Interactive 3D Graphics*, pp.127-134(1999).

[9] Okajima, Takahara, Ujike and Takase, "Development of an Image-Rendering System of Focal Blur Information Synchronized with the Fixation Point, and its Perceptual Effects", *Transactions of the Virtual Reality Society of Japan*, Vol.5, No.2, pp.875-880(2000).

[10] G.Matter: "The use of images blur as a depth cue", *Perception*, Vol.26, pp.599-612(1996).

[11] Hirose, OGI and Hiratsuka: "Visual Acuity Simulation for Town Planning", *Proceedings of the Virtual Reality Society of Japan, Forth Annual Conference*, pp.27-30(1999).

[12] Paul Haeberli, Kurt Akeley: "The accumulation Buffer: Hardware Support for High-Quality Rendering", *proceedings of ACM SIGGRAPH'90*, pp.309-318(1990).

ICAT 2000

Analysis of resolution of elbow joint rotation with visual information

○ Tohru HAYASHI, Keitaro NARUSE, Hiroshi YOKOI, Yukinori KAKAZU

Laboratory of Autonomous Systems Engineering,
Research Group of Complex Systems Engineering,
Graduate School of Engineering. Hokkaido University
Nihi-8, Kita-13, Kita-ku, Sapporo, 060-8628, JAPAN
{tohru, naruse, yokoi, kakazu}@complex.eng.hokudai.ac.jp

Abstract

When we control a force-display system, some problems always happen. One of the problems is how fast and accurately it moves after it receives a control command.

One of the methods of the improvement of the problem can be given as the improvement of the accuracy of the force-display system itself, for example, improving the stiffness and the output of actuators. However, in the cases, it has the drawback of lost of the reality in other side because of it being bigger.

Another method of the improvement of the problem can be given as the improvement from an operator side. It can be said that there can be a method of the investigation of the perception ability of human being, because the error of the force-display system is accepted if an operator can't perceive it.

The level of the quality of a imaginary sense decreases if the accuracy of the force-display system is rough, and of course, the operator feels the incongruity if it decreases to less than the level of standard. However, we don't know the level of the quality which human being feels the incongruity. In other words, it isn't cleared how accurately human being perceives the accuracy of the force-display system. We can clear an acceptable range of the error of the force-display system if we can clear how accurately he perceives his own conditions and moves his own body.

In this study, we pay attention to the problem of how accurately a human operator discriminates displayed senses from the force-display system, and verify the acceptable range of the error of the force-display system from the results of the experiments. We investigate the perception sense of the position of the human operator with the purpose of control of the force-display system. There are several senses of the position perception, for example, the perception of the distance and the angle of rotation of joints and so on. In this study, we investigate the perception of the angle of rotation of an elbow joint. Then we deal with the elbow joint as intimate relation from several joints with the force-display system, and did some experiments and verified their results. We also did the combined experiment of an elbow and a wrist and verified their results to investigate relation between the joint of the elbow and that of wrist.

Key words: Virtual Reality, Force-Display, Joint of Elbow, Joint of Wrist

1.Introduction

Virtual reality enables to have an illusion to be in imaginary environment but our being in real one. It is necessary to display much more imaginary senses to sense organs of an operator as possible to have an illusion to be in the imaginary environment. But the operator feels awkward if the imaginary senses displayed were contradictory. That is displayed the imaginary senses are need to be united each other.

Many researches for the development of the system, which is able to display the sense of sight or hearing have

been reported, but compared with them, the system is able to display the sense of force and little. Without displaying the sense of force, the system would lose reality because of lack of some information (weight, hardness, shape and so on). It is necessary to display the sense of force for being lost in the imaginary environment. Recently, it is important to display the sense of force, and the research on this field is focused on. The system, which displays the sense of force on imaginary world, is called "force-display". Some of methods of force-display have been developed.

1) Master arm
2) Joy stick [1]
3) Wire [2]
4) Wearable hand (glove)[3]
5) Wearable arm [4]

But, each system has the problem need to improve.
1) Too large-scale
2) Have a limit of degree of freedom
3) Small movable range
4), 5) Unable to display weight of imaginary object

Although, the research of virtual reality with force feedback have been done with taking advantage of their features.

In this study, our purpose is to develop a light and compact system, so that operators lost in imaginary environment. However, this system cannot be enough rigidity, force and speed of response due to the design specifications. The defects bring out error to this system. However, it has not been cleared how human perceive this error. If we find the accuracy of the perception of speed and force of a human sense organ, we see allowable quantity of the error of the system.

Many researches have been done with human sense for developing making an agreeable imaginary environment. Yoshizawas[5] pays attention to the fact that a human being judges it only by the limited information of a difference of sight of both eyes when the depth of the solid image is perceived. They cleared that when each is separately exists, a human being can perceive an actual object and an imagination object. Ifukubes[6] turned fixed quantity of the error that a somato sensory system could be permitted and utilized for development virtual reality system. They cleared that the recognition of human being of the deviation in the front direction was more difficult than that in the back direction. Ishikawas[7] did the experiment of the adaptation between the information on the sight and touch. As a result, they cleared that senses were unified well by sight's indicating touch movement. Kurokawas[8] investigated how human movement changes when a sight target changes on a high-speed location movement, and cleared that it isn't a little influence when an angle is greatly being adjusted, but a big influence appears when it changes to a grade to adjust small from the grade when an angle is greatly being adjusted.

In this way, it can't be said that a human sense and the ability which human unifies them are perfect. Then we investigated the error of human sense of spatial position to see allowable quantity of an error of a virtual reality system. We focused on the angle of the rotation of an elbow. Then, We did an experiment to investigate relations with it and the information on the sight and so on.

This paper is organized as follows: in section 2, we investigated the angle perceptible resolution of an elbow joint in different conditions. Then we model an error mechanism of the elbow joint. Section 3 shows the control scheme of an angle in the force display system using the allowable error of the elbow joint, followed by the control results. Section 4 shows the case of a wrist joint. Section 5 shows the analysis of the combined case of the elbow and a wrist joint. This paper is concluded in Section 6.

2.Experiment 1: Analysis of Allowable Error Resolution of Elbow Joint

In Experiment 1, we measured an angle of the rotation of an elbow when examinee rotated his elbow, and investigated the error between the measured angle and target one. The experiment procedure is done as follows.

A total of 5 students (4male, 1female) participated in the experiment. The mean age was 23.4 years (range: 21 to 25). All examinees set an angle measurement device (which is developed for measuring angle of the rotation of the elbow) to their left-arm, rotated their elbow to several targets of angle with their spontaneous timing.

We set up following conditions on this experiment.
1) Set seven targets (0, 15, 30, 45, 60, 75, and 90 degree) at random
2) Rotated elbow 50 times each target

3) Rotated elbow with and without visual information
4) Rotated vertically without fixation of joint of shoulder (Figure 1), vertically (Figure 2) and horizontally (Figure 3) with fixation

Because, 1), 2) we need many data of each angle to compare average of all examinees' data with personal and to investigate standard deviation of measurement result of examinees' and personal.
3) To investigate the influence of the visual information.
4) To investigate the influence of the way of rotation and posture.

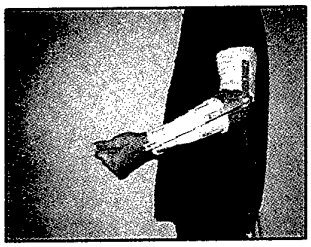


Fig. 1 Experimental Setup (1)

Fig. 2 Experimental Setup (2)

Fig. 3 Experimental Setup (3)

Figure 4 and Figure 5 are the average of the measurement angle of the whole examinees in each condition (vertically without fixation, vertically with fixation, horizontally with fixation). Figure 4 is the case of the experiment of the rotation of the elbow with the visual information. Figure 5 is the case of without the visual information. The vertical axis in the graph is a measurement angle, and the horizontal axis is the target angle.

From Figure 4, in the case of with the visual information, measured angle was almost equal to the target on each target degree, though they exceeded a little angle. On each condition, (rotated vertically without the fixation of joint of shoulder, vertically and horizontally with the fixation) an influence by the difference in the condition isn't seen in the measurement result. From Figure 5, the measurement angles were greater than the angle of the target in all the angles except for 0 and 90 degree. Because we thought that 0 and 90 degree are thought to be comparatively easy to distinguish for the human being [9], it was natural that the measured angle was almost the target on each target degree on 0 and 90. Except for 0 and 90 degree, the case when the examinees measured without the visual information have a bigger error of the perception of the angle than the case when the examinees measured with the visual information in all the angles. The error of the perception became biggest in 15 degree, and it decreases gradually to 75 degree. The point to which it should pay attention is that a difference appears in the case of the fixation of the shoulder joint and no fixation. The case when it was moved without fixation was a bigger error of the perception of the angle than the case when it was moved with the fixation. From the above, when human being rotate his elbow with visual information, the sense of the space position resolution of the elbow joint doesn't take an influence by the posture, but when without the visual information, the sense of the space position resolution of the elbow joint loses correctness, and it knows that an influence is taken in the posture and the way of the rotation of the joint of the elbow as well.

Figure 6 and Figure 7 are the average of the standard deviation of the measurement angle of the whole examinees in each condition. Figure 6 is the case of with the visual information. Figure 7 is the case without the visual information. The vertical axis in the graph is a standard deviation, and the horizontal axis is a target angle.

This result is obtained in the same way as the case of the measurement angle. The case when it was moved without the visual information was more difficult to rotate to the examinees' own target than the case when it was moved with the visual information. However, about the posture, unlike the case of the angle measurement, the fixation of the joint of the shoulder joint has a greatly influence in the target angles from 15 to 60 degree on both conditions with and without visual information.

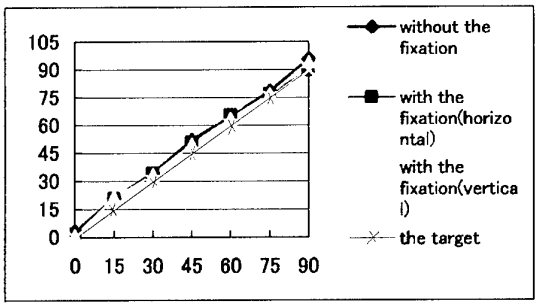


Fig. 4:Average of the measured angles with the visual information among five examinees for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

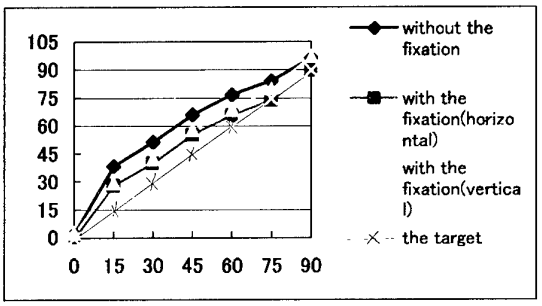


Fig. 5:Average of the measured angles without the visual information among five examinees for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

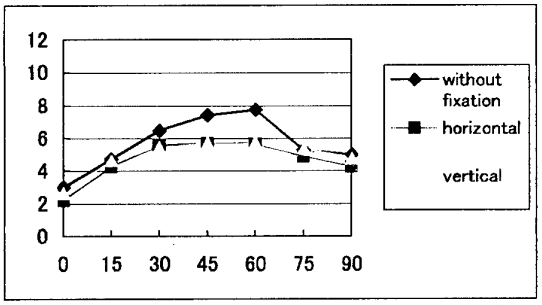


Fig. 6:Average of the measured standard deviation with the visual information among five examinees for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

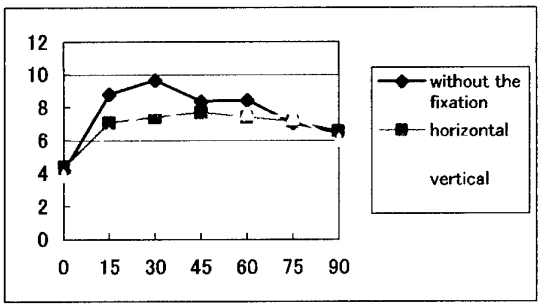


Fig. 7:Average of the measured standard deviation without the visual information among five examinees for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

Here, the cause that the errors of the rotation of the elbow become as big as the little angle is verified. It can be regarded about a human arm as Y and Z (Figure 8), and the change rate of θ due to the slight change of X by the rotated elbow was examined from equation (1).

The vertical axis in Figure 9 is K (equation (3)) ,which is a term to show a change rate of equation (2), and the horizontal axis is θ. We investigated how much the change rate of θ changed when Y and Z were supposed an upper arm part and a former arm part and the ratio of the length was changed.

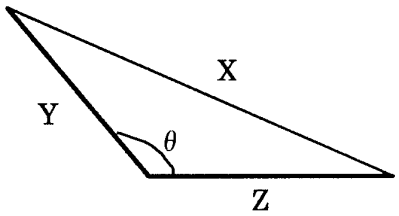


Fig 8: Model of imaginary arm

$$\Delta\theta = \left(Y^2 + Z^2 - 2XY\cos\theta\right) \cdot \Delta X / XY\sin\theta \quad \text{..... (1)}$$

$$K = \left(Y^2 + Z^2 - 2XY\cos\theta\right) / XY\sin\theta \quad \text{..... (2)}$$

Figure 9 shows that as the ratio of the former arm part and the upper arm part becomes big, the change rate of θ becomes big, and that the change rate of θ is as high as θ is small, when X changed slightly. It knows that the angle of the rotation of the elbow changes greatly, when an arm is moved only a little, as the angle of the rotation of the elbow is small, and as a former arm part mores longer than the upper arm part. Figure10 shows the value, which was got by multiplying the value of K by the value of each angle. This is similar to the result of this experiment. So, it can be considered that this change rate of θ has relations with the error of human sense of spatial position.

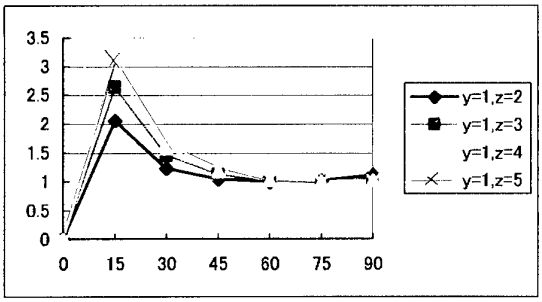


Fig. 9:Change of the sensitivity for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

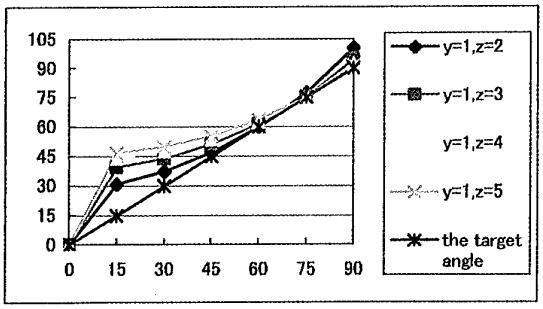


Fig. 10: The graphs which are multiplied target angle and change of sensitivity for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

3. Experiment 2: Angle Control Utilizing Angle Perception Error

In Experiment 2, we displayed imaginary arm on the screen of HMD (Head Mounted Display) to the examinees (Figure 11). For the elbow, the angle of the rotation is different from that of the actual one is indicated on the screen. Then the quantity of the error was increased gradually, and examined which condition to feel a sense of incongruity for examinees. The way of the presentation of the angle is following. From the result obtained from the experiment 1 without the visual information on each condition, the graph interpolated between the 6th-type crossing the origin like equation (3). For example, Figure 12 is result of experiment 1 (no fixation, without visual information) and the graphs (which made by making the difference between the measured and the target angle change) before interpolation in the sixth-type equation. The reason why we complemented the graph in the sixth-type is the reason is to connect the measured six points smoothly.

Among 0 and 90 degrees, a target angle was used for getting the change of the measurement angle, because they don't take the influence of the visual information and posture so much. By this, the angles' connections with the front and back become smooth. And we could make it that the environment in which a sense of incongruity was little was prepared for the examinees.

$$Y = \alpha X^6 + \beta X^5 + \gamma X^4 + \varsigma X^3 + \xi X^2 + \zeta X \quad \text{..... (3)}$$

A total of 4 students (3 male, 1 female) participated in the experiment. We asked the examinees if they feel incongruity as displayed a magnified angle of the equation with a certain degree.

Fig. 11: Experimental Setup (4)

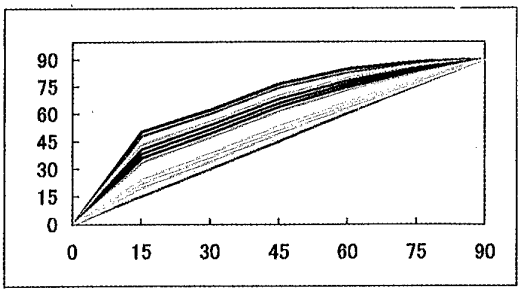


Fig. 12: Examples of the approximated formula of Fig. 5

— Examinee A —
without fixation···1.4times
with fixation (vertical)···1.1times
with fixation (horizontal)···1.4times

— Examinee B —
without fixation···1.4times
with fixation (vertical)···2.1times
with fixation (horizontal)···2.0times

— Examinee C —
without fixation···1.5times
with fixation (vertical)···1.8times
with fixation (horizontal)···1.2times

— Examinee D —
without fixation···1.4times
with fixation (vertical)···1.2times
with fixation (horizontal)···1.5times

Fig. 13: The results of Experiment 2

Figure 13 is the results that we obtained on this experiment, when we asked examinees whether it felt the sense of incongruity against the movement of the imaginary arm on the screen. On the condition without fixation, the average of feeling a sense of incongruity

was 1.425 times. Though the error of the angle of the rotation of the elbow is biggest on this condition in experiment 1, the examinees didn't feel a sense of incongruity against the angle in spite of greatly bigger one than that of the average on experiment 1. In addition, it was about the same result in the whole examinees. Without the fixation of the joint of shoulder, it can be said that the sense of the space position resolution loses correctness. On the condition with the fixation, when rotated the elbow horizontally, the average of feeling a sense of incongruity was 1.55 times. When rotated the elbow vertically, the average was 1.525 times. On this condition, the considerable difference was seen between the examinees. However all examinees didn't feel the sense of incongruity against the angle on experiment 1 on all conditions. It is much easier to set up the illusion by being lost the correctness of the space position resolution by thinking of an imaginary arm the examinees' one.

4.Experiment 3: Wrist Angle Analysis

In Experiment 3, we measured angle of the rotation of the wrist when an examinee rotated his wrist, and investigated the error measured and target angles in the same way as the case of Experiment 1. The following explains experimental methods.

1 student (male) participated in the experiment. His age was 26 years. The examinee had the lever of the experiment device (Figure 14), which was developed for measuring angle of the rotation of the wrist by making use of three-dimensional position sensor with his left hand. He rotated his wrist to several targets of angle with his spontaneous timing. We set up following conditions on this experiment in the almost same way as Experiment 1.

1) Set four targets (0, 20, 40, and 60 degree) at random
2) Rotated wrist 50 times each target
3) Rotated wrist with and without visual information
4) Rotated horizontally

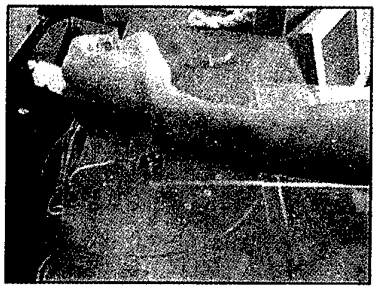


Fig. 14: Experimental Setup (5)

Figure 15 shows the average of the measurement angle of the examinee in two conditions (in the case of with and without visual information). The vertical axis in the graph is a measurement angle, and the horizontal axis is the target angle.

From Figure 15, in the case of with the visual information, the measured angle was almost same as the target degree, though they exceeded a little target angle in the same way as the case of Experiment 1. And in the case without the visual information, the measurement angles were greater than the target angle in all the angles except for 0 and 60 degree. Because we thought that 0 degree is thought to be comparatively easy to distinguish for the human being in the same way as the case of Experiment 1,and about 60 degree, it can be thought that a big error doesn't happen easily for the reason of nearly limit of the angle of the rotation of wrist. The case when examinee measured without the visual information occurred a bigger error of the perception of the angle than the case when examinees measured with the visual information in all the angles. The error of the perception became biggest in the 15 degree, and it decreases gradually to 60 degree.

From the above, when human being rotate his wrist with the visual information, the sense of the space position resolution of the joint of the wrist doesn't take an influence, but when without the visual information, the sense of the space position resolution of the joint of the wrist loses correctness in the same way as the case of joint of the elbow in Experiment 1.

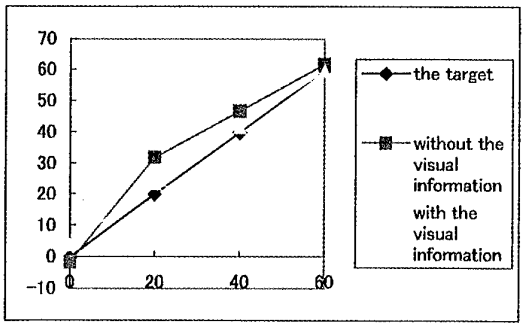


Fig. 15: Average of the measured angles with and without the visual information of the examinee for the four target angles 0, 20, 40 and 60 degree

Figure 16, is the average of the standard deviation of the measurement angle of the examinee with and without the visual information. The vertical axis in the graph is a

standard deviation, and the horizontal axis is the target angle.

As a result, in the same way as the case of the measurement angle, the case when it was moved without the visual information was more difficult to rotate to the examinee's own target than the time when it was moved with visual information. This result is also in the same way as the case of the joint of the elbow in Experiment 1.

From the above, it can be said that characteristics of the joint of the wrist is similar to that of the joint of the elbow.

However, it is necessary to do some experiments further in order to prove these results because we didn't do the experiments about more than one examinee and other postures in Experiment 3. We will do some experiments further from now on.

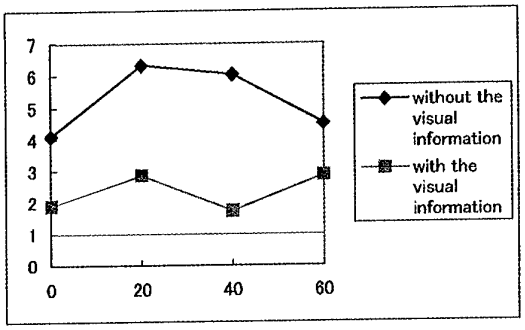


Fig. 16: Average of the measured standard deviation with and without the visual information of the examinee for the four target angles 0, 20, 40 and 60 degree

5.Experiment 4: Combined of Elbow and Wrist

In Experiment 4, we measured an angle of the rotation of the elbow when examinees rotated their elbow with fixation of the wrist in a certain fixed. In other words, we did experiment which is the same as Experiment 1 with fixation of the wrist. The following is to say explain experimental methods.

1 student (male) participated in the experiment. His age was 26 years. He is the same person in Experiment 4.

The examinee had the lever of the experiment device (Figure 17), which was developed to measure the angle of the rotation of the wrist by making use of three-dimensional position sensor with his left hand. And he rotated his elbow (with fixation of the wrist in a certain fixed) to several targets of angle with his spontaneous timing.

We set up following conditions on this experiment in the almost same way as Experiment 1

1) Fixation of the wrist (0, 20, 40, and 60 degree)
2) Set seven targets (0, 15, 30, 45, 60, 75, and 90 degree) at random
3) Rotated elbow 50 times each target
4) Rotated elbow with and without visual information
5) Rotated horizontally

Fig. 17: Experimental Setup (6)

Figure 18 shows the average of the measurement angle of the examinee in each condition (in the case of fixation of the wrist) with and without the visual information. The vertical axis in the graph is a measurement angle, and the horizontal axis is the target angle.

From this result, the case when it was moved without the visual information was more difficult to rotate to the examinee's own target than the time when it was moved with the visual information in the same way as the case of Experiment 1, though the difference isn't large.

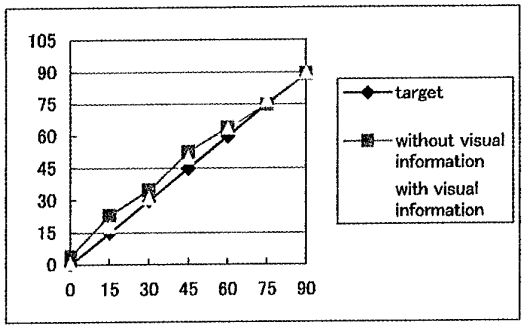


Fig. 18:Average of the measured angles with and without the visual information for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

From Figure 19, in the case of without the visual information, the measured angle was almost same as target angle in each target though they exceeded a little

target angle in the same way as the case of Experiment 1. The point to which it should pay attention is that the more bigger the angle of the joint of the wrist was, the correctly measured angle was. As the reason, it can be thought that the ratio of the former arm and the upper arm changed by the rotation of the joint of the wrist (cf. Figure 9).

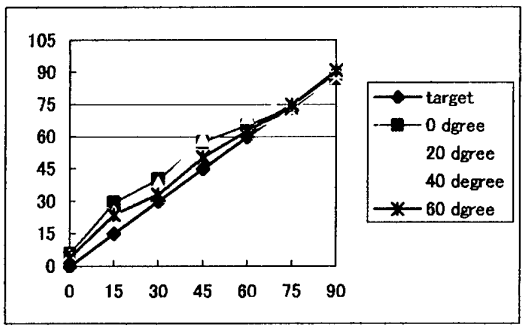


Fig. 19:Average of the measured angles with and without the visual information for the seven target angles 0, 15, 30, 45, 60, 75 and 90 degree

However, since we didn't do the experiments for more than one examinee and other postures in this Experiment, as the case of Experiment 3, it is necessary to do some experiments further in order to prove these results. As the future works, we have to do some experiments further from now on.

6.Conclusion

The following things cleared by the result of the experiment on this research.

1. The force-display system lets an operator have an illusion by operating the visual information.

2. There is the difference on the quantity of illusion due to the difference and the change of the operator's posture.

3. There is the difference in the quantity of illusion between individuals.

4. The force-display system is allowed to have bigger the error to display a small angle than that of a big angle.

5. The force-display system is allowed to have an error when it displays a same angle several times.

6. The force-display system lets an operator have an illusion by display bigger angles than an actual angle of the elbow joint.

7. The force-display system lets an operator have an illusion by display bigger angles than an actual angle of the wrist joint.

Reference

[1] H.Iwata, Y.Asada: "Augmented Reality with Force Feedback", Proceedings of The 1st Virtual Reality Society of Japan Annual Conference, 1996, pp.13-16

[2] H.Kushida, M.Ishii, M.Sato: "Multimodal Human-scale Virtual Environment –Big SPIDAR－", Proceedings of The 2nd Virtual Reality Society of Japan Annual Conference, 1997, pp.131-132

[3] H.Takahashi, S.Kanai: "An Application of the Artificial Reality to the Free-Form Surface Modering", Journal of the Robotics Society of Japan,1992, Vol.10, No.7, pp.898-902

[4] A.Nakai, Y.Kunii, H.Hashimoto: "Design of Force Sensor System of Arm Type Human Interface", Proceedings of The 15th the Robotics Society of Japan Annual Conference, 1997

[5] K.Sawada, E.Ishikura, T.Kawahara, T.Yoshizawa: "Hand Movement Reached to Real Object and/or a Virtual Object", Proceedings of The 1st Virtual Reality Society of Japan Annual Conference, 1996, pp.235-236

[6] Y.Li, C.Wada, S.Ino, T.Ifukube: "The role of tactile feedback on depth perception", Proceedings of The 3rd Virtual Reality Society of Japan Annual Conference, 1998, pp.265-268

[7] M.Akamatsu, M.Ishikawa: "Study on Visuo-Tactile Sensory Integration in Shape Perception", Bio-mechanism 10, pp.23-31

[8] K.Mishima, T.Kurokawa, H.Tamura: "Bimodel control of Fast Forearm Movements through Central Commands", Jounal of The Society of Instrument and Control Engineers, 1982, vol.18, no.4, pp.400-406

[9] T.Hayashi, H.Kita, K.Naruse, H.Yokoi, Y.Kakazu: "Analysis of resolution of elbow joint with visual information", Proceedings of The 4th Virtual Reality Society of Japan Annual Conference, 1999, pp.195-196

ICAT 2000

An Orthopedic Virtual Reality Surgical Simulator

Ming-Dar Tsai[1], Shyan-Bin Jou[1], Ming-Shium Hsieh[2]

[1] Chung Yuan Christian University, Institute of Information and Computer Engineering
Chung Li, 32023 Taiwan, R.O.C.
[2] Taipei Medical College Hospital, Orthopaedics and Traumatology Department,
252, Wu Hsing Street, Taipei, Taiwan, R.O.C.
{tsai@ice.cycu/jou@earth.ice.cycu/shiemin@mail.tmc}.edu.tw

Abstract

This paper describes a highly interactive virtual reality orthopedic surgery simulator. The simulator can section, reposition and join volume-represented structures. By these functions, the simulator allows surgeons to use various surgical instruments to operate on virtual bones for simulating every procedure of complex orthopedic surgeries.

Key words: Virtual reality, Orthopedic surgery, Volume based visualization and simulation

1. Introduction

Orthopedic surgeries usually involve complex geometry and topology changes in bone morphology. Current training methods for interns and residents in teaching hospitals do not adequately raise spatial perception about the geometry and topology changes of bone morphology. Two reasons for the inability are trainees can only observe an operation before he participates in surgery and preoperative rehearsal usually involves 2D paper surgical simulations based on X-ray images. Orthopedic visiting doctors may also fail in real operations (e.g. 10%~20% for high tibia osteotomy [1,2] and 5%~15% for anterior fusion of the spine [3,4]) because there exists geometric and topology failures in bone morphology. These failures include false sections on bones, poor contact surfaces, inappropriate size and shape of bone graft and improper reduction position. The reason is considered the visiting doctors can only use 2D paper simulations to rehearse and confirm surgical plans.

The application of virtual reality (VR) to surgical training gives a more realistic human machine interaction than traditional 2 dimensional simulations and has already become a useful surgical planning and training tool. Several VR surgical simulators have been developed to provide detailed information regarding simulated tissues, tools and actions of surgeons [5].

VR simulation systems provide virtual environment by rendering a surface model that may be reconstructed from video data (for simulating endoscopic or laparoscopic surgery [6]), X-rays (for leg surgery [7]), or synthetic surfaces (for ophthalmic surgery [8]). However, the surface models are difficult to be employed to compute topology changes because of no interior information. Contrast to surface models, a volume (stack of 2-dimen-sional grayscale images) model represents a body as regularly partitioned cuboids (voxels) is suitable to reveal relations between tissues (with resolution limits but no projection errors [9, 10]) and simulate surgeries with topology changes.

Many excellent algorithms have been developed for visualizing a volume. For example, tissue surfaces can be well approximated by hundreds of thousands of triangulated isosurfaces [11]. These isosurfaces can be quickly rendered by current PC platforms. Some orthopedic simulators have employed the isosurfaces of anatomic structures to generate a virtual environment for training arthroscopy [12] and fixing [13]. However, manipulating the isosurfaces can not simulate orthopedic surgeries involving topology changes.

Surgical simulation algorithms usually manipulate voxels directly to simulate surgeries especially the ones with topology changes on structures. For example, most commercial imaging systems use a simple method of manipulating voxels, a cut-away operation to remove the voxels of one side of a cutting plane for removing obscurations. However, many cut away operations must be used for simulating a procedure (even a simple section) of orthopedic surgeries. Two approaches of manipulating voxel-represented objects have been discussed. One uses 2D pointer array to record the result of a series of cut away operations [14]. Then, adding the translation to the voxels in the lists can simulate the visual effects of translating a structure. Another approach extends the contents of each voxel, for example, 6 links to represent relations between a voxel and its six face neighbors. Adding or deleting links are easily implemented for local manipulations such as cutting or joining two objects [15]. However, link

additions and deletions are time and memory consuming. Moreover, global manipulations such as repositioning objects or joining separate objects are difficult although they are also necessary in orthopedic surgeries.

This paper describes a VR orthopedic surgery simulation system manipulating volume data. To reposition a voxel-represented object, voxels are distinguished in a sub-tissue level. A structure code is assigned to achieve this purpose. By searching the voxels with the same structure code, all voxels of a structure can be traversed to manipulate (remove, reposition or assign as another structure to join). We have presented several algorithms that manipulate the structural voxels in 3D ways to simulate surgical procedures including cutting, identifying, removing and repositioning a structure, joining two structures into one, and testing collision during moving a structure. Combining these procedures, our system can provide surgical functions that operate a 3D image (virtual patient) as actual procedures on a real patient and ensure the accuracy of anatomic morphology in interactive responses. Through the 3D visual input and output environment, spatial perception of every procedure and its result give more effective simulations.

2. System Overview

The system was first reported in 1996 [16], and has since been modified and improved. The software is implemented in C++ (Visual C++ ver 5.0) under Microsoft Windows of a PC platform, and uses the OpenGL libraries to render isosurfaces without special graphics hardware. The PC must be equipped with a shutter glass and a tracker.

Figure 1 shows the system architecture. A user wears a shuttle eyeglass to observe stereographic images and uses a surgical instrument attached with a 6 dimensional degree tracker to simulate surgical procedures. The system includes an interface module, volume conversion module, isosurface reconstruction module, rendering module, and simulation module.

2.1 Interface module

The interface module provides virtual instruments and selectors including menus and data slide-bar. Using the menus, the user can choose a volume to simulate, determine a simulation function to operate, and input bone grafts and prostheses that have been designed by an AutoCAD system and change parameters of the shading model about light and material properties. Through the slide-bars, the user can easily change (slide) perspective conditions including viewing positions and angles, disparities of stereographic images to choose suitable ones.

The tracker is attached to one end of a surgical instrument to simulate a virtual instrument. Based on the position and attitude of the tracker and the shape data obtained from the instrument, the system can compute spatial data for the virtual instrument. Using the spatial data, the system can render the instrument to obtain its 3D image and compute the intersections between the instrument and the volume for simulating surgeries. The system currently provides the following virtual instruments: bone saw and osteotome for sectioning bone, virtual plate and staple for fixation, virtual dissector and currector for removing tumors, and virtual hand for moving bones, bone grafts and prostheses. The tracker is also used as a positioning instrument that partitions a volume into several subvolumes for the convenience in rendering tissue surfaces.

2.2 Data conversion module

For manipulating voxel-represented objects, every voxel is assigned three 6-bit distance-levels to simulate tissue surface changes, six 1-bit face codes indicating whether the voxel faces are on the boundary and one byte indicating a tissue type and structure number. A total of 4 bytes of memory are used for each voxel. Bone grafts and prostheses are designed by the AutoCAD system first, then converted to voxel-represented objects.

2.3 Isosurface reconstruction module and rendering modules

In contrast to thresholding techniques that determine a sample point on a tissue surface (isosurface) by one over-threshold voxel and one under-threshold voxel, one distance-measured voxel can determine a sample point [17]. Therefore, the three distance-levels are interpreted as three sample points on the three main axes respectively. Our system then use the marching cube algorithm that employ the sample points on the main axes to reconstruct triangulated isosurfaces [11].

2.4 Simulation module

The "sec tion" funct function first int erpret s the p attitudes of a tracker as swept sectioning surfaces, computes distance-levels for sectioned boundary voxels, and assigns a structure code to the voxels. The "recogniti on" functi on function identifies a separates uses an efficient 3D seed and flood algorithm to assign the voxels the same structure code inside a closed boundary composed of voxels with the same structure code (sectioned boundaries) and voxels of different tissues (natural boundaries). Unlike straightforward seed and flood algorithms that put six neighbors into a stack for recursion, voxels along some axis are directly computed and not stacked in the algorithm [18]. Therefore, voxels for recursion are considerably

reduced.

The "rem Tremoval" funct function assigns all voxels of to be air voxels that can also be implemented by the 3D seed and flood algorithm. The "fusion" function re-recognizes one anatomic structure (separate bones, prostheses or bone grafts) from another and joins them together. The structures may contact each other and no new structure voxels are generated. New structural voxels may be generated to help in the fusing process. In this situation, this function generates closed boundary voxels between two user-specified curves on fusing structures. The system then recognizes the voxels inside the new boundary voxels with the old structures as one structure.

The "col Ecdlisiontest" function det function dete bones, prostheses, vessels and nerves. He proposed an efficient collision detection method that maps all objects into a map of regular cells, then detects collisions if objects occupy other object's spaces [19]. This grid intersection method was not adopted to detect collisions in our system because other functions are implemented during the collision test. One such function determines the distance between structures when a structure is placed onto another structure in a "fusing" simulation H The other function assigns a structure code to traversed soft-tissue voxels in a "healing" simulatio he aling" si m an efficient ray traversal algorithm to detect collisions (whether bone or nerve voxels exist on the path of a moving anatomic structure or surgical instrument). This algorithm is the most efficient because it has the fewest additions and comparisons [20].

The "rep Treposition" function function translates structure to another position by first implementing a Ecdlision test" to detect co to detect collisions, t structure into a series of stacks and clearing the structure by the seed and flood algorithm before popping the structure to the new position. The three components of the translating vector are not limited integers. This means the system allows an unaligned translation that usually occurs when a structure is moved along the slice direction.

3. Results

In the following, we demonstrate two simulation examples operated by a visiting doctor. The CPU times were obtained under implementing on a PC with a Pentium-III 800 MHz CPU and 256 Mbytes of main memory.

3.1 Arthroplastic simulation example

In the simulation of arthroplasty operations, the user sections (using the "section" Usectio) bones until one or more anatomic structures are separated (and thus recognized by the "recognition" funcTe) from the skeleton. The user may remove (using the "removal" function) the structures to correct the skeletal morphology, to accommodate the prosthesis or in the case where the structures are abnormal bones. A prosthesis is used to replace a removed joint. The surgeon may reposition the structures (using the Trepositi on" functio) to correct the skeletal morphology, and then fix the structures and fuse (using the "fus Hfusing" fun) them into the skeleton.

Figure 2 shows the image rendering results of an example knee arthroplasty operation that was performed to replace a destroyed joint and correct a malposition of the tibia. The volume was constructed with 24 CT slices at a 256×256 resolution. However, we enlarged the volume as 35×256×256 resolution for manipulating a user-input prosthesis. The computation time was 2.2 seconds to reconstruct the bone isosurfaces for the whole volume and 0.29 seconds to obtain a 3D image with this system. After the prosthesis was input into the volume, the isosurface reconstruction time becomes 3.2 seconds for the bone and prosthesis isosurfaces and the rendition time becomes 0.42 seconds.

Figure 2(a) shows the proximal tibia being sectioned by the saw. The interface slidebars for determining the parameters of the various perspectives and menus for determining a simulation function are also shown in the left top and bottom respectively. Figure 2(b) shows the results after two flat sections on the femur and tibia respectively, followed by recognition and removal of a near flat bone fragment of the femur and a wedge-shaped fragment of the tibia. A hand (a virtual instrument) began to reposition the tibia. Figure 2(c) shows the tibia was repositioned to correct the mal-position. A vertical bone fragment was sectioned away so that the femur can accommodate the posterior of the prosthetic femur. The virtual hand was removing the bone fragment. Figure 2(d) shows a vertical bone fragment and an oblique one were sectioned away such that the femur can accommodate the anterior of the U-shaped prosthetic femur. An oblique section on the posterior of the femur for accommodating the U-shaped prosthetic femur and an oblique section on the patella for accommodating the prosthetic patella were then sectioned away.

Figure 2(e) shows the prosthesis has been recognized (by recognition functions) as three separate structures: a curved femur part, disk-like tibia part and dome-like patella part. The tibia part has been repositioned for insertion on the tibia by the virtual hand. This figure also shows the dome of the prosthetic patella can well slide inside the groove of the prosthetic femur and the prosthetic tibia also well matched to the tibia plateau. The three prosthetic parts are good choices for working the knee functions. Figure 2(f) shows the U-shaped prosthetic femur has been repositioned for insertion on

the femur. The prosthetic patella has also been repositioned for the insertion on the patella. However, we can not observe it well because it is almost hidden by the patella. The prosthetic femur well accommodated to the femur, therefore the previous sections on the femur were appropriate. The U-shaped curve of the prosthetic femur can also slide well inside the grooves of the prosthetic tibia. Therefore, the prosthetic femur and tibia are considered well positioned.

The simulation example provides an anatomical demonstration that the knee arthroplasty can correct the mal-position of the tibia, accommodate the tibia and femur to fit the prosthesis and insert the prosthesis into the correct position. The complex changes in bone morphology involved in this surgery were well simulated by our system. The results of every procedure can be thoroughly demonstrated with a high-quality 3D image. Table 1 shows the computer response times for the simulations involved in the knee arthroplasty. A complete simulation is defined as including completion of the specified function, reconstruction of the isosurfaces and rendering of the corresponding image. Because most the simulations responded in 2 seconds, we considered our system could achieve the requirement of interactive responses.

3.2 Open ostetomic simulation example

Open osteotomy is used to open bone in order to remove tumors inside the bone. Upon simulation of this technique using our system, the user sections a bone until a window structure separates. He then repositions the structure away to indicate opening the bone by using the recognition function and then the reposition function. Then, he dissects the tumor (using the section function) and removes it. The user may simulate implantation of a bone graft by inputting a bone graft and repositioning it to the tumor position. The user then finally repositions the separate window structure to the original position and fuses the window structure with the original bone together to simulate closure of the bone.

Figure 3 shows the rendering results of a knee open osteotomy for removing a tumor inside the proximal tibia. The volume was constructed with 28 CT slices at a resolution of 256×256. Figure 3(a) 256. Figure 3(a) sh a knee where the proximal tibia was being sectioned by a virtual saw. Figure 3(b) shows a window-shaped bone fragment that has been sectioned, recognized and repositioned away using the virtual hand. The area of the tumor is marked with an orange color. A dissector (indicated by a red color) is available to dissect the tumor. Figure 3(c) shows the tumor being dissected by the dissector. Figure 3(d) shows that the tumor has been removed and a graft bone (lower left corner) has been prepared (already recognized) to fill the space of the resected tumor. Figure 3(e) shows that the graft bone has been implanted and that the window-shaped bone fragment was being repositioned again to its original location. Figure 3(f) shows the results after fusing the bone fragment with the knee. The results suggest that the position and size of the window fragment is a reasonable choice for opening the knee and allowing the tumor to be completely removed. The graft bone is suitable to fill up the tumor space.

4. Conclusion and Future Work

Computed tomography (CT) or magnetic resonance imaging (MRI) scanning has become a standard procedure to reveal interior anatomies. Visualizing a volume constituted by transversal slices can ease observation of anatomies to improve diagnoses. Beyond the volume visualization, manipulating volume data to simulate deformation or topology changes of tissues during a surgery can verify surgical plans, rehearse procedures and predict prognoses.

Our simulation methods can manipulate voxel-represented bone structures to model interactions between the bones such as: cutting, fusing, repositioning, recognition, and collision testing of a moving bone. By these functions, our system can simulate complex geometry and topology changes of bone morphology for every orthopedic procedure. These capabilities are necessary to provide helpful spatial information for most orthopedic surgeries. Therefore, the simulator is useful in the preparation for many kinds of difficult surgical procedures that are often performed in the orthopedics department without putting patients at risk.

The future works can focus on improving some drawbacks of the prototype system. Improvements in the user interface can ease users to operate the system. For example, we hope to assign distinct colors to different structures for high-lighting some structure. As the arthroplasty example shows, the bones and the prosthesis are considered as the same tissue because they can be fused together. The more important prosthesis is better to be high-lighted for easy observation. The improvement in reducing response time of the system would help improving realism and ease to use. Because the system uses a volume to simulate every surgical procedure, rendering isosurfaces reconstructed from a volume to obtain a 3D image is usually computationally demanding. Use of decimation techniques to reduce triangles of isosurfaces or other volume visualization techniques such as accelerated volume rendering techniques can be tried to save the rendition time.

Acknowledgment

The authors would like to thank the National Science Council for financial support of this research under Contract No. NSC-86-2213-E-033-036, NSC-87-2213-E-033-005, NSC-89-2320-B-038-019-M08 and NSC-89-2213-E-033-070. We would also like to thank all of our colleagues at the computer research lab, the visiting doctors, residents and interns at our orthopedic department for participation in the study. We also like to thank the Radiology Department of Taipei Medical College Hospital for providing the CT and MRI data and the United Orthopedic Corporation of Taiwan for providing the prosthesis data.

References

1 J.A. Finkelstein, A.E. Gross, A. Davis, Varus osteotomy of the distal part of the femur: A survivorship analysis, J. Bone Joint Surg. Am. 78A (1996) 1348-1352.

2 H. James, Beaty, Orthopaedic Knowledge Update. In: San. Howard, T.L. Randall (Eds.), Spine. Rosemont, American Academy of Orthopaedic Surgeons, 1999, pp.561-754.

3. J.J. Regan, H. Yuan , P.C. Mc Afee, Laparoscopic fusion of the lumbar spine: minimally invasive spine surgery, a prospective multicenter study evaluating open and laparoscopic lumbar fusion, Spine 24 (1999) 402-411.

4. M. Muschik, H. Zippel, C. Perka, Surgical management of severe spondylolisthesis in children and adolescents. Anterior fusion in situ versus anterior spondylodesis with posterior transpedicular instrumentation and reduction, Spine 22(17) (1997) 2036-2042.

5. R.M. Satava, Virtual reality, the current status of the future, in: S. Weghorst, H. Siegurg, K. Morgan (Eds.) Health care in the information Age, IOS Press, Amsterdam, 1996, pp. 542-545.

6. R.M. Satava, Virtual reality surgical simulator: The first steps, Surgical Endoscopy 7(3) (1993) 203-205.

7. L. Caponetti, A.M. Fanell, Computer-Aided Simulation for Bone Surgery, IEEE Computer Graphics and Applications 13(6) (1993) 87-91.

8. I.W. Hunter, L.A. Jones, M.A. Sagar, S.R. Lafontaine, P.J. Hunter, Ophthalmic Microsurgical Robot and Associated Virtual Environment, Computer in Biology and Medicine 25(2) (1995) 173-182.

9. S. Seipel, I.V. Wagner, S. Koch, W. Schneide, Oral implant treatment planning in a virtual reality environment, Computer Methods and Programs in Biomedicine 57(2) (1999) 95-103.

10. S. Baumrind, F. Moffit, S. Curry, The Geometry of Three-Dimensional Measurement from Paired Coplanar X-Ray Images, Am. J. Orthod., 84 (1983) 313-326.

11. W.E. Lorensen, H.E. Cline, Marching Cubes: A High Resolution 3D Surface Construction Algorithm, ACM Computer Graphics, 1987, pp.163-169.

12. R. Ziegler, G. Fischer, W. Muller, M. Gobel, Virtual Reality Arthroscopy Training Simulator, Computer in Biology and Medicine 25(2) (1995) 193-203.

13. A. Sourin, O. Sourin, H.T. Sen, Virtual Orthopedic Surgery Training, IEEE Computer Graphics and Applications, 20(3) (2000) 6-9.

14. J.K. Udupa, D. Odhner, Fast Visualization, Manipulation and Analysis of Binary Volumetric Objects, IEEE Computer Graphics and Applications, 11(5) (1991) 53-63.

15. S. F. Frisken-Gibson, Using Linked Volumes to Model Object Collision, Deformation, Cutting, Carving, and Joining, IEEE Transaction on Visualization and Computer Graphics 5(4) (1999) 333-348.

16. M.D. Tsai, M.S. Hsieh, W.C. Chang, S.K. Wang,, Volume Manipulation Algorithms for Simulating Musculoskeletal Surgery, Proceedings of the Fourth Pacific Conference on Graphics and Applications, 1996, pp. 220-234.

17. G. Sealy, K. Novins, Effective Volume Sampling of Solid Models using Distance Measures, roceedings of IEEE Computer Graphics International, 1999, pp.26-32.

18. L. Feng, S.H. Soon, An Effective 3D Seed Fill Algorithm, Computer & Graphics 22(5) (1998) 641-644.

19. T. He, A. Kaufman, Collision Detection for Volumetric Objects, IEEE Computer Society Press, Proceedings of IEEE Visualization, 1997, pp.159-166.

20. D. Cohen, A. Kaufman, 3D Line Voxelization and Connectivity Control, IEEE Computer Graphics and Applications 17(6) (1997) 80-87.

Table 1 Total simulation time for knee arthoplasty simulations

	1	2	3	4	5	6	7	8	9	10	11	12
Time spent(sec)	0.95	1.24	1.98	0.96	1.95	0.93	1.88	1.97	1.75	2.05	3.20	0.93
Voxels involved	8879	11845	37365	5252	4990	4880	3985	3754	4750	19605	39630	1250

1. sectioning away (sectioning, recognizing and removing) the high-tibia; 2. sectioning away the low-femur; 3. recognizing and repositioning the tibia; 4. a vertical section on the anterior part of the femur; 5. recognizing and removing the vertical section; 6. an oblique section on the anterior part of the femur; 7. recognizing and removing the oblique part; 8. A vertical section on the posterior part of the femur then moved away; 9. an oblique section on the posterior part, then moved away; 10. repositioning the prosthetic tibia; 11. repositioning the prosthetic femur; 12. repositioning the prosthetic patella.

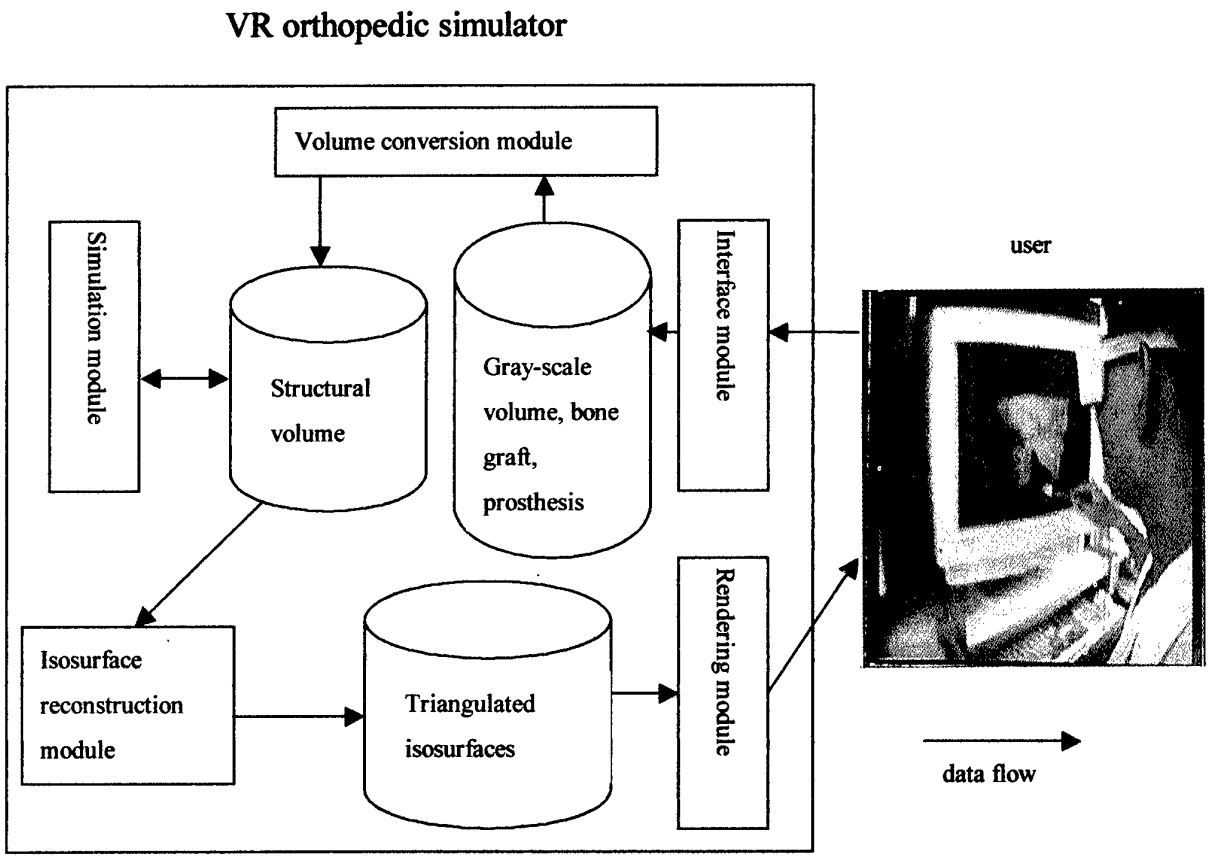
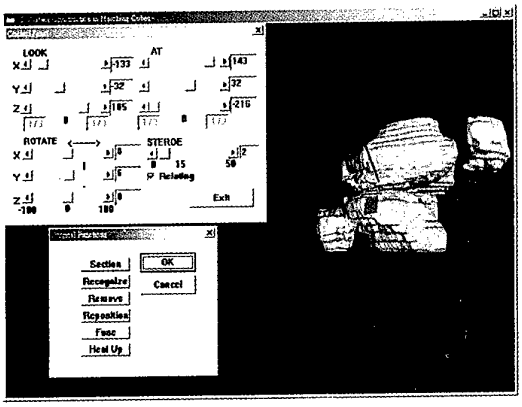
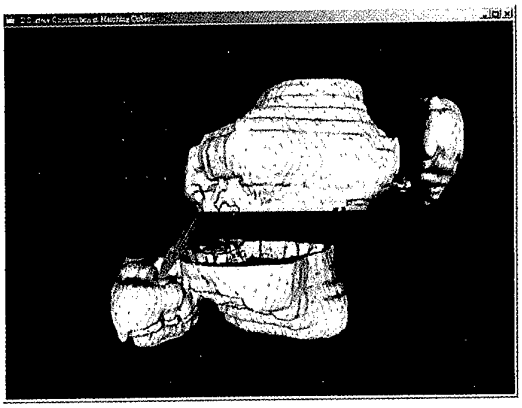


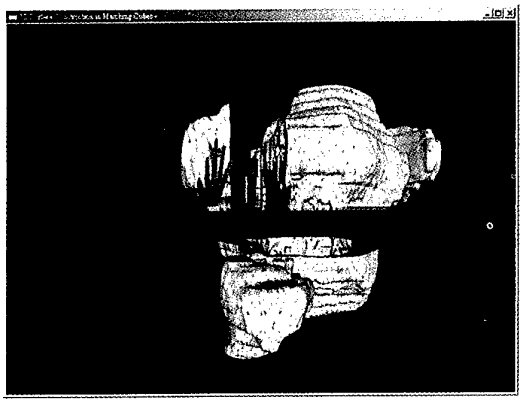
Figure 1 System architecture



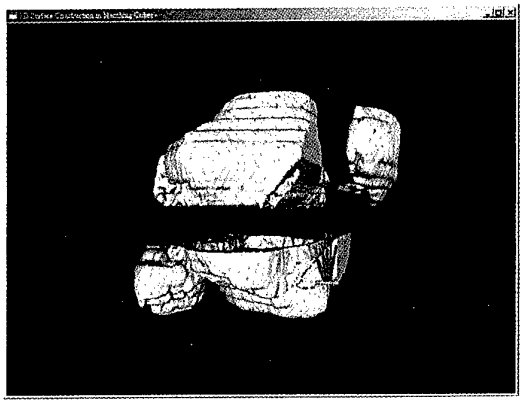
(a) Proximal tibia is being sectioned by the saw



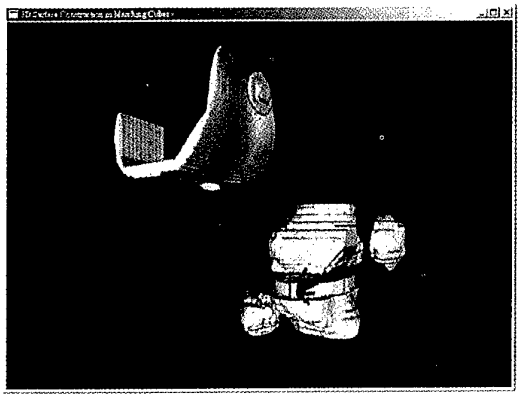
(b) Knee joint has been sectioned away, tibia is being repositioned



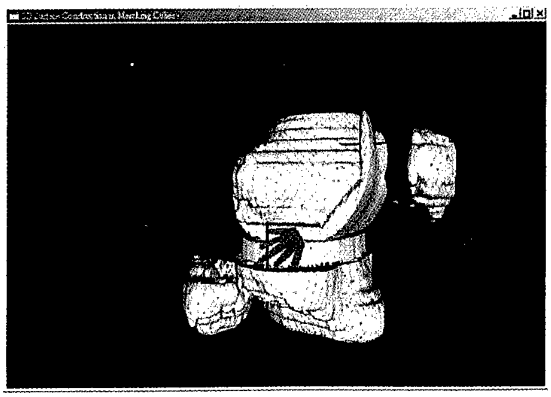
(c) Posterior femur part has been sectioned away to accommodate the U-shaped femur part of the prosthesis



(d) Anterior femur part has been sectioned away to accommodate the U-shaped femur part of the prosthesis

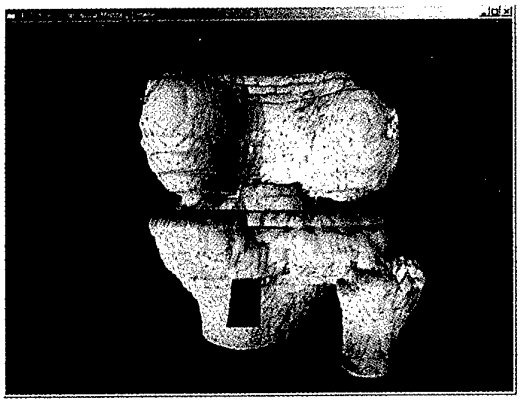


(e) Tibial part of the prosthesis has been inserted into to the knee joint

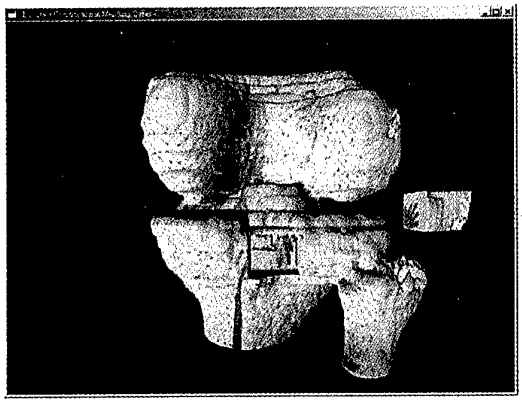


(f) Femur part of the prosthesis has been inserted into the knee joint

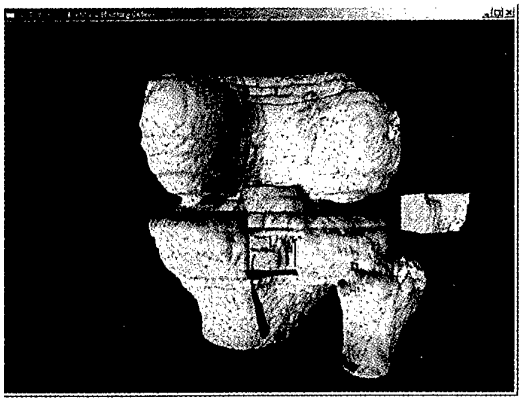
Figure 2 Arthoplasty for replacing the knee joint and correcting mal-position of the knee



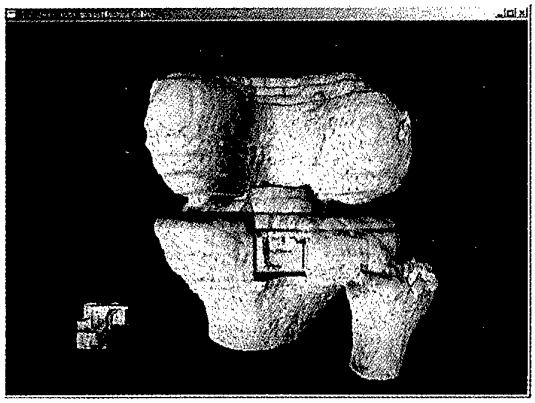
(a) Proximal tibia is being sectioned by the saw



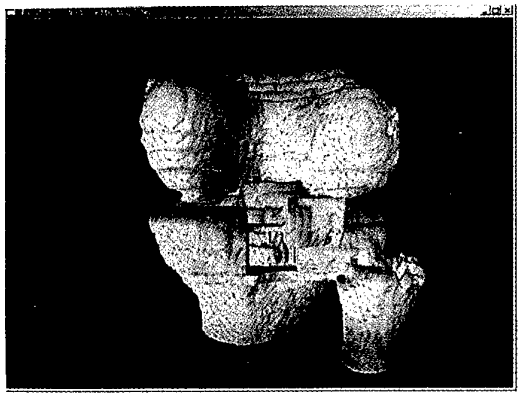
(b) Window-shaped bone fragment is sectioned and repositioned away



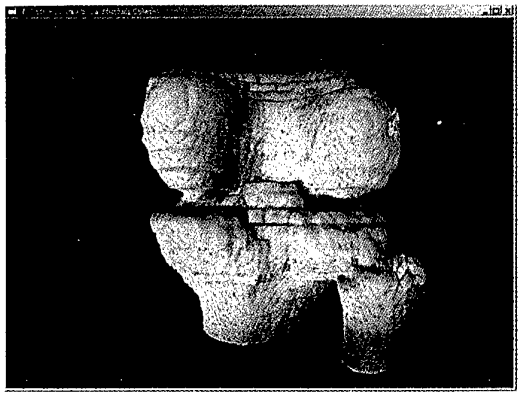
(c) Tumor is being dissected by the dissector



(d) Bone graft is prepared to fill-up the space of excised tumor



(e) Bone graft has been implanted, and the window shaped piece of bone is being repositioned to the original position



(f) Window shaped bone is repositioned and fused with the tibia

Figure 3 Open osteotomy for removing a tumor in the tibia

ICAT 2000

Animation synthesis for virtual fish from video

Hiroki TAKAHASHI, Junji HATOYA,
Naoki HASHIMOTO and Masayuki NAKAJIMA

Graduate School of Information Science & Engineering, Tokyo Institute of Technology, Japan
2-12-1 Ookayama Meguro-ku, 152-8552, Tokyo, Japan
Email: {rocky, jun, naoki, nakajima}@img.cs.titech.ac.jp

Abstract

CG(Computer Graphics) is widely used in many kinds of fields, because of the recent progress of the CG techniques, especially, in modeling and rendering techniques. In the field of VR(Virtual Reality), CG techniques are very important in order to make two dimensional or three dimensional virtual world. It is, however, very difficult to represent virtual creature with locomotion and behavior.

In this paper, estimation method of fish position and posture from video using object matching technique is proposed. Then, some basic behaviors are segmented from the obtained three dimensional positions and postures based on changes of the angle and speed. Finally, a novel locomotion is generated using the segmented basic behavior.

Keywords: Animation synthesis, Object matching, Motion estimation, Motion control

1 Introduction

It has become possible to generate CG images with reality by the improvement of CG(Computer Graphics) techniques, especially modeling and rendering techniques. In commercials or weather reports, many CG images have been utilized frequently. Not only in movies but also in VR(Virtual Reality) researches, CG techniques have become important ones to make two or three dimensional world.

As regards the technique to represent a stable creature at present, it is possible to generate a highly realistic image. If motions are tried to be represented, however, it is difficult to generate visually natural motions. Many trials for generating natural movements of creatures by using CG(Computer Graphics) techniques have been studied. Some of these researches[1][2] represent natural fish locomotions. Sanmiya et. al.[3], [4] proposed a method which define mathematical models by analyzing the motion of creatures in observations. Tu et. al.[1] proposed a method to make an artificial creature move by defining principal motion patterns of fishes and giving parameters of mental status which are factors to cause the motion. Each of which, however, remains some problems in easily to use and generality.

In this paper, by using object matching for a fish, a method to estimate 3D position and posture of the fish from video is proposed. Moreover, the time sequence of the obtained position and postures in 3dimensional world is segmented to basic movements based on the changes of angle and speed. As a conclusion, a method to generate a novel locomotion and behavior of a virtual fish is shown by synthesizing stored segments.

In the following section, a literature of generating an animation of artificial creatures is reviewed. In the section3, a method to estimate motion parameters of a fish from video using object matching is explained. In 4th• 5th sections, motion parameters obtained in the section3 are divided into motion segments that denote a fundamental unit of motions and a method to generate automatic motion and a method to control trajectory by assigning control points are explained, respectively. Finally, a conclusion and future work are discussed in the last section.

2 Literature

In 3DCG, in order to generate animations, many researches using kinematics, inverse-kinematics, dynamics have been proposed. In these days, a method to represent a motion by using a physical model[5] or researches using artificial life[6], ¸citeAL:Sims94a are proposed to generate a motion of artificial creatures. Especially in researches regarding a motion generation of an artificial creature, the following researches try to represent a motion of fish.

A framework for animation with minimal input from the animator is proposed by Tu et. al.[1]. They define a physics-based, virtual marine world, in which artificial fishes inhabit. These motion patterns are decided by using a few mental state variables. In addition, a method to learn a fish's motion automatically is proposed[7]. Because these methods use complicated algorithm, however, it is difficult to generate motions in real time. Moreover, Manabe et. al.[8] try to represent a motion of fish by applying a fan movement of caudal fin based on vibration wing theory. It makes a virtual colored carp swim by vibrating the fin

with a constant frequency. It, however, has a problem that the algorithm has to set the angle of bend in the body against the fin by comparing with the swimming of a real carp heuristically.

On the one hand, some researches aim to realize a real time generation of fish motion[9]. Though this method enables a school of fish to locomote in real time, a motion to vibrate the body is given heuristically. Moreover, researches in regard to motion generation of an artificial creature which aim at interaction with users[10],[11] are proposed. Yamaguchi et. al.[?] propose a method to interact with a fish in a virtual water tank. It extracts a motion of a fish which swims in the water tank on a real world and makes a virtual fish swim in a virtual water tank using extracted motion parameters. It can generate a motion of virtual fish without analyzing the motion of a real fish itself by using a real image. The virtual fish itself, however, is not deformed, and only motion vector is obtained by a real image. It, therefore, remains some problems about generating realistic motions. Kurihara[11] proposes a system in which user interacts with a dolphin and generates a motion of dolphins in real time. It employs task-oriented approach and path planning to obtain motion primitives such as "swim" or "turn". Then an autonomous motion is generated using these motion primitives. Schödl et. al.[12] proposes a method to generate a non-periodic video sequence which is called "video texture" by using several parts of consecutive video images. In this paper, an image of several fish's swimming in the water tank is generated for one of applications. Since a two dimensional video image is used, however, it is difficult to represent three dimensional motion with depth.

Aiming at assistance of fishery, some researches to represent a school of fish have also been discussed. Sugiyama et. al. record tempral and spatial movement of a school of fish for a long period of time and define a mathmatical model which represents it. A method to search optimal parameters from the obtained parameters is proposed[3]. In order to model a motion of a school of real fish, however, various motions in a water tank under a uniform condition have to be collected. It is, moreover, necessary to define appropriate functions which presents various motions.

The authors, therefore, propose a method that three dimensional motion of a fish is extracted from video in which a fish is swimming by using object matching. Then, an animation of a virtual fish is reproduced using the extracted motion parameters. It is not necessary to analyze the motion of fish itself. It needs just only a simple object as a model. Moreover, motion segments are generated from the obtained consecutive motion parameters. By synthesizing the motion segments, a novel locomotion is generated.

3 Motion parameters extraction from video

This section describes a method to obtain three dimensional motion parameters from video using object matching.

3.1 System overview

There are two kinds of swimming styles of fishes which inhabit in the sea or river. One is a horse mackerel type which vibrates the rear half of the body and the other is an eel type which vibrates the whole body from the head to caudal fin. The fashion of movement in the creatures under water is wavy motion which begins from the head like progressive wave and, increases the amplitude of the vibration and makes the wavelength short in the tail of the body. It is said that fish changes its figure as a consecutive curve fits along the length of fish.

In considering this characteristic of swimming fashion of fish, the system used in this paper is configured as illustrated in Fig.1. Motion parameters of a virtual fish for generating animation are obtained by the images captured by two cameras, one of which is the upper side of a carp that is swimming in a water tank, and the other of which is the side of it. The optical axis of these cameras are set to be perpendicular each other and the synchronized images of two digital video cameras are used as input images in order to obtain vibrating motion of the fish easier. Object matching is employed by using a region of the fish which is obtained by the difference of a fish image and a background image, and by using the edge image obtained by canny edge detector. Three dimensional model which makes smooth deformation is used in object matching. A center of gravity of the fish, an inclination of the head and curvature of the backbone are obtained as motion parameters[13]. The obtained motion parameters are divided into motion segments that denote a fundamental unit of motions. Then, automatic locomotion is generated using these motion segments. Moreover, other locomotion is also generated by assigning control points. The process overview is shown in Fig.2.

3.2 A 3D virtual fish model

A three dimensional virtual fish model utilized in object matching process and animation generation is described in this section. The model is configured based on a structure of a real fish.

3.2.1 Characteristics of a fish

In this paper, motion parameters of a fish are obtained from a real motion of a fish. A carp is employed as target fish for the reason of the body size and its habit. The swimming fashion of a carp belongs to the horse

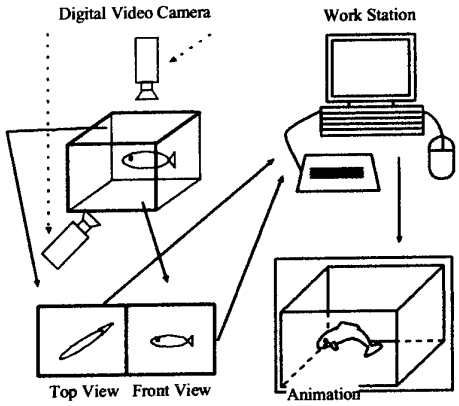


Figure 1: System configuration

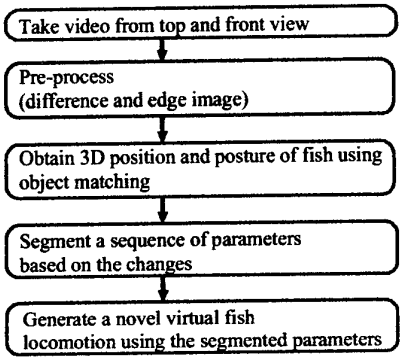


Figure 2: Process overview

mackerel type mentioned above. In this swimming fashion, a caudal fin moves as a fan and then it gains driving force by repellent force of momentum against the water. It is, therefore, necessary for fish to vibrate its body and gain the driving force in order to advance,

From a skeletal structure point of view, head is composed of bone and head length of a fish is almost one-third of its total length. Furthermore, it has a backbone composed of a great many of spines. The head, therefore, cannot deform, but the rear part of the body deforms smoothly. As a result of preliminary observation, the head hardly deforms, and only inclination and direction in the water tank coordinate system change. On the other hand, the body deforms smoothly along the backbone. The observed fish forms are shown in Fig.3.

3.2.2 A virtual fish model

As described in the previous section, the characteristics of the deformation of a fish are different from each part. Three dimensional model of a virtual fish employed for object matching is configured in which

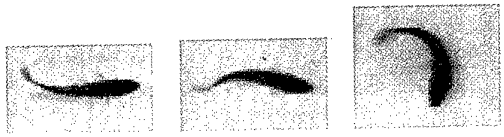


Figure 3: Forms of a fish in preliminary observation

a body structure of a fish is divided into the following three parts. In this paper, the positions of fins are not extracted in the parameters extraction process, but models of fins are created in order that the same model should be used in generating animation based on the obtained motion parameters.

Head: Because the head hardly deforms and it can be regarded as rigid body, it is synthesized by the simple polygons.

Body: Body has a backbone composed of a great many of spines, and it deforms smoothly. In order to realize a smooth deformation, NURBS(Non-Uniform Rational B-Spline) is employed. As illustrated in Fig.4, a model is configured by some groups of control points.

Fin: Since pectoral fin or caudal fin does not have muscles itself and it is just attached with the body, it is configured by connecting a polygon as a plate with the body.

The body part of three dimensional model of a virtual fish is drawn in Fig.5. A center of gravity of the fish is set to the base of the fish's head which locates at about one-third from the head, and it is set to the origin of modeling coordinate system. The number of parameters obtained in posture estimation process is 9. These are a center of gravity$\{G_x, G_y, G_z\}$, the inclination of a fish's head$\{R_y, R_z\}$ and the angles$\{A_0, A_1, A_2, A_3\}$ between skeletons which determines the posture of the body. These parameters are illustrated in Fig.5, respectively. In this paper, an assumption that there is no twist motion around the x axis in the local coordinate systems of the fish is employed.

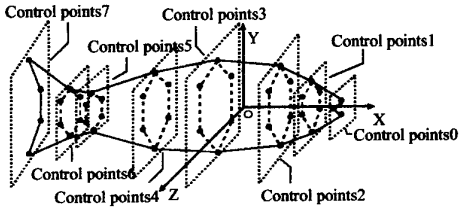


Figure 4: Control points

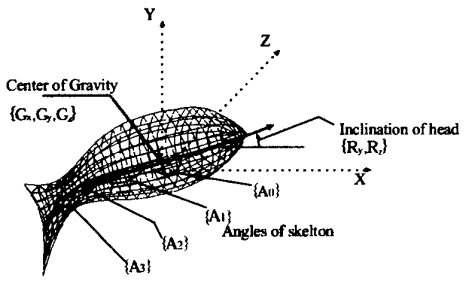


Figure 5: 3D model for object matching

3.2.3 Calibration of a 3D model

In order to create an appropriate three dimensional model which is used for object matching, a fish is captured under restricted conditions in which the posture of fish is ideal straight. Then, difference images against a background image and edge images of the upper side and the front side are generated. Next, a small size model is used as an initial model, and matching is performed varying the width and height of the 3D model for each image. The most appropriate model which has the highest evaluation value is employed as a 3D model for object matching.

3.3 Acquisition of motion parameters using object matching

In order to obtain the motion parameters of a fish, object matching is employed for each frame of consecutive input images and the posture of fish is estimated. As initial values of posture in estimation process of each frame, the posture of a previous frame is used. Moreover, a reduction of matching process is realized by hierarchical matching.

3.3.1 Acquisition of motion parameters of the first frame

In the first frame, a procedure to extract motion parameters is different from it on and after the second frame. The motion parameters are obtained from the first and the second frame. First, a difference image between the first frame and a background image in which a fish doesn't exist is generated and then the image is binarized. Next, the region in which a fish exists is extracted by performing dilation and erosion. A skeleton is extracted for the image captured from the upper side of the water tank, in order to estimate the inclination of the head. In the same way, a skeleton is extracted in the second frame. Center points of two obtained skeletons are obtained, and then a translation vector of a fish is calculated. The progress direction of the fish is estimated from the translation vector, and it assumes that the fish's head is in the same direction of the progress direction. Then, the skeleton is divided into seven equal parts. Line segments formed by the second, third, fourth, fifth of points and the end points make a skeleton of a virtual fish. Finally, motion parameters $\{A_0, A_1, A_2, A_3\}$ are obtained from the skeleton of the virtual fish.

3.3.2 Evaluation function

The motion parameters of the first frame obtained in the previous section are utilized as the initial values of the next frame, the motion parameters of each frame are obtained in turn. In extraction of the motion parameters, object matching is employed by using a difference image obtained from the upper side and the front images and using the edge image detected in canny edge detector[14].

Define that the parameters which determine the posture of three dimensional model is denoted as $\mathrm{X} = \{x_i \mid 0 < i \leq n\}$. Where, n denotes the total number of parameters of three dimensional model. As shown in Fig.6, S denotes the area of a fish in the difference image and $P(\mathrm{X})$ denotes the area by projecting three dimensional mode onto the image plane.

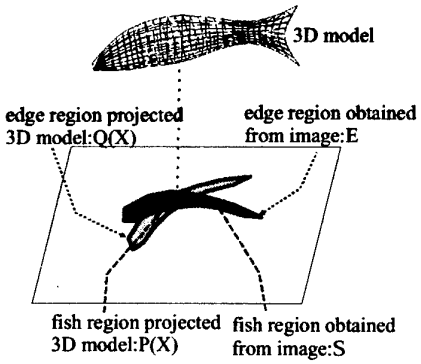


Figure 6: Evaluation of the posture of 3D model

$f(A)$ is defined as a mapping function which calculates the area of region A on the image plane. An evaluation function with regard to area is defined as follows;

$$J[\mathrm{X}] = \frac{f(P(\mathrm{X}) \cap S) - f(P(\mathrm{X}) \oplus S)}{f(P(\mathrm{X}))} \qquad (1)$$

where, $\oplus$ is XOR operator of both regions and $\cap$ is AND operator of them. In the same way, suppose that E denotes the edge region obtained from canny edge detector and $Q(\mathrm{X})$ denotes the edge region in which the model is projected onto image plane, and an evaluation function with regard to edge is defined as the following formula.

$$K[\mathrm{X}] = \frac{f(Q(\mathrm{X} \cap E) - f(Q(\mathrm{X}) \oplus E)}{f(Q(\mathrm{X}))} \qquad (2)$$

The weighted sum of the formula(1) and (2) is employed as the evaluation function of the object matching. The function V [X] is shown in equation(3).

$$V[\mathrm{X}] = J[\mathrm{X}] + W \times K[\mathrm{X}] \quad (3)$$

where W is weight.

3.3.3 Posture estimation from upper image

First, matching of the head region is performed by using the image captured from the upper side of the water tank and x, z coordinate values of a center of gravity of a fish$\{G_x, G_z\}$ and inclination around y axis $\{R_y\}$ are estimated.

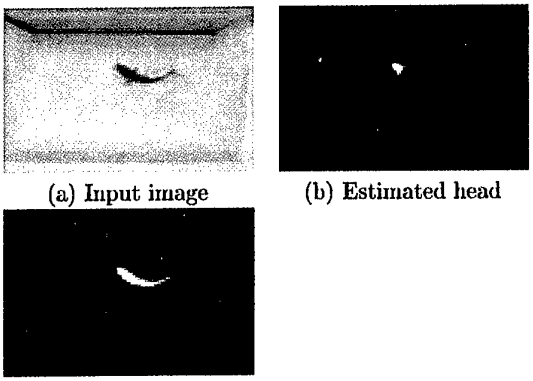
In the estimation of the head region, the region is searched within 100×100 pixels centered on the center of gravity of previous frame. Fish usually swims toward the head direction at a certain speed, because of its body structure and the swimming fashion. Therefore, suppose that the head region on the current frame exists around the prolongation of the inclination of the head on the previous frame and the head region is searched within a restricted area. Because a fish goes forward without changing its direction in most cases, the matching can perform effectively. In the case it does not satisfy this assumption such that a fish twists its body to accelerate or twists the body to change direction, that is to say, in the case the value V[X] is smaller than a certain threshold, region restriction is not employed, but matching is performed around the head region of previous frame. The matching procedure is explained as follows;

1. The position and angle are predicted roughly, as the value of $\{G_x, G_z\}$ are changed 3 pixels and that of $\{R_y\}$ is changed 3 degree, respectively. The search range is limited in the restricted region. In this prediction, the matching ratio with regard to edge which is denoted in formula(2) is used for an evaluation function.

2. If the matching ratio mentioned above is larger than a certain value, $\{G_x, G_z\}$ and $\{R_y\}$ are changed 1 pixel and 1 degree within ± 3, respectively. The most optimal parameters which make it the largest that the value of evaluation function shown in equation(3) is searched.

3. In the case matching ratio of step1 is always smaller than a certain value, it is judged that the region prediction is failed. Then, process 1 is performed without any region restriction.

The upper input image and the result of estimated head region are shown in Fig.7(a) and (b), respectively. In the posture estimation process of the body, the optimal parameters are searched by changing the value $\{A_0, A_1, A_2, A_3\}$ in turn. The result of the estimated body is drawn in Fig.7(c).



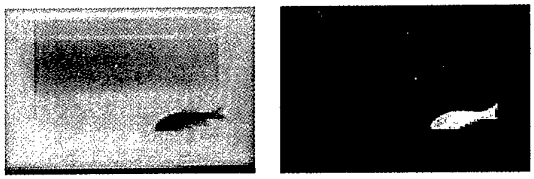
(a) Input image (b) Estimated head

(c) Estimated body

Figure 7: Posture estimation from upper image

3.3.4 Posture estimation from front image

Based on the parameters$\{G_x, G_z, R_y, A_0, A_1, A_2, A_3\}$ obtained from the upper image, the y coordinate value of the center of gravity$\{G_y\}$ and $\{R_z\}$ which is the inclination around z axis are searched. Though the fish usually vibrates the body in order to gain the driving force, the body cannot be changed its direction upper and lower direction rapidly. In the matching process, G_y and R_z are, therefore, changed by 1 pixel and 1 time, respectively, so that the optimal parameters which make evaluation value the largest are searched. Fig.8(a) and (b) illustrate the input image and the result of posture estimation from the front image.



(a) Input image (b) Estimated posutre

Figure 8: Posture estimation from front image

4 Motion generation of a virtual fish

4.1 Classification of the movements based on motion parameter

A fish gains driving force by repellent force of momentum against the water. In changing its course, it makes the direction of the head turn around by twisting the whole body. In the movement of a fish, a behavior of the body itself and a locomotion in underwater have tightly relations in each other. Therefore, it is necessary that the behavior like a twisting and

transition in the locomotion should be regarded as one movement. In this paper, a basic unit of the movement represented by the combination of such behavior and locomotion is called "segment".

4.1.1 Classification of fish posture

Postures of a fish at each frame are classified into basic posture$P$ and non-basic posture$\bar{P}$. Basic posture denotes that a fish stretches its body almost straight, and that each value of parameters$\{A_0, A_1, A_2, A_3\}$ obtained at the previous section is within ± 4 degree. The rest of postures, in which the fish leans its body, denotes non-basic posture.

4.1.2 Division into segments

The motions which have vibrating motion and deceleration motion are extracted from the motion sequence. The vibrating motion denotes a motion from P to $\bar{P}$ and the returning to P again. The deceleration motion denotes the movement which keeps up the posture of P successively. The concept of the classification of the movement is shown in Fig.9. The movement as $F_1 \rightarrow F_2 \rightarrow F_3$ in Fig.9 is the one in which a fish makes the body vibrate only for a certain amplitude and returns to the basic posture. These changes of postures are regarded that the fish accelerates for moving forward or leans the body for turning. On the other hand, in the movement $F_0 \rightarrow F_1$ in Fig.9, a fish moves toward a certain direction without vibrating the body. In this movement, it is regarded that the fish advances just by inertial force and that the speed decelerates.

However, in the case that the direction of the fish hardly changes as $F_1 \rightarrow F_3$ and $F_3 \rightarrow F_5$ illustrated in Fig.9, that is, such a movement of vibrating the body to accelerate, it is also divided into segments. Therefore, the merge of vibration behavior is performed by the following procedure described in the next paragraph.

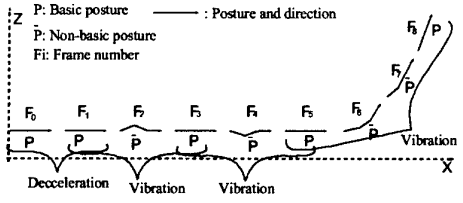


Figure 9: Segmentation into basic behaviors

4.1.3 Merge of segments

In the case that the amount changes of the head's inclination is smaller than a threshold in consecutive vibrating motions, plural vibrating motions which were described in the former are merged into one segment. If the angle$\{R_y\}$ around y axis at the first basic posture and the angle at the last basic posture should be smaller than a certain threshold T_h in the two consecutive vibrating motions, they are merged. On the other hand, if the angle is larger than the threshold, segments are kept divided as different movements. Three motions, which are deceleration motion as $F_0 \rightarrow F_1$, acceleration motion as $F_1 \rightarrow F_5$, and right turn or left turn motion as $F_5 \rightarrow F_8$ are obtained as shown in figure10.

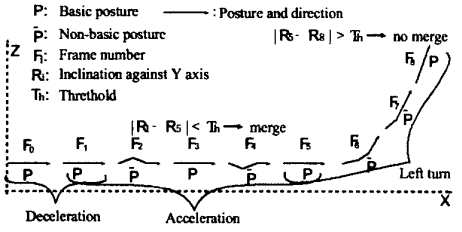
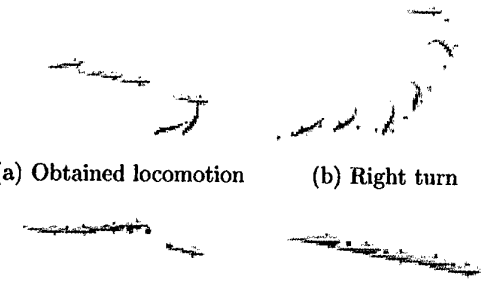


Figure 10: Segmentation into basic locomotion

4.1.4 Results of motion segments

A movement of virtual fish generated by obtained motion parameters is illustrated in figure11(a) and the results of the divided segments are shown in figure11(b)~(d). The results of the segments are shown on the local coordinate system of the first basic posture. Fig.11(a) shows every 3 frames of the trajectory of the center of gravity of the fish in 150 frame motion parameters obtained in the previous section as black points. It also shows the posture of a fish at the beginning and ending of the segment. Fig.11(b)~(d) are segment which is obtained by the movement of Fig.11(a), and each shows right turn, acceleration and deceleration motion. These figures show trajectory of the center of gravity of the fish in every 3 frames, and illustrate the posture of a fish in every 5 frames.



(a) Obtained locomotion (b) Right turn

(c) Acceleration (d) Deceleration

Figure 11: Segmented locomotion

4.2 Motion Synthesis of a Virtual Fish

A method to generate a novel motion of a virtual fish by connecting several segments is discussed in this section.

4.2.1 Rule for connecting segments

As shown in Fig.12, a motion synthesis of virtual fish is realized by connecting the end points of each segment. In connecting segments, coordinate system is transformed in order that the new segment should be aligned with the final basic posture of the previous segment. As described in the previous section, the motion of a fish is connected smoothly because all postures of the fish in the ends of segments are basic postures. Also in a novel movement generated by connecting segments, unnatural motions by changing the body postures don't appear.

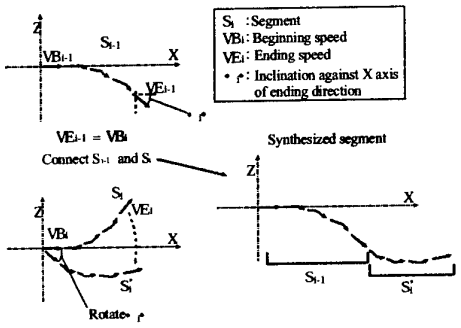


Figure 12: Connection rule of segments

On the other hand, unnatural movements are generated in the case that the arbitrary two segments are connected because the speed of virtual fish is different from each segment. Therefore, the following restricted conditions are given in connecting two segments S_{i-1} and S_i. It is connected only in the case that the ending speed Ve_{i-1} of segment S_{i-1} and the beginning speed Vb_i of segment S_i are almost the same. As a result of preliminary experiments, the beginning speed Vb_i and the ending speed Ve_i of each segment were within the range of $0 \sim 6$ pixel/frame. Therefore, in this paper, the speed is divided into 3 ranges and the connection should be permited if the speed is within the same range.

4.2.2 Results of motion synthesis

The generated animation using the connecting rule described in 4.2.1 are shown in figure13. The points in the figure denote trajectory of the fish's center of gravity, and virtual fishes in the end of segment are illustrated. In generating animation, segment which satisfies the possibility of connection is selected at random and then connected. In the case it doesn't satisfy the possibility, segment data is selected from a set of data at random until the one which satisfies is selected.

(a)Deceleration • •left turn (b) complex locomotion

Figure 13: Examples of jointed segments

5 Motion synthesis of virtual fish by assigning control points

In this section, a trajectory control method in which virtual fish passes through control points, is described.

5.1 Trajectory control using control points

To control the trajectory of a virtual fish's movement, the connection conditions are more restricted than the one in previous section. As shown in figure14, to generate a motion aiming to go towards control point C_i, a segment which makes the position to be closest to C_i is selected. Moreover, by virtual fish's center of gravity passing through the region of a certain distance from the control point C_i, the target control point is changed to the next control point C_{i+1}.

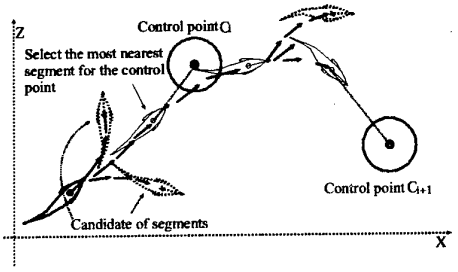


Figure 14: Selection of segment based on control points

5.2 Experimental results

The result of virtual fish movement by assigning several control points is illustrated in figure15. Figure15 shows the movement every 60 frames. Each control point is drawn by a small cube, and it shows a virtual fish is moving in specific order.

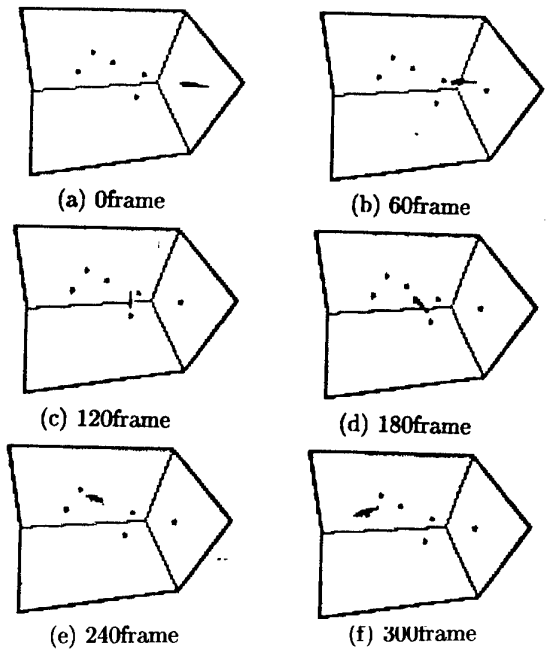


Figure 15: Control of virtual fish locomotion using control points

6 Conclusions

In this paper, a method to obtain motion parameters of a fish is proposed by employing posture estimation of the fish. In posture estimation, object matching technique is performed to each frames of the video taken from a real fish in a water tank. Motion parameters denote a center of gravity, the inclination of a fish's head and the angles between skeletons which determines the posture of the body. These represent a transition of the movement of fish with the change of times. The method to divide the motion parameters into several segments by using the characteristics of the motion of fish is proposed. The method presents the virtual fish's movement which is quite natural by synthesizing according to the segment connection rule.

We are planning to propose a more accurate and more high performance extraction method of motion parameters. It is also a charming theme to make a motion data base and its retrieval. Furthermore, we are trying to generate retargetting motion from the obtained motion parameters and compare with physical-based model technique quantitatively.

References

[1] X. Tu and D. Terzopoulos: "Artificial fishes: Physics, locomotion, perception, behavior", Computer Graphics Proceedings, Anual Conference Seriese, pp. 43 50 (1994).

[2] J. Hatoya, M. Nakajima and H. Takahashi: "Generation of virtual fish movement from video", Proceedings of the VSMM'98, pp. 122 127 (1998).

[3] N. Sannomiya and K. Matsuda: "Modeling of the behavior of fish schools", Journal of the society of Instrument and Control Engineers, **19**, 7, pp. 704–707 (1980).

[4] N. Sannomiya, H. Nakamine and H. Iwasaki: "A study on the validity of a physical model for fish behavior", Transactions of the Institute of System Control and Information Engneers, **3**, 1, pp. 14 20 (1990).

[5] G. Miller: "The motion dynamics of snakes ad worms", Computer Graphics, **22**, 4, pp. 169 177 (1988).

[6] C. W. Reynolds: "Flocks, herds, and schools: A distributed behavioral model", Proceedings of SIGGRAPH'87, pp. 25–34 (1987).

[7] R. Grzeszczuk and D. Terzopoulos: "Automated learing of muscle-actuated locomotion through control abstraction", Computer Graphics Proceedings, Anual Conference Seriese, pp. 63–70 (1995).

[8] T. Manabe, T. Harima, Y. Anzai, N. Chiba and N. Saito: "Motion simulation of virtual varicolored carp based on the vibration wing theory", The Journal of the Institute of Image Information and Television Engineers, **52**, 9, pp. 1374 1378 (1998). (in Japanese).

[9] T. Kawamura, H. Dohi and M. Ishizuka: "Real-time cg animation of realistic fishes using nurbs, inverse kinematics and a co-operative motion model", The Journal of the Institute of Image Information and Television Engineers, **49**, 10, pp. 1269–1304 (1995).

[10] T. Yamaguchi, T. Masaki, Y. Kitamura and F. Kishino: "Interactive digital fishtank based on live video imges", Proceedings of The Third VRSJ Annual Conference, **3**, pp. 221–222 (1998).

[11] T. Kurihara: "Real-time animation method of artificial creatures 'MDolphins'", The Journal of the Institute of Image Electronics Engineers of Japan, **27**, 4, pp. 338–347 (1998). (in Japanese).

[12] A. Schödl, R. Szeliski, D. H. Salesin and I. Essa: "Video textures", Computer Graphics Proceedings, Anual Conference Seriese, pp. 489–498 (2000).

[13] J. HATOYA, H. TAKAHASHI and M. NAKAJIMA: "Synthesis of virtual fish locomotion and behavior from video", ITE Technical Report, **24**, 29, pp. 33–38 (2000). (in japanese).

[14] J. Canny: "A computational approach to edge detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, **8**, 6, pp. 679–698 (1986).

ICAT 2000

Multimedia Virtual Laboratory on the Gigabit Network

Tetsuro Ogi[1), 2), 3)], Toshio Yamada[1)], Makoto Kano[2)]
Koji Yamamoto[3)], Koichi Hirota[2)], Michitaka Hirose[2)]
[1)] Gifu MVL Research Center, TAO
[2)] The University of Tokyo
[3)] Mitsubishi Research Institute
tetsu@iml.u-tokyo.ac.jp

Abstract

Multimedia virtual Laboratory is a distributed virtual environment in which remote researchers can communicate mutually sharing research resources though the broadband network. In order to realize this concept, immersive projection displays CABIN at the University of Tokyo and COSMOS at the Gifu Technoplaza were connected through the Japan Gigabit Network. In particular, stereo video avatar and immersive database interface technologies were developed. These technologies were implemented in the CABIN to COSMOS network, and the high presence communication sharing data in the virtual world was realized.

Key words: Immersive Projection Display, Shared Virtual World, Broadband Network, Video Avatar, Database Interface

1. Introduction

Recently, according to the advances in the broadband wide area networks, real-time transmission of a large amount of data such as three-dimensional models or video images has become possible between remote places. For example, the Japan Gigabit Network (JGN) was equipped by the Telecommunications Advancement Organization of Japan in 1998. JGN is a nationwide optical-fiber network and it has been used for research and development activities. This kind of network enables the remote researchers to collaborate sharing the research resources such as the computers and the data. In particular, a three-dimensional virtual world can be shared between remote places by connecting virtual reality environments through the broadband network [1].

This study aims at constructing a high presence shared virtual world in which remote researchers can communicate mutually as if they are in the same place, by connecting immersive environments through the JGN network. This type of research environment is called multimedia virtual laboratory (MVL). In order to realize this concept, it is necessary to develop a shared virtual world in which remote researchers can communicate with a high presence sensation accessing and sharing data.

This paper describes the prototype system of the multimedia virtual laboratory environment developed in this study, and in particular, the video avatar communication and the immersive database interface which are key technologies of the multimedia virtual laboratory are discussed.

2. CABIN to COSMOS Network

Multimedia virtual laboratory is a concept of distributed virtual environment in which remote researchers, research equipment and information are connected via the broadband network as if they are in the same place. Fig. 1 shows the concept of the multimedia virtual laboratory. In this example, the computer scientist, the experimental engineer and the designer are jointly working in the shared virtual world to develop an airplane. Although these researchers are not usually in the same place, the multimedia virtual laboratory enables them to meet and hold discussions through the network.

In order to realize the multimedia virtual laboratory, MVL Research Center was founded in the University of Tokyo and the Gifu Technoplaza in 1999. Fig. 2 shows the research environment of the MVL Research Center. At the University of Tokyo and the Gifu Technoplaza, large-screen immersive projection displays CABIN and COSMOS were developed and are used respectively [2][3]. CABIN is a multi-screen cubic display that has five screens at the front,

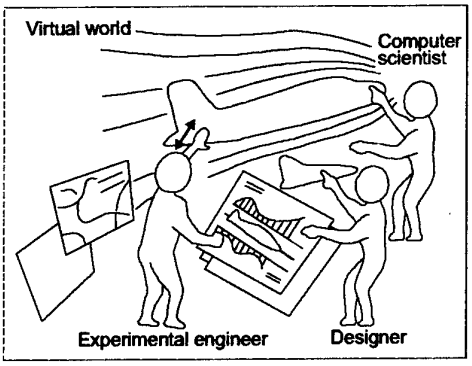


Fig. 1 Concept of multimedia virtual laboratory

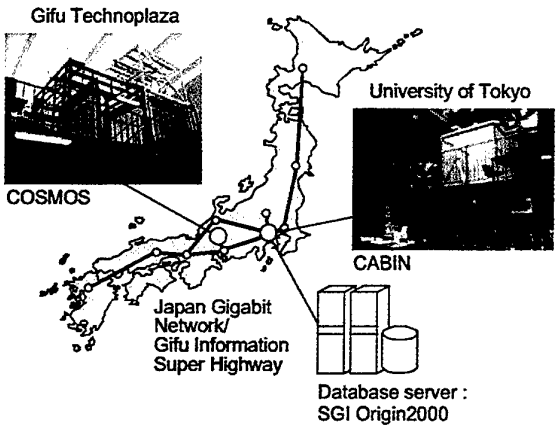


Fig. 2 Research environment of MVL Research Center

Fig. 3 Handykey Pointer used in the immersive projection display

on the left, right, ceiling and floor, and COSMOS is a complete immersive display that has six screens by adding the back screen. These displays can generate a highly immersive virtual world by surrounding the users with stereo images projected on the multiple screens.

In the immersive projection display such as a CAVE, a joystick type input device called wand is generally used [4]. This type of device is useful for walkthrough or handling object in the three-dimensional virtual world. However, it cannot be used to input characters that is an indispensable function to access database system. Therefore, in this study, Handykey Pointer was developed by combining the position sensor Polhemus Ultratrak Pro with the handy keyboard Twiddler made by Handykey Corporation [5]. This device can be used to input characters and point objects by one hand in the CABIN and COSMOS as shown in Fig. 3.

In this study, CABIN and COSMOS were connected by 155Mbps ATM using the JGN network and the Gifu Information Super Highway to construct a prototype system of the multimedia virtual laboratory. Therefore, in the networked environment between CABIN and COSMOS,

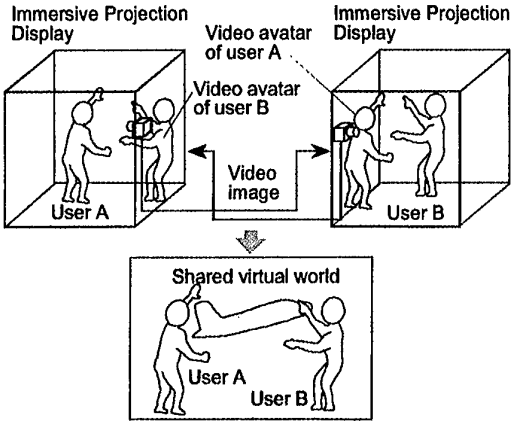


Fig. 4 Concept of video avatar communication

remote users can share the virtual world with a high quality of immersion. In addition, in order to use this environment for the multimedia virtual laboratory, it is necessary that the remote researchers can hold a discussion with high presence sensation while sharing the data such as design models or simulation data in the virtual world. In this environment, the database server SGI Origin2000 was also connected to the network so that the users can easily access data from the virtual world. Thus, the framework of the multimedia virtual laboratory in which remote researchers can access database interactively from the shared virtual world was constructed. In the following chapters, several technologies implemented in the prototype system of the multimedia virtual laboratory were described.

3. Video Avatar Communication

3.1 Concept of Video Avatar

In order to realize a high presence communication in the shared virtual world, it is necessary that the users can see the other user's figure mutually. For the communication method in the distributed virtual world, the computer graphics avatar is often used to represent the participant's figure [6]. This method can represent the user's action in the three-dimensional virtual world. However, it is difficult to represent the facial expression of the user, because it is created using the computer graphics polygon model.

Therefore, in this study, stereo video avatar technology was developed [7]. This method represents the three-dimensional avatar using a live video in the shared virtual world. By transmitting the stereo video avatar mutually between remote places, the users can communicate with a high presence sensation. Fig. 4 shows the concept of video avatar communication in the networked immersive projection displays. In this method, the user's image is captured by a video camera placed within the immersive projection display, and a video avatar is created. This video avatar is sent to the other site and superimposed on the

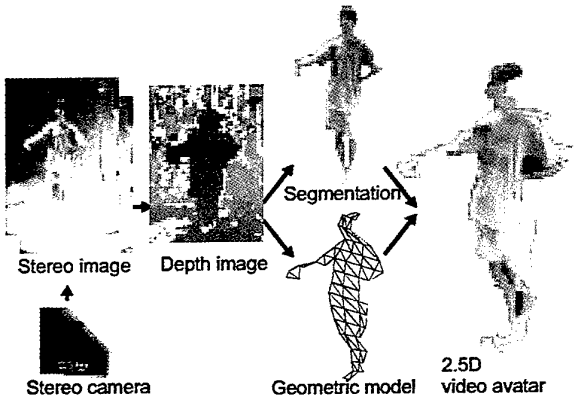


Fig. 5 Basic process of making 2.5 dimensional video avatar

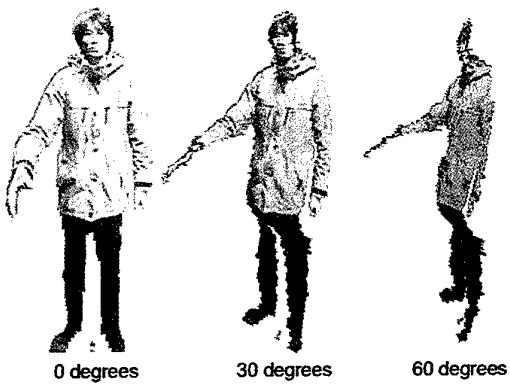


Fig. 6 Appearance of 2.5 dimensional video avatar seen from various directions

virtual world. By transmitting the video avatars mutually, remote users can communicate face to face in the shared virtual world.

3.2 Creation of a Stereo Video Avatar

Fig. 5 shows the basic process of making the stereo video avatar. In order to generate a three-dimensional video avatar, it is necessary to create a geometric model of the user while capturing the user's video image. Therefore, in this study, a stereo camera of the Triclops Color Vision made by Point Grey Research Inc. was used to capture the user's image [8]. By using the stereo camera, depth data can be calculated for each pixel in the captured image using the stereo matching algorithm. Since this stereo camera consists of two pairs of stereo camera modules along the vertical and horizontal base lines, it can create a accurate depth image. The resolutions of the captured color image and the created depth image are 320x240 pixels and 160x120 pixels respectively, and the calculated depth resolution was about 5.0 cm.

Once the depth image is created, only the user's image can be segmented from the background by the threshold of the depth value. In practical applications, the chroma key can also be used in combination with the depth key to create a clear image of the avatar. Additionally, a geometric model of the user is also created by connecting the three-dimensional pixel positions using a triangular mesh. Then, by texture-mapping the segmented user's image onto the geometric model, a stereo video avatar is generated. Since this avatar only has the surface model for the front side that faces toward the stereo camera, it is called a "2.5 dimensional video avatar".

3.3 Multi-camera System

Fig. 6 shows the appearance of a 2.5 dimensional video avatar seen from various directions. When the user sees the 2.5 dimensional video avatar from a viewpoint close to the camera position, avatar's image is well formed. However, when the user's viewpoint moves away from the camera position, the avatar's image becomes distorted, because it only has the surface model that faces to the camera position. Therefore, in this system, multiple stereo cameras were placed inside the immersive projection display and the closest camera to the other user's viewpoint was selected and used. By switching the selected camera according to the positional relationship between users, in effect, a quasi three-dimensional video avatar can be generated. Fig. 7 shows an example of the stereo video avatar superimposed on the shared virtual world. In this example, the user in the CABIN is talking to the video avatar in the shared virtual world.

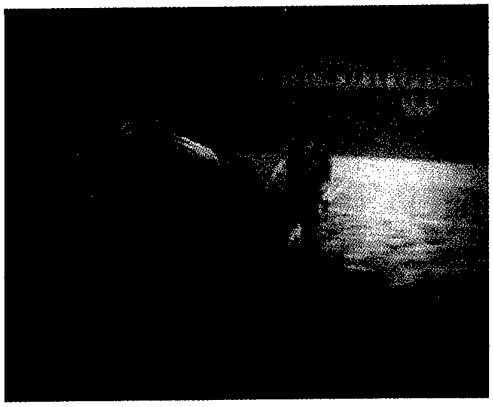


Fig. 7 Video avatar superimposed on the shared virtual world

Though this method uses multiple stereo cameras, only one pair of stereo cameras is used at any given time. So, the stereo video avatar can be generated in real-time. When a Pentium III 700MHz PC was used, the stereo video avatar was generated at a refresh rate of about 9.9 Hz, and the time delay was about 0.6 sec.

Fig. 8 Data retrieval by keyword search in the virtual world

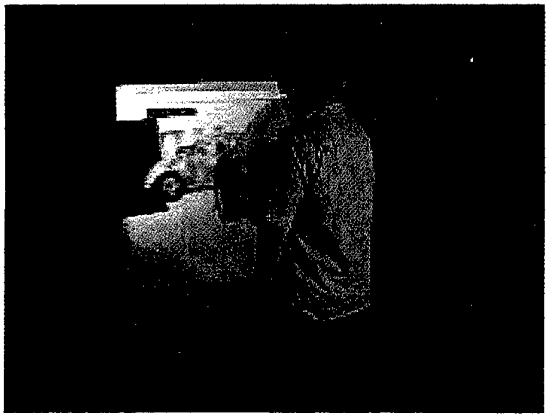


Fig. 9 Spatial data bowsing in the three-dimensional world

4. Immersive Database Interface

4.1 Concept of Immersive Database Interface

In order to realize the multimedia virtual laboratory, it is necessary that the remote researchers can not only talk to each other but also share data such as design models or simulation data in the shared virtual world. Next, in this study, a framework of accessing database server from the three-dimensional virtual world was constructed.

When we manage documents in our office in the real world, we usually use a file and a bookshelf to arrange the documents. For example, some documents are filed and put on the right bookshelf, and other papers are put on the left shelf. In this case, these documents are managed using positional information in the three-dimensional world. On the other hand, when the computer is used to manage information, large memory and computation capability of the computer can be effectively utilized. For example, in the typical data accessing method of the keyword search, related data can be immediately retrieved from the database by simply entering the keywords.

In the virtual world, it is expected that the data accessing method that utilizes both advantages of the real world and the computer can be used, because the virtual world is a realistic world simulated by computer. In this study, an immersive database interface system was developed to access data from the shared virtual world.

4.2 Functions of Database Interface

The database interface developed in this study has the following functions to handle data such as photograph images or three-dimensional models in the immersive virtual world.

4.2.1 Keyword Search

In order to access database from the three-dimensional virtual world, a function of a keyword search using the Handykey Pointer was implemented. In this method, the user opens a data search window in the virtual world and inputs a keyword using the Handykey Pointer. From the inputted keywords, a SQL (Structure Query Language) query command is generated and it is sent to the database server. By using the SQL query command, the user can access an arbitrary database system on the network. When the data is retrieved from the database server and taken into the virtual world, the abstract data is visualized as a concrete object being filed in a book. Fig. 8 shows the example of retrieving data using the keyword search in the virtual world.

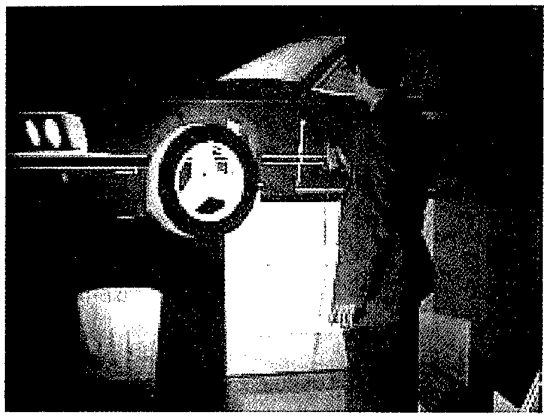


Fig. 10 Extracting data from the book into the virtual world

4.2.2 Data Browsing

After retrieving data from the database system, the user can search a target data from the data filed in the book, by using various data browsing methods. The most simple data browsing method is turning over pages of the visualized book by key operation of the Handykey Pointer. The data filed in the book can also be scrolled spatially by flying

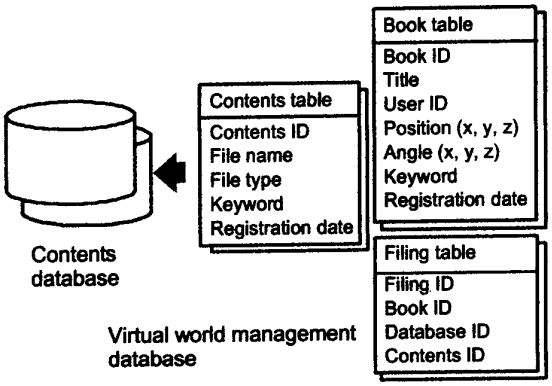


Fig. 11 Database structure for data management in the virtual world

data out of the book in the three-dimensional virtual world as shown in Fig. 9. This method is thought to be utilizing the characteristics of wide viewing field of the immersive projection display effectively.

When the target data is found, the user can take it out of the book and place it in the virtual world. Fig. 10 shows the example of extracting the selected data. In this way, design model or simulation data can be taken into the shared virtual world from the database server to discuss in the multimedia virtual laboratory.

4.2.3 Data Management

Once the data is taken into the virtual world, positional information is linked to the data, and it can be treated as an object in the three-dimensional virtual world. For example, the user can grasp the visualized data and move it to the other place. By using this function, data can be replaced between books and filed according to the themes. These books can also be arranged put on the bookshelf in the virtual world. In this method, the three-dimensional positional information is effectively utilized in the same way as the office in the real world. Therefore, the data management method using the book and bookshelf in the virtual world is thought to be applying a "office metaphor".

4.3 Database Management

In this system, though the data taken into the virtual world was originally stored in the database server, it is also managed by using the database system in the virtual world. As for the database management system, INFORMIX-Universal Server is used to treat data without contradiction in the virtual world. Fig. 11 shows the data tables which were defined to manage data using books and bookshelves. Namely, the data taken into the virtual world is managed using the book table, contents table and filing table.

The book table records the position where each book is placed in the virtual world, and the contents table records the file format and the location of each contents data. And in the filing table, the relationship between each contents and the book that files contents data is recorded. By relating these data tables, the data taken into the virtual world can be managed efficiently.

5. Prototype of Multimedia Virtual Laboratory

In this study, by integrating the stereo video avatar and database interface technologies, a prototype system of the multimedia virtual laboratory was constructed, and the communication experiment was conducted using the CABIN to COMOS network. Fig. 12 shows the system configuration of the experimental setup of the multimedia virtual laboratory. In this experiment, various types of data such as design models, photograph images and simulation results were stored in the database server. And the users in the CABIN and COSMOS accessed these data and took them into the shared virtual world to hold a discussion by transmitting their video avatars mutually.

In this experiment, two stereo cameras were used on each

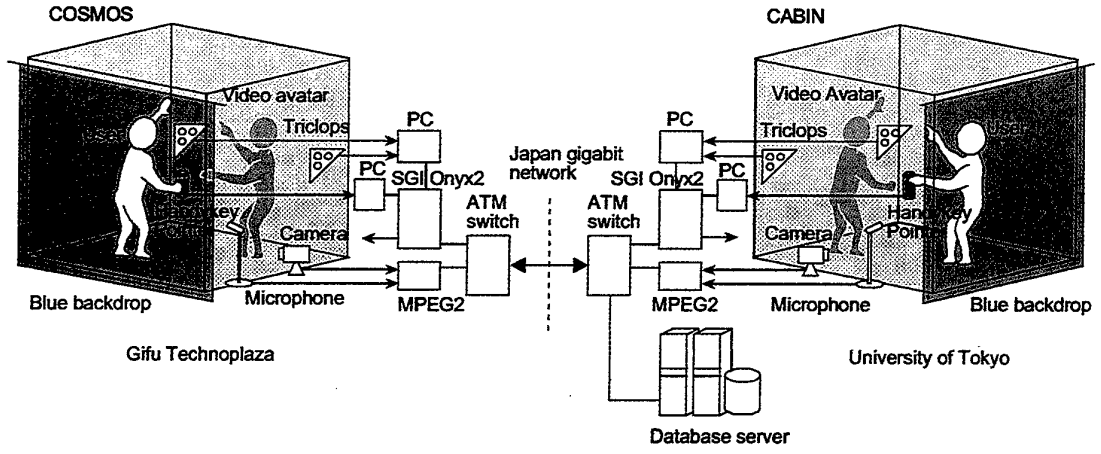


Fig. 12 System configuration of the prototype system of the multimedia virtual laboratory

Fig. 13 Example of the communication in the multimedia virtual laboratory

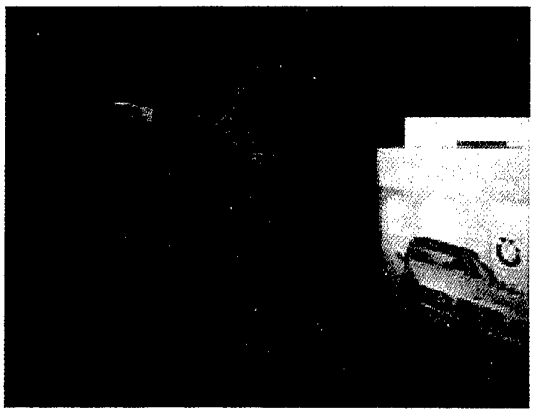


Fig. 14 Example of accessing data in the multimedia virtual laboratory

site to switch the geometric models of the video avatar, and the data of the video avatars were transmitted every refresh time of making video avatar. In order to transmit these data, about 40Mbps bandwidth of the network was used. Additionally, an MPEG encoder and decoder were used to transmit the scene to and from the opposite site, and they are also used to send the voice of the video avatar. On the other hand, in order to share the data in the shared virtual world, only the operation command was transmitted mutually, and the database server was accessed from each site respectively. In this way, the prototype system of the multimedia virtual laboratory enabled the remote researchers to communicate with high presence sensation while accessing data freely in the shared virtual world. Fig. 13 and Fig. 14 show the examples of the communication sharing data in the prototype system of the multimedia virtual laboratory.

6. Conclusions

In this study, the concept of the multimedia virtual laboratory was proposed, and the research environment was constructed by connecting the immersive projection displays CABIN and COSMOS through the broadband network. In particular, in order to realize a high presence communication sharing data in the multimedia virtual laboratory, stereo video avatar and database accessing method were developed. These technologies were implemented in the prototype system of the multimedia virtual laboratory, and the communication experiment was conducted.

The concept of the multimedia virtual laboratory also includes the functions of the multimodal communication and the interactive usage of the supercomputer. Future work will include improving the functions of the prototype system and applying this framework to the practical applications such as collaborative design or discussion between remote places, and the effectiveness of this system will be evaluated.

References

1. Leigh J., DeFanti T.A., Johnson A.E., Brown M.D., Sandin D.J., Global Tele-Immersion: Better Than Being There, ICAT'97, pp.10-17, 1997

2. Hirose M., Ogi T., Ishiwata S., Yamada T., Development and Evaluation of the Imersive Multiscreen Display CABIN, Systems and Computers in Japan, Vol.30, No.1, pp.13-22, 1999

3. Yamada T., Hirose M., Iida Y., Development of Complete Immersive Display: COSMOS, Proceedings of VSMM98, pp.522-527, 1998

4. Cruz-Neira C., Sandin D.J., DeFanti T.A., Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE, Proceedings of SIGGRAPH'93, pp.135-142, 1993

5. Handykey Corporation web page: http://www.handykey.com/

6. Leigh J., Johnson A.E., Vasilakis C.A., DeFanti T.A., Multi-Perspective Collaborative Design in Persistent Networked Virtual Environments, Proceedings of the IEEE VRAIS, pp.253-260, 1996

7. Hirose M., Ogi T., Yamada T., Tamagawa K., Development of Stereo Video Avatar in Networked Immersive Projection Environment, ICIP'99, 1999

8. Point Grey Research Inc. web page: http://www.ptgrey.com/

ICAT 2000

Supporting Team Work in Collaborative Virtual Environments

Gernot Goebbels, Plinio Thomaz Aquino, Martin Göbel
German National Research Center for Information Technology (GMD)
Virtual Environments Group, Germany

Vali Lalioti
Computer Science Department
University of Pretoria, South Africa

Abstract

In this paper we present our approach in creating Collaborative Virtual Environments to provide distributed collaborative teams with a virtual space where they can meet as if face-to-face, coexist and collaborate while sharing and manipulating a set of virtual data in real time. Thereby our approach moves beyond mere integration of video-conferencing and scientific visualization, to create a design framework for CVEs where issues of human-to-computer and human-to-human interaction in projection-based systems are addressed. It focuses on our interaction taxonomy that supports the development of applications which themselves support small groups working together in rear projection-based VEs making use of video conferencing and 6DOF input devices. The approach is exemplified by the design and implementation of a Collaborative Medical Workbench application used for remote education purposes.

Keywords Distributed VEs, Immersive Telepresence, Collaborative Interaction Framework

1 Introduction

The need for high-end collaborative Virtual Environments becomes more pressing due to the globalized nature of today's market. Distributed businesses require support for effective collaboration over distance in order to minimize time and travel costs[2]. Businesses that require high-end visualization of raw data gathered from remote sites [3] [4], as well as remote medical consultation [6] and tele-education, are examples where scientific visualization has been combined with video-conferencing to provide support for collaborative work [8].

In our approach for the design of such an environment, principles developed in the field of human computer interaction and computer supported collaborative work (CSCW) are complemented by techniques for facilitating human to human interaction within a virtual space, for users physically at the same place or for remote collaboration. Design issues involve ways of natural interaction with the virtual data as well as with the remote participants, while preserving shared data consistency.

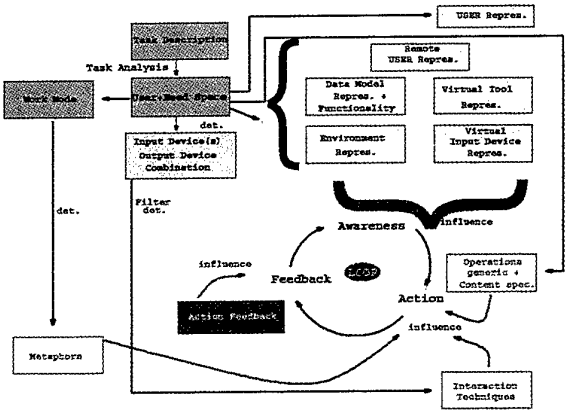


Figure 1: Taxonomy for Autonomous and Distributed, Collaborative Interaction.

2 Taxonomy for Distributed, Collaborative Interaction

This approach is a practical tool that can be used as a framework for design and evaluation of VEs. Therefore, we are concerned only with the utility of a taxonomy for these tasks, and not its absolute "correctness". The objective is to facilitate guided design of applications for supporting team work in VEs. One way to verify the generality of the approach is through the process of categorization. Categorization is a good way to understand the low-level makeup of interaction techniques. This categorization may also lead to new design ideas. User tasks need to be specified which

Representation Components	User Representation	# users : x	none	
	Remote User Repr.	# remote users :y	video texture	
	Data Model Repr.	model 1+ model 2 + model 3 +...		
	Data Model Functionality	f(model 1) + f(model 2) + f(model 3) + f(...)	...	...
	Environment Repr.	environment	inventory	
	Virtual Tool Repr.	3D tool cursor 1	3D tool cursor 2	...
	Virtual Input Device Repr.	colored rays	3D cursors	...
	Work Mode	mode 1		
Devices	Input Devices	1.) Stylus / 3 Button Tool	2.) Cubic Mouse / Pinch Glove	
	Output Devices	RWB / RWB	RWB / Wall	RWB / CAVE
Auxiliary	Metaphors	Tug Of War	Work around a table	...
	Operations	generic : grab, zoom, drag, rotate,	content specific : change mode of vis.	
	Interaction Techniques	body-centered	menu	speech recognition
Loops	Action Feedback	highlighting of actions	sonification of actions	...
	...	action 1+ action 2 + ...		

Figure 2: The User+Need Space.

will then determine the application and interaction requirements, before the correct VE and interaction techniques can be chosen. This implies a good analysis of the special needs of users the application is going to be designed for.

2.1 User's Task Description and Analysis

Figure 1 shows that the approach starts with a *User's Task description (UTD).* A task description can look like in the following:
Assume two users who want to connect two virtual wooden laths with each other. They use a hammer and a box of nails. For pulling nails that are wrong pound into the wood they use a pair of pliers. Both stand at either side of a carpenter's workbench. One user holds the wooden laths and the other user pounds the nails with the hammer or uses the pliers respectively.

This description provides information about the number of users involved in the task, the type of material and the tools they use. It describes where the users stand and how they work together. Now a following *User's Task Analysis (UTA)* determines the so-called *User+Need Space (UNS)* which itself is the originator of the flow within the taxonomy graph. This UNS relays the information extracted by the UTA of the UTD. We recommend to do an extensive description and analysis of the user's task in order to find out how the user's need can be satisfied. From our point of view most of the virtual environments lack the addressing of user needs and thus result in a poor user satisfaction and usability.

2.2 The User+Need Space (UNS)

In order to represent the UNS visually we choose an array-like representation. (see Figure 2) However, any other representation form is possible but we think that the mapping between the requirements of the UNS and the features of the Virtual Environment is much more obvious using this type of representation. The first seven features denote representation components (see 2.4). In addition to the number of local and remote users the corresponding representations are included. Although the UNS in Figure 2 is a UNS template, we added different possibilities of realisations. Consequently when working two-handed, different input device combinations are shown, such as a combination of a stylus and a 3 button tool or the combination of a pinch glove and a cubic mouse respectively(see 2.3). These and other combinations are not obligatory, they are just illustrating the usage of the UNS array. Also the items belonging to the operations, metaphors and interaction techniques in the auxiliary section of the array are just of illustrating nature and shows that more than one item can be taken under consideration. Thereby, if in the rows appears an enumeration, the first item or combination has be interpreted as the most appropriate. Then the application designer has to choose one of the suggestions. If there is no enumeration the row represents a list of items that belong together. Then all have to be taken under consideration within the application design.

2.3 Input/Output Device Combination and Working Mode

It is obvious that not all 6DOF input devices for interaction and output devices for interacting can be combined. For example, it is hard to use a Cubic Mouse together with a stylus in a CAVE-like display system if the stylus needs to be used frequently. The reason is simply that for using the Cubic Mouse the user needs both hands which results in putting other input devices away. Combining these input devices with the RWB as output device for example the user has got the possibility of putting unused devices back on the table of the RWB. But of course this cannot be the only reason for choosing a certain type of input/output combination. The selection of the devices is mainly influenced by other factors. Most important fact for the selection of an adequate output device is the amount of users who work together at the same site and of course the size of the data model. The most adequate display system for an architect who shows the pre-visualized interior of a building to the client is a wall or a cylindrical projection and a Cave rather then a RWB or a ReachIn display system. An adequate combination of input devices and output devices has to be found with respect to the user's task and data set of use. Thus input and output combination of interest is directly derivable from the User+Need space as all needs and requirements are already defined there.
The Work Mode is determined by the user's task too. Different modes of work are:

- stand-alone, autonomously and data sets are locally uploaded
- stand-alone, autonomously and data sets are remotely uploaded
- stand-alone, collaboratively and data sets are locally uploaded
- stand-alone, collaboratively and data sets are remotely uploaded
- distributed, collaboratively and data sets are provided by one of the sites, or by a remote (external) data server

The first two items describe the possibility to work alone where data sets are locally available or must be downloaded remotely from a simulation loop for instance. No collaborative working is enabled at all.
The third and the fourth item described collaborative working together using one display system. The data sets are available locally again or have to be downloaded from a remote data server.
The last item is the more interesting one where at least two sites work together. Now the shared data sets can either be provided by one or even more members of the session or be provided by an external data server. The work mode itself it important to determine the metaphors described in 2.6.

2.4 Representation Components

Representation Components denote a very important part of Virtual Environments. They determine how the visual parts in the application are represented. The components are (see Figure 1):

- User Representation
- Remote User Representation
- Data Model Representation and Functionality
- Environment Representation
- Virtual Input Device Representation
- Virtual Tool Representation

As shown in Figure 1 all components except for the *User representation* belong to a group. The User Representation is of interest only to the user and not to the remote partner. Most rear projection-based Virtual Environments do not need an explicit user representation in contrast to HMDs, where the user is typically represented by a hand or a whole body like in *Third Person Shooting* games.

The *remote user representation* represents the participating user or group of users at the other site. The aim of this representation form is to let this user or the group to appear present in the remote virtual environment. Therefore the factor of realism depends on the task of the users. Sometimes even more abstract user representations fit the requirements. Well-established methods of user representation are avatars and real time video textures. Research on avatars has produced from very abstract to very detailed human representation that include realistic visual and physical models [1]. Research on using real-time video is using stereoscopic or mono video and different texture mapping and image manipulation techniques [8]. The advantages of video conferencing are the high realism and the ease in handling of the video texture in order to position and scale it. The disadvantages are the transfer of video streams of the net and the match-moving of the texture with the virtual tool and input device representations selected by this user.

The *data model representation* is the data set of interest. Depending of the application these data sets can be a human body reconstructed from MR and CT recordings and a saw and drill for the surgeons, the car model with seats and crash test dummies for the engineers or the set of molecules for the chemistry professor. Data sets of interest can either be abstract models or reconstructed from scanner data for example. The best representation form is determined by the possibilities of scientific visualization and the user's task respectively. When interacting with the data the amount of possibilities which denote its functionality has to be represented (see also 2.6). Applications for experts exploit the real-world knowledge of the user which intuitively leads to the right way of interacting with the data whereas in virtual environments for training purposes functionality has to be represented in a perceivable way. There exist two main ways in VEs to show functionality to the user. One is to offer static menus which pack the whole set of operations that are applicable to the data sets. It is obvious that there are plenty of different possibilities to visualize these menus. When choosing this type of functionality representation the application designer and the programmer have to find the most suitable way to do this which is a really tough job. Problems which occur with those static menus are related to the limited interaction space of the displays systems and the uncomfortable usage when clicking through menu levels. It has been proven that it is a much better strategy to ask the data set what its functionality rather than to try to address a certain functionality with a selected tool. Then the data set's answer can be displayed as a menu again which is fixed positioned somewhere in the VE or attached to the user's gaze or hand[9, 7].

The *environment representation* reflects the ambience the users are working in. These representations can either be an operation theatre for surgeons, a lecture room for a professor and the students or a laboratory for a group of engineers. Environment representations are able to increase the feeling of

immersion as the users feel more comfortable in their natural working environment than in an abstract one. Especially when using virtual environments for training purposes environment representations facilitate to transfer the learned in order to repeat it in real world.

The *virtual input device representations* reflect the active physical input device the user has chosen. These representations usually are virtual coloured rays when using the stylus or the multiple button devices. These rays enable the user to see where the physical input device or the hand points to. These representations facilitate the selection process.

The *virtual tool representations* reflect the active tool a user has chosen. These representations are 3D icons which are connected to the physical input device in use. Thus they follow the movements of the physical input devices or hands. With the help of these tool representations the user is aware of the possibilities of the active tools at any time.

2.5 The Application+Interaction Space

The Application+Interaction space describes how users interact, with each other and the data set of interest, collaboratively in the virtual environment. In order to find the best interaction we first have to understand the low-level makeup of interaction. Therefore we have to narrow down interaction tasks and to find interaction templates which are combinable to form more complex interactions.

2.5.1 Awareness-Action-Feedback Loops (AAF)

Awareness-Action-Feedback loops denote such interaction templates. These AAF loops give us the possibility to understand and analyse very tiny steps in interactions.

2.5.2 Autonomous AAF Loop

Before explaining complex collaborative interactions we start with autonomous interaction (see Figure 3).

The autonomous AAF loop is divided into four blocks. The first two blocks belong to the awareness phase where the user starts with proprioception as it was defined by Mine[9]. The proprioception lets the user be aware where s/he stands and looks to, the position and orientation of body parts like arms, hands and fingers and everything that is needed for interaction. This means that the user perceives itself in relation to the environment. The next step is to be aware of the physical input devices held in the users hands and the virtual tool representations connected

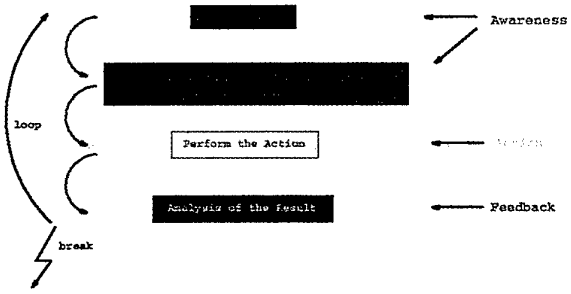


Figure 3: The Autonomous Awareness-Action-Feedback Loop.

to them. The position and orientation of the virtual data set is perceived in this phase as well. After the user is aware of the representation components and itself the action phase follows. This action can simply be to move the hand together with the physical input device. After the action phase the feedback phase follows. This feedback is meant to be action feedback without it would not be possible to analyse the result of the action. In this case the user perceives the movement of the virtual tool representations as s/he moved the input device together with the hand. After the perception of the status of the situation the user has to decide whether the task is completed and therefore wants to break the loop or whether the task is not completed yet and therefore prepares for the next action. We exemplify the AAF loop for the real scenario of a carpenter who wants to pound a nail into a piece of wood with a hammer. The steps of the AAF loop are:

1. **Proprioception** → *Awareness*
   Where am I ? Where do I look at ? Where are my hands, my fingers ?
2. **Perception of the physical/virtual input device and data set** → *Awareness*
   Where do I hold the stylus ? Is the hammer connected to my hand ? Where is the piece of wood ?
3. **Perform the action** → *Action*
   Interaction of human body (hands, fingers etc.) and physical input device. Position the nail on the wood and position the hammer !
4. **Result Analysis** → *Feedback*
   Perceiving the status of the situation. Perception of position, orientation and status of the virtual data and input device. (e.g. Did the data set allow to the operation ? Is the nail positioned

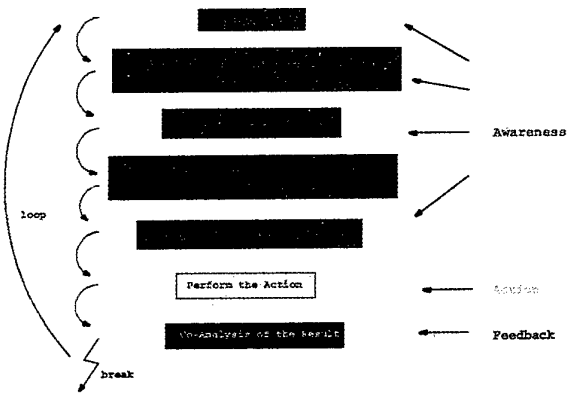


Figure 4: The Collaborative Awareness-Action-Feedback Loop.

correctly ? Is the hammer in place and ready to pound ?)
Depending on the status return to step 1. and proceed or break the loop (e.g. I am not ready yet so proceed with pounding the nail !)

5. Repetition of steps 1/2/3/4.

2.5.3 Collaborative AAF Loop

Collaborative Awareness-Action-Feedback loops are of the same structure as the autonomous AAF loops (see Figure 4).

The main difference between them is that the collaborative AAF loop has to address collaborative requirements that are necessary when working in a team. Again the collaborative AAF loop starts with the proprioception block and the perception of the own physical input devices and the virtual tool representations. After this but still in the awareness phase the user perceives the co-presence. It is comparable to proprioception but now information about the remote partner is queried like: Where is my partner, where does he look to, where are his hands, fingers etc.. Similar is the perception of the physical input device and the virtual tool representations together with the virtual data set. An interesting component represents the perception of co-knowledge and co-status. It is often not sufficient to know where you and your partner are located and where the object and the tools are when working in a team. We found out that knowing that your partner is aware of you is one of the most important steps in the awareness phase. To know that your partner is aware of what you are intending to do and how do you want to achieve this is essential for team work. Everything that supports this type of awareness increases the amount of collaboration. While perceiving the co-status the users check the situation. For the confirmation of this status check the users can do this by voice or with help of a gesture like "thumbs up". The action and the feedback phase abut to the already explained of the autonomous AAF loop. In order to apply the collaborative AAF loop to a real scenario we assume two carpenters who again want to pound a nail into a piece of wood with a hammer. One carpenter holds and positions the nail on a piece of wood and the other carpenter pounds the nail with a hammer. We are then able to describe the whole interaction task from the sight of the carpenter who holds the hammer like done in the following.

1. **Proprioception** → *Awareness*
   The same as before (see AAF loop).

2. **Perception of the physical/virtual input device and data set** → *Awareness*
   The same as before (see AAF loop).

3. **Perception of co-presence** → *Awareness*
   Where is my partner ? Where are his hands and fingers ? Where does he look to ?

4. **Perception of co-physical/co-virtual input device and data set** → *Awareness*
   Where does my partner hold the nail and the wood ? How is the relationship between nail and wood ?

5. **Perception of co-knowledge and co-status** → *Awareness*
   Is my partner aware of me ? Does he know where I am, where I am looking to and where I hold the hammer ? Does he know what I am doing and what I want to do ? Is everything ready now ? Confirmation of the status check by voice or "thumbs up".

6. **Perform the action** → *Action*
   The same as before (see AAF loop).

7. **Result Analysis** → *Feedback*
   The same as before (see AAF loop).

8. The steps 1. to 7. are repeated until the task is finished.

2.6 Operations, Metaphors, Interaction Techniques

Awareness-Action-Feedback loops like shown in the Figures 3 and 4 are templates. With the help of operations, metaphors and interaction techniques it is now possible to give those templates a "face". This means that depending on the user's subtask the appropriate operations, metaphors and interaction techniques have to be chosen for each action.
Operations defined in our taxonomy provide the means for supporting manipulation of virtual data and shared

manipulation between remote participants. They describe what can be done with the virtual data in terms of how the data can be explored. They can be data independent (i.e. basic operations such as selecting), or data dependent (i.e. slice through a 3D volume of data).
Metaphors for interaction and collaboration make use of everyday interaction and collaboration paradigms to provide intuitive ways of interaction in virtual environments (i.e the metaphor of working around a table). We distinguish between three different kinds of metaphors. *Stand-alone Metaphors* such as walk, fly and teleport, directly use or extent real-life paradigms to allow navigation through a virtual environment. *Content specific metaphors* that allow the user to focus on the part of data set of interest, look closer, hear/touch interesting subpart, as well as additional ones like play video/TV, search information library, can also be adapted from real-life paradigms. *Collaborative Metaphors* are visual and verbal communication between users and sharing viewpoints of participants. Finally, *interaction techniques*, complement the metaphors by determining how to support and implement the different types of operations[7].

3 Application Design

To design a collaborative virtual environment that supports the above requirements, we carefully studied all the issues mentioned in earlier sections of this paper, in order to select the most appropriate representation components, metaphors, operations and interaction techniques. The task description is as follows:
Two users, a medical professor and a medical student work together on a virtual human data set. They stand opposite each other around a table. They are able to walk around the table and to have a look from the other side onto the data. The data set consists of a human skin and an underlying skeleton and heart model. Both users are able to cut the skin in order to see the underlying bones and inner organs, to pick bones and to drag them. The data set is used for anatomical education. Names of all bones can be queried, test scenarios, where a set of bones has to be inserted into the skeleton, can be uploaded. The two users are equal in their possibilities to work on the data set.
After the UTA and the definition of the UNS we came up with the following application design (see Figure 5).
Generic operations such as selecting, zooming, translating, pushing, dragging, grabbing, highlighting and content specific ones, such as labelling of parts of the data sets, cutting, slicing planes, starting/ending video conferencing, were included in the design of the system. We decided to use menus and virtual pick-rays as interaction technique to apply the desired operations to the data sets. Therefore the generic operations are applied using a fixed toolbar with a rotate tool, translate tool, zoom tool, drag and push tool. The content specific operations allow slicing of the 3D representation of the patient's data. These operations are applied by calling an Object bound ring menu. The toolbar is fixed whereas the ring menu, bound to the object, disappears when an operation has been selected. Additional content specific operations for the real patient data sets are colour lookup sliders, compass to obtain the orientation when slipping into the data set, slicing and clipping planes. For the skeleton model content specific operations for material change and fade, and wire-frame and gray value windows are available. Additional operations include viewing of labels bound to different bones, or of animation of the virtual heart model. Additional visualization of interesting medical information is at the user's disposal. Rendering of video sequences in mono or stereo on virtual Screens is also part of the system, to allow video sequences of endoscopic recordings of the stomach or the esophagus to be played at will. As interaction devices in our prototype we use tracked Crystal Eyes shutter glasses, a Polhemus stylus, and for two handed interaction, a three button tool also tracked by the Polhemus Fastrak system. In order to enable teamwork we implemented the following metaphors:

- ring up the remote partner
- join a remote session
- share a tool
- face-to-face communication
- tug of war

The ring-up and join session metaphors were implemented by providing a session name. As soon as the user connects to a session a whole copy of the virtual scene provided by the others is transferred to the local site. In the same moment a video/audio connection to the other Responsive Workbench is established (see Figure 5). The video screen with the remote partner provides the content specific operation to mute or disconnect this video/audio conferencing depending on user's wish. To enable collaborative manipulation of the data, the generic toolbar is distributed together with the patient's body and skeleton model. The content specific operations are also shared since there are bound to the shared data sets. The metaphors we make use of in the collaborative case are the face-to-face communication and mirrored viewpoint or sharing viewpoint (look through other's eyes and/or look over other's shoulder). Finally for the collaborative manipulation we used the tug of war metaphor (see Figure 5).

3.1 Technical Details

For rendering two SGI ONYX IR2 workstations are used with two graphics pipes and six R12000 processors each. Electromagnetic Fastrack tracking systems

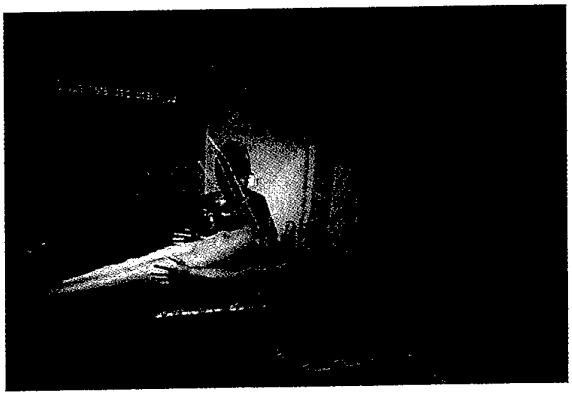


Figure 5: Image taken from a current session. In order to take an image of the session we had to render this scene in mono.

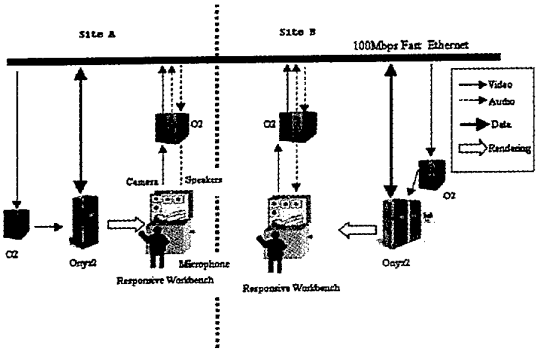


Figure 6: Schematic of the used setup.

from Polhemus are used to track the head and the two input devices, a Polhemus stylus and an own built three button tool. For communication purposes wireless microphones and headphones are available. The video and the audio conferencing is handled by two O2 workstations. Video streams in PAL resolution are grabbed directly from the infra-red video camera, compression using motion jpeg compression and sent over the fast ethernet network to another O2 workstation. There the stream is decompressed and fed into the DIVO boards of the ONYX. The same O2 which handles the video conferencing manages the audio conferencing. The audio stream grabbed from the wireless microphones is compressed and then send to the other O2 where the headphones are plugged in. The software framework we are using is AVANGO developed by GMD. It combines the familiar programming model of existing stand-alone toolkits with built-in support for data distribution that is almost transparent to the application developer. A detailed description of the toolkit and the way distribution is implemented can be found in [10]. A schematic of the built setup is shown in Figure 6.

4 Conclusions and Future Work

We presented our vision in creating Collaborative Virtual Environments that provide distributed collaborative teams with a virtual space where they could meet as if face-to-face, coexist and collaborate while sharing and manipulating in real time the set of virtual data of interest. We discussed the issues involved in bringing together Human Computer Interaction and Human to Human Communication, focusing on projection-based Virtual Environment systems.

The initial evaluation of the prototype was based on heuristic analysis [5] and we are planning to extend it to detailed user-task and ergonomic analysis [5].

References

[1] T. Capin, I. Pandzic, N. Magnenat-Thalmann, and D. Thalmann. Avatars in networked virtual environments. *John Wiley and Sons*, 1999.

[2] C.A. Ellis, S.J. Gibbs, and G.L. Rein. Groupware - some issues and experiences. *Communications of the ACM*, 34(1):38–58, 1991.

[3] I. Foster, J. Insley, C. Kesselman, and M. Thiebaux. Distance visualization: Data exploration on the grid. *IEEE Computer*, 32(12):36–43, December 1999.

[4] B. Froehlich, S. Barrass, B. Zehner, J. Plate, and M. Goebel. Exploring GeoScience Data in Virtual Environments. In *Proc. Visualization 99 (to be published)*, 1999.

[5] J.L. Gabbard, D. Hix, and J.E. Swan II. User-centered design and evaluation of virtual environments. *IEEE Computer Graphics and Applications*, 19(6):51–59, November/December 1999.

[6] G. Goebbels, W. Frings, T. Eickermann, F. Hossfeld, and S. Posse. Global broadcast - supercomputer-enhanced functional mri of the human brain. *In IEEE Concurrency*, 8(1):11–13, Jan.-Mar. 2000.

[7] G. Goebbels, V. Lalioti, and M. Goebel. On collaboration in distributed virtual environments. *In Proceedings of HC-2000 - Third International Conference on Human and Computer*, Sep. 2000.

[8] V. Lalioti, F. Hasenbrink, and C. Garcia. Meet.me@cyberstage: towards immersive telepresence. *Virtual Environments'98, 4th Eurographics Workshop, Germany*, 16-18 June 1998.

[9] M. Mine, P. Frederick, Jr. Brooks, and C. Sequin. Moving objects in space: Exploiting proprioception in virtual-environment interaction. *Proceedings of SIGGRAPH 97, Los Angeles, CA*, 1997.

[10] H. Tramberend. Avocado: A Distributed Virtual Reality Framework. In *Proc. of the IEEE Virtual Reality*, 1999.

Distributed Collaborative Virtual Environment: PaulingWorld

Simon Su[1], R. Bowen Loftin[2], David T. Chen[1], Yung-Chin Fang[1], Ching-Yao Lin[1]

[1]Department of Computer Science
University of Houston
Houston, Texas, USA

[2]Virginia Modeling Analysis & Simulation Center
Old Dominion University
Suffolk, VA 23435

{ssu, chingyao}@cs.uh.edu, bloftin@odu.edu, dave@chen.net, yfang2@bayou.uh.edu

Abstract

This paper introduces Distributed PaulingWorld, a Distributed Virtual Environment application that supports collaborative visualization of molecular structures among multiple users within the same virtual environment. All the participants in the virtual environment have the same level of interaction in the application. In the application, a virtual menu that is attached to the left hand of the user is used to manipulate the molecule and the environment. The user that has the virtual menu has total control of the environment and the viewpoint of the users in the virtual environment. However, the virtual menu can also be transfer to another user in the virtual environment. Only the user that has the menu can chose to transfer the menu. At that point, the other user, upon receiving the virtual menu, will have the capability to manipulate the molecule and the virtual environment. Users are represented by avatars to indicate their location within the virtual environment.

Key words: Distributed Virtual Environment, Virtual Reality, Responsive Workbench, Collaboration.

1. Introduction

An individual computer system can no longer provide sufficient computing power to support the increasing requirements and complexity required in creating a realistic Virtual Reality application. Even on some single user Virtual Reality applications, multiple computer systems are required to create a Virtual Reality application that looks accurate and behaves realistically. Single user Virtual Reality applications have benefited from distributing their sub-processes on different processors to increase their performance. In a Distributed Virtual Reality application, multiple computer systems are used to accommodate multiple users regardless of their locations as long as those computer systems are networked. This communication will provide collaborators with a tool to work together without having to be physically present in the same location.

PaulingWorld (PW) is a Virtual Reality (VR) application that simulates and visualizes molecular structures [4]. It also supports acceptable soft real-time interaction and manipulation performance. PW uses static local two-dimensional control widgets to interact with molecular data. PW allows one user to examine the structure of a molecule via five different representations: ball-and-stick, vanderWaals' spheres, coded sticks, backbone, and icons that replace repetitive structures. Figures 1 and 2 show snapshots of the application using the vanderWaals' representation and partially expanded icons that replace the repetitive structures. The user is free to fly through the virtual environment while examining the molecule representation from different viewpoints. The application also allows the user to scale, translate, rotate, or attach the molecule to his hand to inspect the molecule at different levels of details.

Distributed PaulingWorld (DPW) is a Distributed Virtual Environment (DVE) application that allows more than one distributed operator to interact with the same molecular structure by sharing the same virtual world. DPW allows collaborative visualization of a molecular structure among distributed users. DPW introduces a multi-user mode into PW described in the previous paragraph.

PaulingWorld allows a single user to visualize and investigate in detail the structure of the molecule. However, if the user wants to conduct a collaborative study with another person, the user will be limited by the functionality provided by PaulingWorld. It will also be impossible to share a finding with another person since the user cannot take a snapshot of the view of interest at that moment and share the findings with a second person. DPW provide a perfect solution to the collaboration problem in PaulingWorld by supporting multiple users in the virtual environment. In addition to provide support for multiple users, DPW can also bring together users that are physically dispersed into the same virtual environment without having the users to travel to the same physical location.

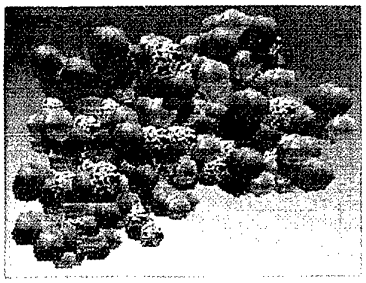


Figure 1 vanderWaals' Representation

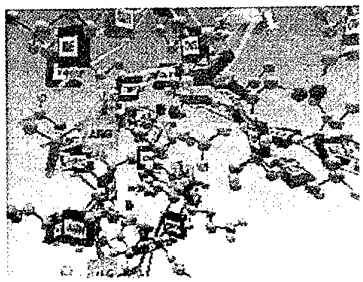


Figure 2 Iconic Representation

2. Related work

Research laboratories funded by both the government and private institutions have developed several practical and promising Distributed Virtual Reality applications [1][5][7][9][11][12][13][14][16][17]. Most Distributed Virtual Reality applications have some common properties. They are comprised of computer systems located at the same site or at geographically distant sites that are networked together, they use multiple processes, and they are used simultaneously by multiple people. The users interact with one another, and they are represented by an abstract representation to notify each other of their positions in the virtual environment. Un-Jae Sung et al [15] outlined some of the general characteristics of a DVE application. In general, DVE applications can be classified into large-scaled and small-scaled applications. A large-scaled DVE application may consists of several hundred nodes or participants, whereas a small-scaled DVE may consists as few as two participants within the same virtual environment.

Stytz at the Air Force Institute of Technology has done much work involving large-scaled DVE training applications that can support hundreds of participants in shared virtual environments [10]. The Synthetic Battlebridge gathered information from both the computer generated actors and human participants in real time and rendered a 3D image of the battlespace and its contents. This DVE application uses the Distributed Interactive Simulation (DIS) protocol [2] to manage its complex and active virtual environments. The DIS protocol governs the communication between hosts participating in the virtual environment.

Close Combat Tactical Trainer (CCTT) is another large-scaled DVE joint US Army-Loral project [14]. CCTT is a US Army training program that will help train ground combat tank and mechanized infantry forces within a realistic virtual environment. This DVE application also utilizes the DIS protocol to manage its complex and real time virtual environment. The simulator and individual workstations exchange data about their state information with respect to the virtual environment over the Fiber Distributed Data Interface (FDDI) using the DIS protocol. This application can support up to several hundreds of manned participants, computer-generated forces, and simulated vehicles.

In a more recent work by C. R. Karr et al [5], Synthetic Soldiers is a US military Joint Simulation System that was intended to create a single distributed virtual environment. The system is intended to provide joint training for all four branches of the armed services. As with most large-scaled DVE, Synthetic Soldiers also employs DIS to manage the communication of the hundreds of entities within the virtual environment.

Other than military research projects, most of the research done by academic institutions can be classified as small-scaled DVE. R. Bowen Loftin's work on the Hubble Space Telescope (HST) training project demonstrated a cross continental collaborative training in a shared virtual environment by astronauts in Houston and Darmstadt, Germany [3]. The virtual environment consists of a model of Space Shuttle payload bay and the HST. In the application, the training took the form of a simple extra vehicular activity (EVA) simulation that enable two astronauts on opposite sides of the Atlantic ocean to train within the same virtual environment. During the training, the two astronauts practiced the changeout of the HST's Us Sol ar Arr ay Di ve Electronics (SADE) and real time hand off of the SADE within the virtual environment. The exchanging of state data was managed by IGD-developed communication software, and the virtual environment was rendered by NASA-developed graphics software. An Integrated Services Digital Network (ISDN) line was used to connect the sites together. Since absolute synchronization of the participants was required, no dead-reckoning algorithms were used in the application. A duplicate copy of the 3D environment database was also kept at all participants site to minimize the network traffic to the state change among the participants of the virtual environment.

Leigh's wo Us wor k in Col labor a ti ve Archit ect ural L Via Immersive Navigation (CALVIN) shows the use of a DVE application to perform an architectural design and collaborative visualization [6][7]. In this DVE application, Leigh emphasized the use of heterogeneous perspectives in viewing an architectural design to aid in the design and the collaborative visualization processes. With heterogeneous perspectives, CALVIN also demonstrated the use of virtual reality technology in the active design phase rather than the just as a walkthrough of the finished design.

Mourant's work Us work in the Dstr ibuted Dri ving Si provided another example on a small-scaled DVE application [11]. Distributed Driving Simulator simulated the driving of a multiple driver within the same virtual environment. As in the case of HST training program, no dead-reckoning algorithms were used since the state change of one driver must be propagated immediately to the other driver to simulate a real time driving simulation. To minimize the network traffic, duplicate databases for the 3D environment and vehicles were also stored at the participants' local site'

Concurrency control within the shared virtual environment is also an important issue that needed to be addressed in a DVE application. The Collaborative Immersive Architecture layout (CIAO) paper described how concurrent actions are coordinated in a multi-user DVE application [15]. It achieved optimal response and notification time without compromising consistency through a new multicast-based, optimistic concurrency control mechanism.

3. Hardware and Software Environments

DPW is currently implemented between sites that have interactive workbenches. At each site, an SGI Onyx2 with multiple graphics channels drives a projector that produces display on the workbench. Tracking of the participants are accomplish by using Polhemus Fastrack™ each with a stylus and two other sources. Both the user's hands and Us hand s andt he viewpoi nt a interactive rates.

Although we chose workbench as the display device, the application can easily be modified to use homogeneous socket communication protocol to display on a multiple-wall CAVE™ display device or a head mounted display device. The use of Polhemus Fastracks™as the tracking device can also be replace with Ascension's Flock of Bi Us Flock of Bird s tr VrTool was the software toolkit used to develop this application [8] (not to be confused with Vr-tools developed by Christian Michelsen for NorskHydro). Figure 3 shows the workbench setup that was used in the application.

4. Application Design

4.1 Application Architecture

The DPW application is controlled by a main VrController process that manages and synchronizes all the states among the participants of the virtual environment. In addition, all the processes on a participant site are managed by their own local VrController. The main VrController is responsible to process and communicate with all the local VrControllers running at the participant site. Figure 4 shows the connection between the local VrController and all the processes at a participant site.

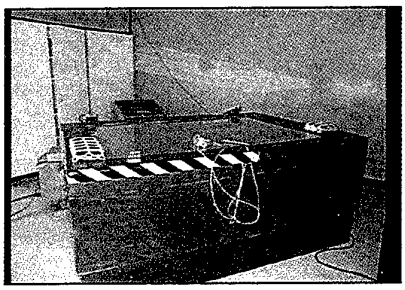


Figure 3 Workhench and Polhemus Fastrack

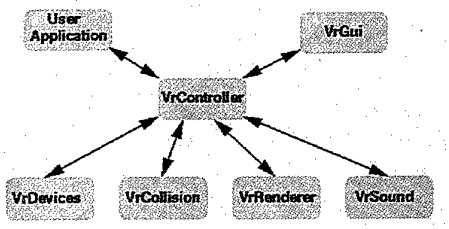


Figure 4 Application Architecture

Each participant process in the virtual environment can be divided into VrDevice, VrCollision, VrRenderer, VrSound, and VrController. The local VrController manages and synchronizes all other 5 processes of the local participant in the virtual environment. VrDevice is a process that is responsible for reading the raw data from the hardware devices and pre-process the data into the format that the application can use. VrCollision is responsible for detecting any collision among the objects that have been registered for collision within the virtual environment. VrRenderer is responsible for traversing through the scene graph that has been continuously updated by the VrController and render the object on the scene graph onto the display device. VrSound is responsible for playing any sound even in the virtual environment. The user application is the process that actually implements all the features of DPW.

Commands that the user executes will be process in this process and the state change is sent to VrController to be updated accordingly.

DPW employs a distributed database model to sustain the DVE. Every site retains a copy of all the models used in the DVE. This replication allows the DPW to be implemented over a regular Internet connection with no dedicated network connectivity with acceptable lag. Since all sites have copies of all the models, only the necessary state change information is propagated to the sites. State changes lost due to communication error is insignificant, since the actual state, not the relative state, are transmitted to all the distributed environments.

4.2 Collaboration Issues

All existing DVE applications allow certain degrees of collaboration among distributed users. Distributed users have been able to see and signal each other through visual gestures in a virtual world [15]. Virtual environments have been synchronized to render the same content in all sites. State changes in one site are propagated to the rest of the connected sites to refresh all local state. One of the most powerful features of a DVE application is the exchange of objects among distributed users. It allows true collaboration among distributed users [10]. However, most of the DVE work supports a sole manipulator and passive observers only. Allowing only one user to control the DVE imposes a great limitation on the level of interaction and collaboration.

Our system provides all users with an equivalent interaction priority in the shared virtual environment. This feature allows a more free and equal collaboration capability for all distributed users to share their opinions about certain objects in the DVE. At any moment, one user can interact with the visualized data by using the two-dimensional control widget while the other distributed user can observe the manipulation process. The current manipulator of the DVE can pass the control widget to other user in the DVE. This enables the receiving user to manipulate and interact with the DVE. Only the user who has the control widget in hand can transfer its control to other users in the DVE.

During the transfer process, the controlling user relinquishes the control to the other user in the virtual environment. After the transfer command has been issue, the controlling user's application will send a command to the other user's application to activat Us ap virtual menu of that user. The other user's application, upon receiving the command, will attempt to turn on the virtual menu of the user. If the virtual menu is successfully turned on, the application will then send a command to disable the controlling user's virtual menu.

The disable command must be acknowledged by the controlling user's application. Us application I f th was not received from the controlling user, the disable command will be resent until an acknowledgment is received. The protocol ensures that at least one user will have a virtual menu on the left hand. This guarantee is important because all interactions with the application are accomplished through the virtual menu. This protocol also guarantees that only one user can have access to the virtual menu at any given time.

4.3 Viewpoint Control

After the initial testing of the prototype application, we found that when distributed users were exchanging opinions about an object, they were occasionally discussing two different objects. This situation occurs because every user has different viewpoints. To eliminate this problem, we designed our system to provide a feature that will give all users a coherent viewpoint. The manipulator of the virtual environment can synchronize all viewpoints to a temporarily coherent viewpoint.

This methodology can guarantee that all distributed users are observing and discussing exactly the same object in the DVE. The controlling user can activate temporarily coherent viewpoints and restore the original viewpoints through the virtual menu. This feature enables the user who has the virtual menu on hand to show the other distributed users the viewpoint of interest and guarantees that distributed users are observing the object from the exact same viewpoint and discussing about the same issue. However, the user that has the control of the widget will also have to reset the viewpoint to its original settings before relinquishing the virtual menu to another user.

5. Future work and Conclusion

The current implementation of DPW supports two simultaneous users. However there is no pre-determined limitation of the number of simultaneous users that DPW can support. The user process of DPW can be easily modified to support more than two simultaneous users. The limitation on the number of users will depend only on the network bandwidth that is available to support an acceptable real time interaction among the distributed users.

Future development planned for this DVR application includes the verification of the usefulness of the features supported in DPW. We plan to run human subject to determine the usefulness of the viewpoint control and virtual menu transfer features.

DPW provides a homogenous collaborative working

environment for remotely located scientists to cooperate designing a new drug, new gas ...etc. With team members scattered all over the globe, remote collaboration should substantially reduce the turnaround time in the design phase. DPW can also be use as a distance-learning tool. Both an instructor and a student can be immersed in a virtual environment at the same time to examine an object. The teacher can illustrate the construction of a molecular structure in a way never before possible to the trainee even in a virtual world.

Acknowledgement

The first author would like to thank Dr. R. Bowen Loftin for his support and guidance over the past years.

This material is based upon work supported in part by the National Science Foundation under Grant No. NEC95-55682, NASA grant NAG9-985, and funding from the Institute of Somatic Sciences. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agencies.

Reference

1. A. Johnson, J. Leigh, and J. Costigan, "Multiway tele-immersion at Supercomputing 97", *IEEE Computer Graphics and Applications*, pp 6-9, (1998).

2. B. Blau, J. M. Moshell, and B . McDonald, "The DIS (Distributed Interactive Simulation) Protocols and Their Application to Virtual Environments," *Proc. Of the Meckler VR 93 Conf*, Mecklermedia Press, Westport, Conn,. (1993).

3. Bowen R. Loftin, "Virtual Environm 8V rtua aerospace training," *Proceedings of WESCON 1994*, pp. 384-387, (1994).

4. C. Dede, M.C. Salzman, and R.B. Loftin, 5Sci enceSpace: virt ual realiti es for l ear ri ng co mplex and abstract scientific concepts," *Proceedings of Virtual Reality Annual International Symposiom*, pp246-252, 271, (1996).

5. C. R. Karr, D. Reece, and R. Franceschini, 5Syrt hetic sd diers [milit ary trai ri ng si mlat or],," *IEEE Spectrum*, pp39-45, (March 1997).

6. J. Leigh, A.E. Johnson, "CALVIN: an immersimedia design environment utilizing heterogeneous perspectives," *Proceedings of the Third IEEE International Conference on Multimedia Computing and System*, pp. 20-23, (1996).

7. J. Leigh, A. E. Johnson, C. A. Vasilakis, and T. A. DeFanti, "Multi-perspective collaborative design in persistent networked virtual environments," *Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium*, pp. 253-260, 271-2, (1996).

8. LinCom Corporation, VrTool User's Guide and VrTool Application Programmer's Interface Functions, LinCom Corporation, 1020 Bay Area Blvd, Suite 200, Houston, TX 77058-2682, pp. 196-201, (1995).

9. Martin. R. Stytz, J. Vanderburgh, and S.B. Banks, 6The Sdar Syste m Modeler," *IEEE Computer Graphics and Applications*, pp. 47-57, (Sept.-Oct. 1997).

10. Martin R. Stytz, E. G. Block, B. B. Slotz, and K. Wilson, "The Syn 6The Synthet i c Battle bri dge: a large-scale Ves," IEEE Com IEEE Computer Gr Applications, pp. 16-26, (1997).

11. R. R. Mourant, N. Qiu, and S.A. Chiu, "A distributed virtual driving simulator," *IEEE Virtual Reality Annual International Symposium*, pp 208, (1997).

12. S. Stansfield, D. Shawver, N. Miner, and D. Rogers, "An appl #An applicat i on of shared vi rtual re situational training," *Proceedings of Virtual Reality Annual International Symposium 1995*, pp 156-161, (1995).

13. S. Stansfield, D. Shawver, and A. Sobel, / Med Si m a prot otype VR syste m for trai ri ng medical first responders," *Proceedings of IEEE Virtual Reality Annual International Symposium 1998*, pp 198-205, (1998).

14. T.W. Mastaglio, and R. Callahan, "A large-scale complex virtual environment for team training," *Computer*, pp 49-56, (July 1995).

15. Un-Jae Sung, Jae-Heon Yang, and Kwang-Yun Wohn, "Concu %Concu rrency c ont r ol in *Proceedings of IEEE Virtual Reality 1999*, pp 22-28 (1999).

16. V. D. Lehner and T. A. DeFanti, "Distributed virtual reality supporting remote collaboration in vehicle design," *IEEE Computer Graphics and Application*, Vol. 17, Issue 2, pp. 13-17, March-April (1997).

17. W.D. McCarty, S. Sheasby, P. Ambrun, M. R. Stytz, and C. Switzer, "A virtual cockpit for # distributed interactive simulation," *IEEE Computer Graphics and Applications*, pp49-54, (Jan. 1994).

The Study of a Computer-Supported Collaborative Virtual Design System with VRML-JAVA-EAI

Hao-Ren Ke[1], Hung-Chun Chiu[2], Chien-Hung Tsao[2], Zen-Chun Shih[2]

[1]Library, National Chiao Tung University, 1001 Ta-Hsueh Rd., Hsinchu, Taiwan, R.O.C.

claven@lib.nctu.edu.tw

[2]Department of Computer and Information Science, National Chiao Tung University, 1001 Ta-Hsueh Rd., Hsinchu, Taiwan, R.O.C.

{gis87547, gis87563, zcshih}@cis.nctu.edu.tw

Abstract

This paper proposes a computer collaborative virtual design system (CSCVD) implemented by VRML, Java, and EAI. This system is a WWW-based client-server system, and messages are encapsulated in Protocol Data Unit (PDU) to transmit. PDUs are delivered via TCP. To overcome the shortcomings of TCP, this paper proposes a buffering method. The dead reckoning technology is also employed to predict the future positions of objects in order to reduce packets transferred on the network.

Keywords: Computer-Supported Collaborative Virtual Design (CSCVD), VRML, JAVA, External Authoring Interface (EAI)

1. Introduction

In fields like stage lighting, architecture, and industry design, the interaction, collaboration and communication among people (for example, designers and designers, customers and designers) are beneficial to create new ideas, reduce the time of design cycle, and design perfect products [1][6]. With the great improvement of the network speed, and the CPU power and graphical capability of computers, it is gradually becoming possible to develop systems participated by multi-users via networks; consequently, now is the right time to develop Computer-Supported Collaborative Virtual Design (CSCVD) systems that encourage multiple users to share their ideas and participate in design processes without the limitation of location and time.

In the literature, several researches have proposed networked multi-user VR systems before; however, most of these systems are proprietary, and as a result, they are not very portable, extensible, and flexible. On the other hand, a few networked multi-user VR systems with an open architecture have also been proposed. DIVE is one representative system, which is a VRML-based system [4]. DeepMatrix [7] is another example, which is implemented with VRML, JAVA, and EAI (External Authoring Interface).

The purpose of this paper is to explore how to apply the techniques of computer graphics, virtual reality, and computer networks to the traditional collaborative design process, and present a CSCVD prototype system.

Similar to multi-user VR systems, a CSCVD system has to deal with the following three issues: (1) the rendering of virtual scenes and objects, (2) the control of virtual objects, and (3) the network communication among participants. Taking into account issues like interaction, real-time, portability, extensibility, and data sharing, we implement this system by VRML [10], JAVA, and EAI [11]. VRML is a standard for modeling 3D virtual worlds under WWW and we use VRML to render virtual scenes and objects. EAI is an interface to external application programs for controlling the local scene data of a VRML environment and we choose Java to implement EAI. As the current version of VRML does not support interaction between multiple users, we use JAVA to enable network communication among participant computers.

Our system is a WWW-based client-server system. Users taking part in a design process can connect to the system by a VRML-enabled WWW Browser. In our system, users can manipulate objects of a virtual world in many ways they want, and the manipulation of an object by a client will be seen simultaneously by all other clients. Users can also add objects into a virtual world by using a model database stored in the server or uploading objects stored locally in their desktops. In addition, they can exchange and share ideas and experiences with one another by a chat room. With the functionality incorporated in our systems, the goal of cooperative design can be achieved easily.

The organization of this paper is as follows. Section 2 describes the architecture of our CSCVD system. Section 3 focuses on the functions of our CSCVD system. Section 4 introduces the prototype system and preliminary simulation result. Section 5 gives the conclusion.

2. System Architecture

2.1. Overview

The CSCVD system proposed in this paper is a client-server VRML-JAVA-EAI system. Users participate in the system through VRML-enabled WWW browsers. Messages are transmitted by TCP. Figure 1 illustrates the interrelation among all the techniques exploited in our system [2].

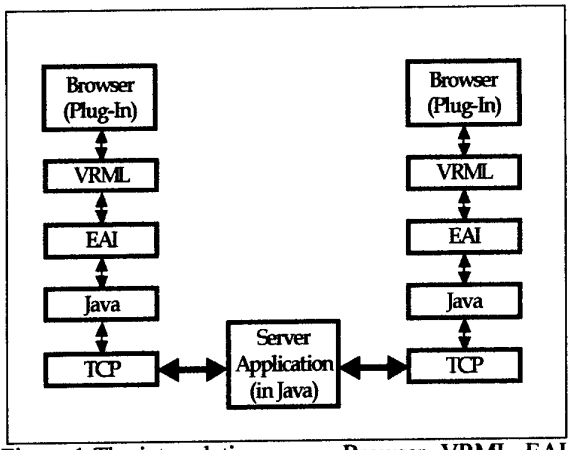


Figure 1 The interrelation among Browser, VRML, EAI, Java, and TCP

The primary tasks of a server include:

User authentication. If a user is a first-time comer, the server will request personal information from the user; otherwise, the server retrieves the user's information and usage history that is stored in a database maintained by the server. After a user logins to our system, the server will send the most up-to-date scene data to the client via HTTP.

User information maintenance. The server stores all the related information about users, including the IPs from which users connect to the server, and the avatars that are employed to represent users. In our implement, the H-anime format [5] is used to represent avatars.

Virtual scene maintenance. The server stores all the related information about a scene, objects in the scene and their associated properties. The server updates all the related information according to how users manipulate the scene data.

Message processing. The server is responsible for collecting messages sent by a client, and sending them to all other involved clients.

The primary tasks of a client include:

User interface. The user interface is responsible for accepting actions performed by users, sending messages to the server, receiving messages from the server, and invoking *EAIControl* to update the local scene.

EAIControl. EAIControl is an EAI-enabled Java class to manage the dynamic changes of a virtual world. EAI builds a bridge between a virtual world and external Java applets that manipulate it. The extensibility of our system is achieved by writing Java methods belonging to EAIControl. Most of the functions for collaborative virtual design are achieved by EAIControl.

User information and scene data maintenance. Similar to the corresponding tasks of the server.

Chat Room. The chat function is for user communication. It is responsible for sending chat messages to the server, and the server will relay messages to other clients.

2.2. Server Architecture

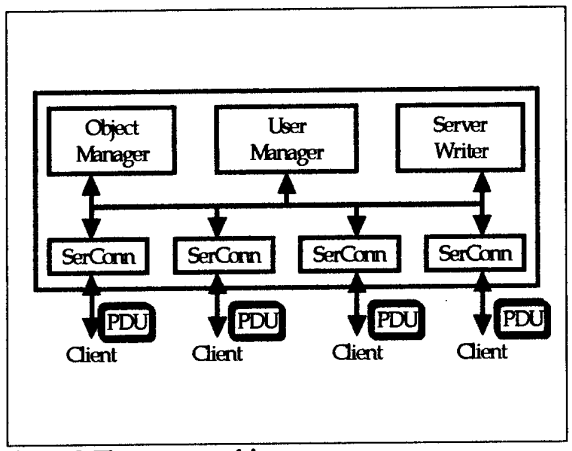


Figure 2 The server architecture

Figure 2 depicts the server architecture. The server is composed of four components: *Object Manager*, *User Manager*, *Server Writer*, and *SerConn*.

Object Manager. Our system stores virtual objects (e.g. a table, a chair, etc) in separate VRML files. From the server point of view, a virtual scene consists of instances of virtual objects, and Object Manager keeps the related information of each object instance in a scene, including an object name, the corresponding VRML file name, and its owner, position, moving direction, size, etc.

User Manager. User Manager keeps the related information of each participant, including the IP address from which the user connects to the system, user identifier, corresponding avatar model, and the object instances owned by the user.

Server Writer. Server Writer takes charge of the transmission of messages, which are encapsulated in Protocol Data Unit (PDU) [3], to clients. Our system transmits PDUs via TCP, which is reliable but slow. In order to expedite the process of message passing, PDUs are stored first in buffers managed by Server Writer, packed into a larger PDU at intervals, and then sent out. We further describe PDU and the buffering concept in Sections 2.4 and 2.5.

SerConn. SerConn is the primary control process to communicate with clients. Each time a new participant

connects to the server, a SerConn thread is created to communicate with the participant's client until the client disconnects. The tasks of a SerConn include: (1) receiving PDUs sent from clients, (2) performing necessary update according to the received PDUs, and (3) sending PDUs to the clients.

2.3. Client Architecture

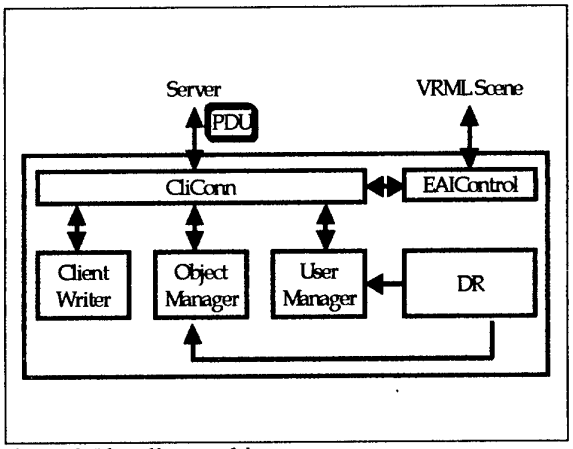


Figure 3 The client architecture

Figure 3 shows the client architecture. The client is composed of six components: *Object Manager*, *User Manager*, *Client Writer*, *CliConn*, *EAIControl*, and *DR* (Dead Reckoning [8]). The responsibilities of the client-side Object Manager, User Manager, and Client Writer are similar to their server-side counterparts, and the information stored in client-side and server-side Object/User Managers has to be synchronized. The following describes *CliConn*, *EAIControl*, and *DR*.

CliConn. CliConn is the primary control process to communicate with the server. The tasks of CliConn include: (1) sending PDUs to the server, (2) receiving PDUs from the server, (3) performing necessary update according to the actions performed by the user and/or PDUs sent by the server, and (4) calling EAIControl to update the virtual scene.

EAIControl. EAIControl facilitates virtual design. EAIControl is a Java class and consists of several methods, each of which conducts a design operation. CliConn invokes a suitable EAIControl method to conduct a design operation designated by a PDU. We further describe EAIControl in Section 3.1.

Dead Reckoning (DR). DR is an approach proposed in Distributed Interactive Simulation (DIS) [3] to forecasting the future position of an object. If the difference between the accurate and forecasting positions of an object is within a predefined threshold, no PDU for updating the object position is required to transmit; in this manner, the number of messages transmitted over the network can be reduced. In this system, we apply DR to forecast the position of objects in uniform motion and uniform acceleration motion.

2.4. Protocol Data Unit (PDU)

The concept of PDU was originally developed in DIS. As a standard packet format, PDU was used for communicating messages among distributed simulation systems. DIS proposed 6 classes and in total 27 kinds of PDUs. The PDUs designed by DIS are for military simulation and are very complicated; therefore, instead of using the original PDUs, we develop PDUs that meet the requirements of collaborative design. The PDUs proposed in the paper are divided into two classes: data transmission, flow control.

- Data Transmission: Chat PDU, File PDU, PositionUpdate PDU, OrientationUpdate PDU, AddObject PDU, DeleteObject PDU, AddAvatar PDU, DeleteAvatar PDU, and DirectionMove PDU.
- Flow Control: Login PDU, Logout PDU, Reconnect PDU, PDUPack PDU, Get PDU, and Release PDU.

Table 1 depicts the format of the *PositionUpdate PDU*. For the formats of other PDUs, please refer to [2] for details.

Content	PDU Flag	Time Stamp	Object Name	Position XYZ
Data Type	Integer	String	String	Float [3]

Table 1 The format of the *PositionUpdate* PDU

2.5. TCP Buffering

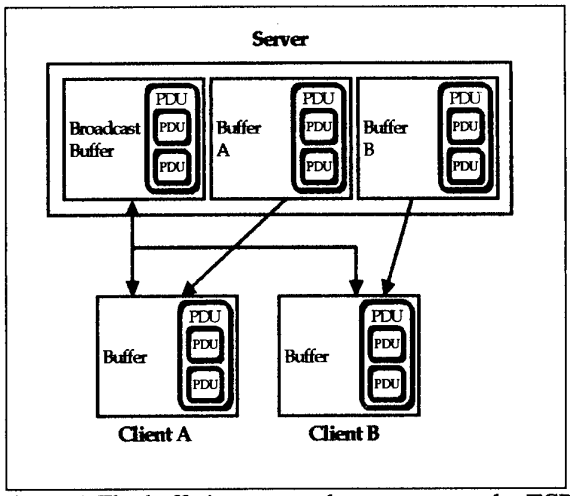


Figure 4 The buffering approach to overcome the TCP drawback

Our system uses TCP to transmit data, which is reliable but slow; furthermore, if TCP is used to transmit many small data in a very short period of time, it is neither efficient (the sender has to confirm the receipt of data) nor effective (the header of the TCP packet may be larger than the actual data). By our experiment, if TCP is used directly, on average only 30 PDUs can be

transmitted per second, which is unacceptable for a multi-user VR system. To overcome this drawback, we propose a buffering method. Instead of sending PDUs immediately, the system keeps PDUs in a buffer, packs PDUs into a *PDUPack* PDU at intervals, and then sends out the *PDUPack* PDU. This buffering approach is performed by the Server Writer and Client Writer. Figure 4 Illustrates the idea of buffering. From this figure, we can see that the Server Writer has a separate buffer for each client and has a broadcast buffer for transmitting PDUs to all clients.

2.6. System Operation Flow

In this section, we present the operation flow from the user viewpoint and system viewpoint.

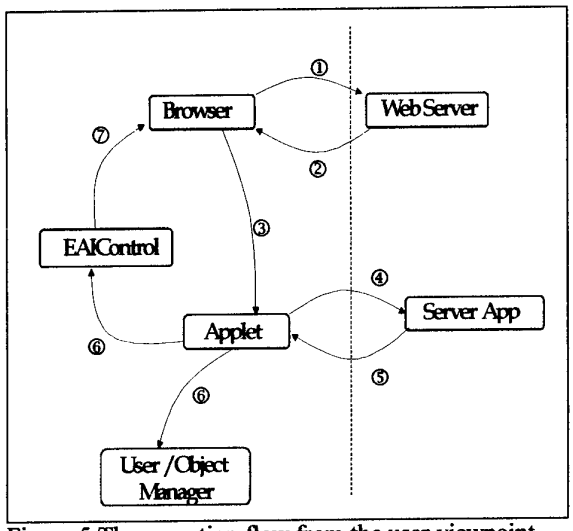


Figure 5 The operation flow from the user viewpoint

Figure 5 illustrates the operation flow from the user viewpoint:

1. A user participates in the CSCVD system by connecting to the Web server via HTTP.
2. In addition to an HTML file, the client downloads a VRML scene and the main Java Applet from the Web server.
3. The client's Browser invokes the VRML plug-in and executes the main Java Applet. The browser window shows the VRML scene, function buttons, chat room, and other menus, and the client waits for user actions.
4. The main Java Applet builds connection with the server and sends out PDUs according to user's action.
5. The server processes PDUs and sends out PDUs to those clients that should receive the PDUs.
6. The main Java Applets updates the User/Object Manager according to the PDU received, and invoke EAIControl.
7. EAIControl updates the VRML scene by invoking the EAI interface of the VRML plug-in.
8. Repeat 3-7.

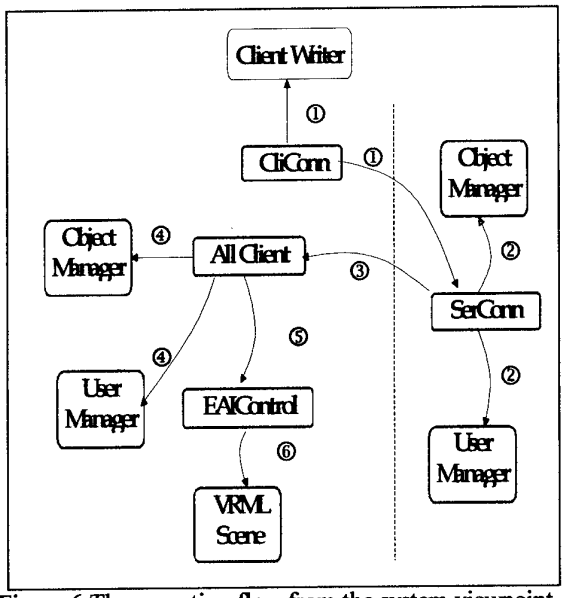


Figure 6 The operation flow from the system viewpoint

Regarding the operation flow from the system viewpoint, we take the update of an object's position as an example. See Figure 6 for illustration:

1. CliConn receives the action to move an object, creates a *PositionUpdate* PDU and stores the PDU in Client Writer. Client Writer packs the PDU along with other PDUs into a *PDUPack* PDU and sends *PDUPack* PDU to the server.
2. The server unpacks the *PDUPack* PDU and processes each PDU. When the server processes the *PositionUpdate* PDU mentioned in 1, Object Manager updates the object position.
3. The server has to inform all clients of the new position of the object; therefore, the server creates a *PositionUpdate PDU* and stores the PDU in Broadcast Buffer of Server Writer. Server Writer packs the PDU along with other PDUs into a *PDUPack* PDU and sends *PDUPack* PDU to all clients.
4. When a client receives the *PDUPack* PDU, it unpacks this PDU and processes each PDU. When the client processes the *PositionUpdate* PDU mentioned in 3, Object Manager at the client side updates the object position.
5. CliConn invokes the corresponding method in EAIControl to move the object.
6. EAIControl updates the VRML scene by moving the

object.

3. Collaborative Virtual Design

Based on the system architecture presented in the previous section, a CSCVD system is developed [9]. The purpose of this CSCVD system lies in facilitating the collaborative arrangement of 3D models and lighting, and enabling the communication among participants. In this section we briefly describe the functions of the system.

3.1. EAIControl

EAIControl is the kernel for implementing most functions supporting collaborative virtual design. Basically, EAIControl is a Java class inheriting the EAI's EventOutObserver interface. EAIControl is a collection of methods to manipulate objects or obtain the information of objects (see Figure 7). For example, we can invoke *EAIContol.getNodeTranslation(object_name)* to obtain the position of an object. Although the methods of EAIControl fulfill different functions, the underlying principle of writing methods is very similar, as illustrated in Figure 8, and we briefly describe the steps to write a method as follows. (1) Call getBrowser() to obtain the reference to a specific VRML scene. (2) Call getNode() to obtain the reference to a specific object that we want to manipulate. (3) Call getEventIn() or getEventOut() to obtain the events of an object. (4) Manipulate the object by using the events of an object.

```
public class EAIControl implements
EventOutObserver {
        public EAIControl( ) { }
        public void callback( ) { }
        public float[] getNodeTranslation( ) { }
        public void setNodeTranslation( ) { }
        public void addNewObjectNode( ) { }
        public void removeObjectNode( ) { }

}
```

Figure 7 The specification of EAIControl

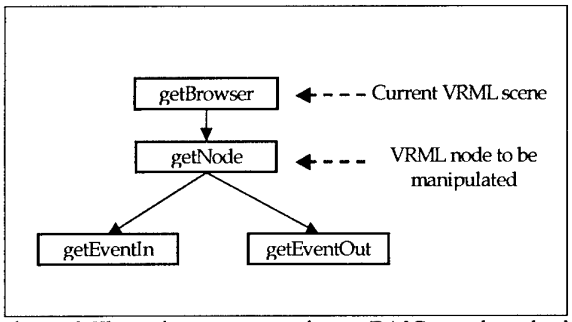


Figure 8 The naive steps to write an EAIControl method

However, in our implementation, we find that if we follow the aforementioned steps to write an EAIControl method, the performance is unsatisfactory because the references to the browser, objects, and events have to be obtained repeatedly on the fly. To improve the performance, we store the references to the EventIn and EventOut of an object into an array when the object is added into the virtual scene. While an EAIControl method is invoked to manipulate an object, it retrieves the object's references directly from the array. Figure 9 shows the modified steps to write EAIControl methods. To further enhance the performance, a hash table is employed to retrieve the reference arrays of all objects.

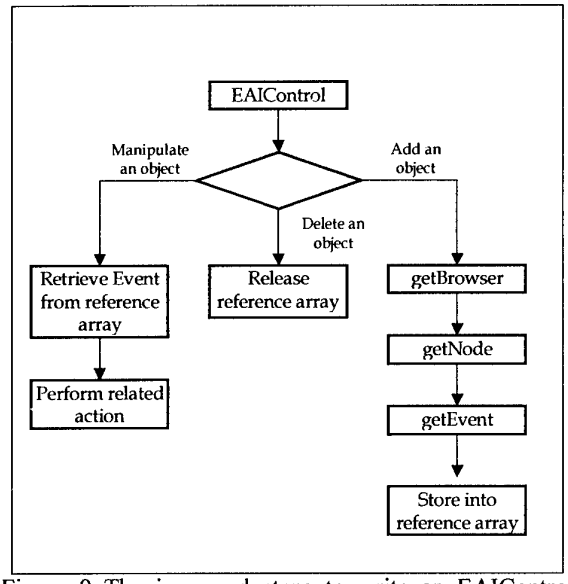


Figure 9 The improved steps to write an EAIControl method

3.2. Virtual Design Functions

Model Database. A model database stores objects that can be used by participants. The manager of our system can pre-load well-designed models into the database. In addition, users can upload models from their desktops, and in this way participants can share models they design. A Java Applet is employed to read a VRML model from a client, encapsulate the VRML model into a *File PDU*, and send the PDU to the server.

Object Selection. Before a participant performs any actions on an object, he/she has to select the specific object. We accomplish the selection of objects by using VRML's TouchSensor attached on geometry nodes and EAI's listening mechanism.

Basic Transformation. We design three kinds of interface for users to control the translation, rotation, and scaling of objects. First, users can input the precise values to control the transformation of an object; second, users can click function buttons to control the variation of an object from its current status; third, users can control the transformation by dragging an object.

Object Dragging. It's convenient for users to manipulate an object by dragging. We achieve the dragging of

objects by using VRML's PlaneSensor, SphereSensor and CylinderSensor.

Lighting Control. Our system provides users a mechanism to control the lighting of a scene. Users can manipulate spot and point light sources (by VRML's SpotLight and PointLight), fog effects, and viewpoints.

Advanced functions. In addition to the above basic functions, we devise a few advanced functions to facilitate the manipulation of objects. For example, the well known Copy and Paste functions are convenient for users to duplicate objects; the Group and Ungroup functions can treat many objects as a whole and manipulate them uniformly.

Chat Room. In a CSCVD system, communication among participants is very important. In general, communication channels can include image, video, audio, text, and among others. We implement a chat room in our system to transmit text messages among participants. *Chat* PDU is employed to carry chat messages.

3.3. Scene Loading and Storing

Usually, several runs are necessary for finalizing a design. Therefore, it is essential that a CSCVD system incorporates a mechanism to store a draft design for follow-up modification. Due to a few restrictions of VRML and EAI, we propose a method for loading and storing scenes. Basically, we have a non-VRML definition file to store the information of a scene, which is created by using the information stored in Object Manager. While a user wants to modify a previous scene, the system gives the user an empty VRML scene and adds nodes dynamically into this empty scene according to the corresponding definition file. This approach has three advantages: (1) it is easy to implement and maintain; (2) because the server has stored already the VRML files of objects, the non-VRML definition file for a scene only needs to store objects' VRML file names and coordinates, which results in a small file size; (3) by this approach, users can control every object in a scene, which is very difficult to accomplish by storing the whole scene as a VRML file.

4. Results

We have implemented a prototype system by using the ideas proposed in this paper. The server side can be executed on any machine that has installed a Web Server and Java Runtime Environment. The client side can be executed on any machine that has installed a Java-enabled WWW browser and VRML 2.0 Plug-in.

Figure 10 shows the current appearance of the CSCVD server. The upper-left part is the function buttons to start/stop the server, remove users, and add/remove objects; the middle-left window displays the actions performed by users and chat messages among participants; the lower-left part is the function buttons to move the position of objects; the upper-right and lower-right windows show the objects and users in a virtual scene, respectively.

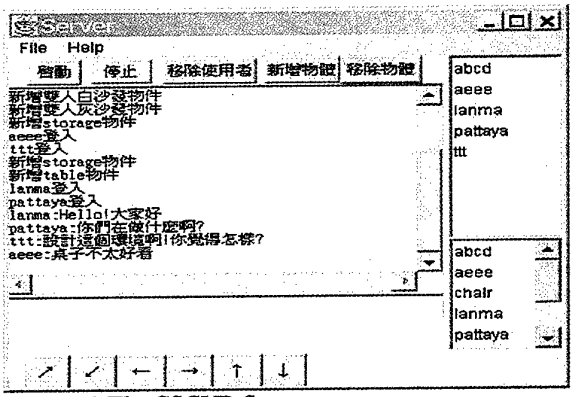


Figure 10 The CSCVD Server

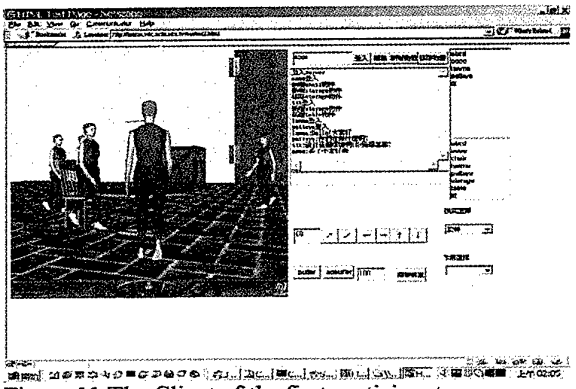


Figure 11 The Client of the first participant

Figure 11 and Figure 12 are the snapshots of the virtual scene from two participants' viewpoints. The left windows is the main window to show a virtual world; the right part is a chat window for users to share ideas and function buttons from which users can issue manipulation on scene objects, upload object models to the virtual world, identify other participants, etc. User can also manipulate objects directly via a mouse and keyboard.

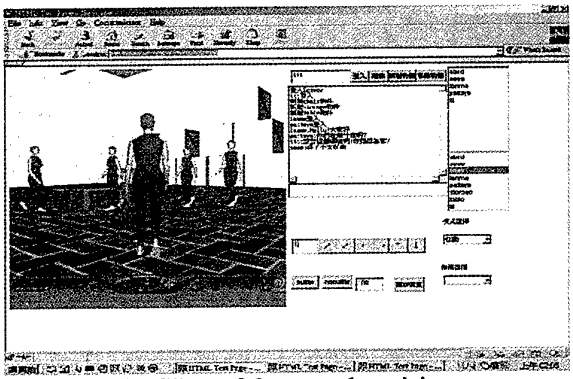


Figure 12 The Client of the second participant

To evaluate the improvement in TCP by using the buffering method, we performed a preliminary simulation. 3000 *PositionUpdate* PDUs were sent from

the server to a client. Two simulation tests were performed, one when the server and the client were connected by a LAN, and the other when they were connected by 56K modems. Table 2 shows the comparison of buffering and non-buffering method. We can see from this table that under the 56K modem and LAN networking environment, 10.19 times and 21.99 times improvement were obtained, respectively. In the future, we will perform detailed performance measure.

	Buffer	No Buffer	Improvement
56K Modem	10.015 Sec	102.19 Sec	10.19
	299.6 PDU/Sec	29.4 PDU/Sec	
LAN	4.54 Sec	99.82 Sec	21.99
	660.8 PDU/Sec	30.1 PDU/Sec	

Table 2 The comparison of TCP buffering/ non-buffering. 3000 *PositionUpdate* PDUs were sent.

5. Conclusion

This paper describes the implementation of a computer-supported collaborative virtual design system. This system is implemented by VRML, JAVA, and EAI. As VRML, JAVA, and EAI are open standards, our system fulfills the needs of portability, extensibility, and flexible. We refine the PDUs proposed in DIS to encapsulate messages transmitted in our system. A buffering method and dead reckoning are leveraged to overcome the drawbacks of TCP.

A prototype system has already been implemented, and more advanced functions will be incorporated into the prototype very soon. A preliminary performance measurement has been undertaken. With our system, cooperative design can be accomplished efficiently and effectively.

Acknowledgment

The authors would like to thank National Science Council for financially supporting this research under Contract No. NSC-88-2213-E-009-046.

Reference

1. Bridges and D. Charitos "On Architectural Design in Virtual Environments", *Design Studies*, 18(2): 143-154, 1997.
2. H. C. Chiu, H. R. Ke, and Z. C. Shih, "A Study on Distributed Multi-User Virtual Reality System," Master Thesis, National Chiao Tung University, 2000.
3. DIS, Standard for Distributed Interactive Simulation-Application Protocols", Draft standard from Institute for Simulation and Training, University of central Florida, 1994.
4. O. Hagsand, Interactive Multi-user VEs in the DIVE system", *IEEE Multimedia*, 31: 30-39, 1996.
5. The Humanoid Animation Working Group, http://ece.uwaterloo.ca/~h-anim/spec1.1
6. D. Marca and G. Bock, "Groupware: Software for Computer-Supported Cooperative Work," IEEE Computer Society Press, Los Alamitos, Calif., 1992.
7. G. Reitmayr, S. Carroll, A. Reitemeyer, M. G. Wagner, "DeepMatrix – An open technology based virtual environment system", *The Visual Computer*, 15: 395-412, 1999.
8. J. Towers and J. Hines, "Equations of Motion of Equa 2.0.3 Dead Reckoning Algorithm, " *10th DIS Workshop Proceedings*, 1995, pp.431-462.
9. C. H. Tsao, H. R. Ke, and R. C. Chang, "A Web-based Virtual Reality System for Real-Time Cooperative Design," Master Thesis, National Chiao Tung University, 2000.
10. VRML Standard Version 2.0, http://vrml.org/VRML2.0/
11. VRML External Authoring Interface Specifications, http://vrml.org/WorkingGroups/vrml-eai/

ICAT 2000

Shared Virtual Reality Interior Design System

Tomi Korpipää[123], Koichi Minami[3],
Tomohiro Kuroda[13], Yoshitsugu Manabe[13], Kunihiro Chihara[13]
[1] TAO Nara Research Center
[2] VTT Electronics, Oulu, Finland
[3] Nara Institute of Science and Technology
Tomi.Korpipaa@ele.vtt.fi, {koich-mi, tomo, manabe, chihara}@is.aist-nara.ac.jp

Abstract

Traditional methods in interior design usually lack depth and sense of realism, as well as require the designer and the client to meet in one place. These problems can be solved by utilizing shared virtual reality in the design process. This paper proposes an interior design system using remote heterogeneous virtual reality platforms. Using the system, the designer and the client can work together without the need to meet in the same place. The proposed system can be used to greatly enhance the feeling of presence.

To realize a useful shared virtual environment, a portable graphics engine is required to allow running the system in a number of hardware configrations, allowing using the resources available. Also, a smart network protocol is needed to ensure smooth operation and avoid unnecessary delays. This paper introduces a portable and configurable 3D engine developed for the system as well as a non-locking network protocol to realize the shared space.

Keywords: Shared, Virtual Reality, Network protocol, Interior design

1 Introduction

The idea for the system was initiated from the notion of hardships in modern interior design. Nowadays interior design is usually made by the designer and the client both being in the same place, which requires some traveling for at least one of them. Traditional methods also lack depth and sense of realism and require quite a bit of imagination to comprehend what the result actually would look like in the real environment.

To solve this problem, this paper proposes an interior design system using heterogeneous virtual reality platforms. Using the system, the designer and the client can work together without the need to meet in the same place. The designer can stay in his office and the client in a place convenient for him, for example nearest place offering a virtual reality platform, or even at his own home. Moreover, the designer can access extensive furniture database right in his office. This paper introduces a test arrangement of the proposed system.

The proposed system can be used to reduce the problems mentioned by greatly enhancing the feeling of realism. However, developing such a system has several aspects and a number of technical problems. The main topics in this paper are the graphics engine and the network protocol developed for the system.

2 System Overview

The system under development aims to realize a network protocol and a 3D graphics engine that allow the same virtual space be used in two or more remote systems even of significant performance difference. In a trial-and-error situation such as interior design, the network protocol has to be able to maintain the coherency of the virtual space dependless on what users are doing in separate environments. Graphics engine on the other hand has to be easily portable to different operating systems and visualization systems.

The system allows client users to perceive the same space at the same time, ie. in real-time. The users are also able to see each other as avatars, move inside the virtual space and communicate through the avatars. Users are also able to make modifications to the virtual space, such as add, remove and move the furniture, and the changes can be perceived by all clients almost simultaneously.

3 Test System Equipment

The test system arrangement uses two remote immersive Virtual Reality platforms. One end of the system is a CYLINDRA [1] at the Information Science Department building at Nara Institute of Science and Technology (NAIST) in Nara, Japan. The other end is an Immersive Multi-Display System in Telecommunications Advancement Organization of Japan Nara Research Center (TAO NRC) near Nara Institute of Science and Technology. These two Virtual Reality platforms are connected via a 150Mbps optical network. Picture of the test system arrangement on a whole can be seen in figure 1.

NAIST's CYLINDRA system consists of 6 CRT video projectors and an 8-CPU SGI Onyx2 with a 2-graphics pipe InfiniteReality2 graphics subsystem. Display is a

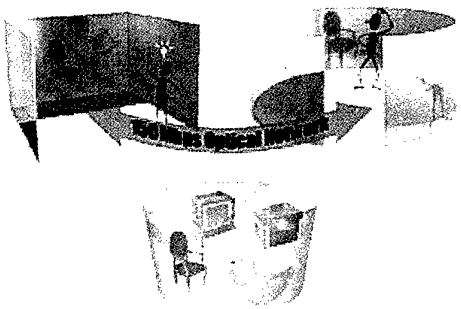


Fig 1. Test system arrangement

330-degree cylindrical wall, 6 meters in diameter and 2.4 meters high; see figure 2. CYLINDRA can produce stereo view for LCD shutter glasses by altering image for left and right eye. A simulated view from cylindrical display setting can be seen in figure 3 and an actual photo in figure 4.

The Immersive Multi-Display System consists of 8 LCD video projectors, 8 400MHz Pentium II PCs and a fast local network. Display consists of back wall, left and right side wall and a partial front floor; see figure 5. Stereo view can be produced for polarized glasses using 2 video projectors with polarized lenses for each screen, one computer handling the output of each projector. In the test arrangement only 3 projectors and 3 computers are used, as stereo view for this platform is not implemented and floor screen is not used. A simulated view from the Immersive Multi-Display System display setting can be seen in figure 6 and an actual photo in figure 7.

the graphical simulation and user interfaces, and uses the network protocol for sharing the virtual space.

The 3D Engine has to be portable, scalable and easily configurable to allow using it in several different systems and display configurations. Portability and scalability set serious restrictions for technologies that can be used, and easy configurability calls for ability to control almost all things through configuration files or at run-time.

At the moment, 3D Engine is programmed in C using OpenGL [4, 5] for graphics routines and GLUT (OpenGL Utility Toolkit) [6, 7] and GLX (OpenGL for X Window System) [8, 9] for window system dependent code used for rendering window and input device handling.

GLUT was originally chosen as the only window handling code to be used in the engine because of its availabity to several operating systems [7]. However, it has some serious restrictions, namely no support for

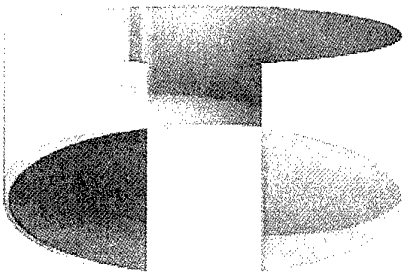


Fig 2. CYLINDRA display setup

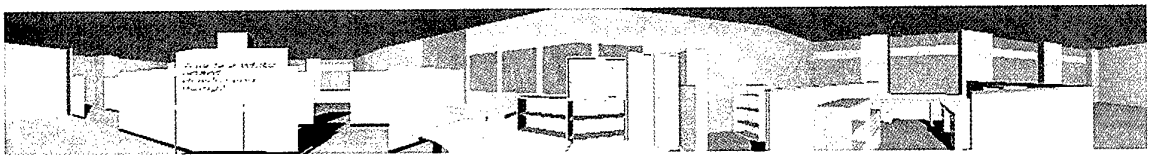


Fig 3. Simulated view from cylindrical display setting

4 3D Engine

Nowadays, there are several types of virtual reality platforms, consisting of different kinds of displays, computers and operating systems. To make an application, especially a shared application, really usable, it must be easily portable to several different platforms instead of having to engineer it separately for each platform. This kind of portability requires a special graphics engine that can be compiled and configured to almost any kind of system without too much work or extra investments for 3D simulation software such as IRIS Performer [2] or Sense8 WorldToolKit [3].

The main application of the system under development is the 3D Engine. It is the part of the system that handles

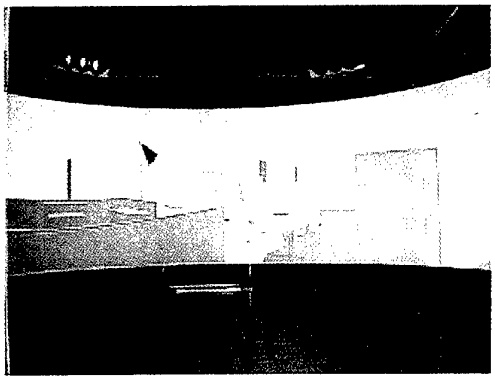


Fig 4. Photo of a scene in CYLINDRA

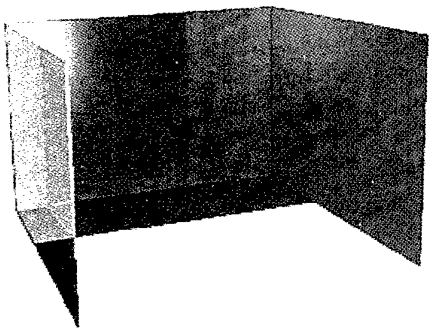


Fig 5. Immersive multi-display system display setup

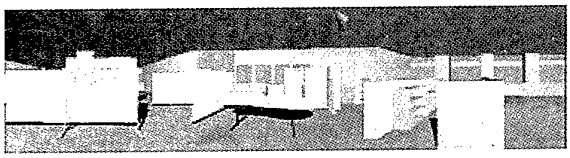


Fig 6. Simulated view from immersive multi-display

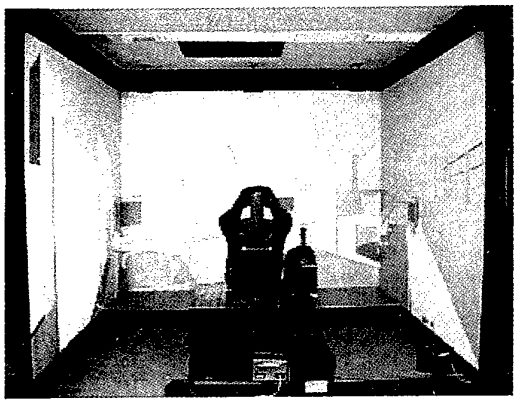


Fig 7. Photo of a scene in immersive multi-display system

multiple CPUs or graphics pipes. This makes GLUT unsuitable for certain systems. CYLINDRA, which is used in the test system arrangement, is one such system. As GLUT would be able to utilize only one of the eight available CPUs, the overall performance is poor. Also, the InfiniteReality2 graphics sybsystem has two graphics pipes, of which GLUT could utilize only one. In CYLINDRA platform this means that only 3 projectors could be used and only half of the display space covered. Due to these restricitions, native window handling code, namely GLX, was added to be used in such X Window Systems GLUT is not suitable for.

Depending on hardware and display configuration, different display handling methods have been implemented into the engine. In a system with a large virtual desktop, which is mapped into separate monitors or screens, one continuous window is created and divided to sections to fit the monitors/screens. Each section in the window can be adjusted as a whole through the configuration files, but the sections cannot be tuned individually. This method is usable with GLUT. Snapshots in figures 3 and 6 have been taken using this method.

In case of separate computers handling the drawing of each screen, the drawing is divided into main clients and help clients. Using this method, the main client handles all the actual work, including handling user input and communicating with the network server to handle scene sharing, and sends screen update commands using either UDP or TCP to the help clients, which only do drawing and nothing more. Whether to use UDP or TCP can be changed through the configuration files. In a normal case UDP provides better performance. Each screen, drawn by a separate client, can be adjusted individually using the configuration files. This method is the most flexible of all implemented methods and is usable with GLUT. Photo in figure 7 is from a setup using this method.

The third method is using GLX and is usable only in X Windows systems as such. It is meant to be used only in systems for which GLUT is unsuitable for, namely systems with more than one graphics pipe and/or multiple CPUs. In this method an X client is created for each separate screen and they can be adjusted as a whole, not individually. Currently the engine supports maximum of 3 graphics pipes and 9 window, but will be upscaled later. Photo in figure 4 is from a setup using this method.

Figure 8 depicts the test arrangement in terms of the display handling method used.

The 3D Engine has been succesfully tested in a number of different configurations, including IRIX 6.5, Linux, Windows 95, 98, 2000 and NT 4 in computers ranging from SGI O2 through several different laptop and desktop PCs to SGI Onyx2. Performance of the engine is, as expected, quite poor in non-3D-accelerated systems and ranging from adequate to very good in a properly 3D-accelerated up-to-date system. Some frame rates in different configurations can be seen in table 1.

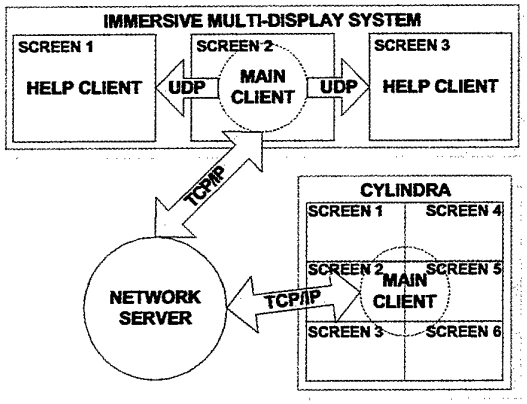


Fig 8. Heterogeneous system arrangement

Table 1. Some performance values

MONITOR - ONE WINDOW	fps : avg (peak)		
640 x 480, no textures, no shadows	**light load (1000 polyg.)**	**medium load (10000 p.)**	**heavy load (50000 p.)**
366MHz, Windows 2000, no 3D acc.	15.8 (20.4)	13.1 (20.4)	8.8 (13.0)
800MHz, Windows 2000, 3D acc.	105.6 (111.1)	86.6 (111.1)	81.7 (111.1)
180MHz MIPS R5000 O2, IRIX 6.5, 3D acc.	18.2 (25.8)	13.9 (25.1)	3.4 (5.1)
640 x 480, textures, shadows	**light load (1000 polyg.)**	**medium load (10000 p.)**	**heavy load (50000 p.)**
366MHz, Windows 2000, no 3D acc.	11.2 (14.5)	8.1 (13.1)	6.8 (10.9)
800MHz, Windows 2000, 3D acc.	80.6 (111.1)	64.5 (111.1)	30.4 (50.8)
180MHz MIPS R5000 O2, IRIX 6.5, 3D acc.	10.2 (12.8)	7.4 (12.8)	2.9 (4.1)
IMMERSIVE DISPLAY - MULTIPLE WINDOWS	**fps : avg (peak)**		
CYLINDRA (6 windows, stereo)	**light load (1000 polyg.)**	**medium load (10000 p.)**	**heavy load (50000 p.)**
1024 x 768, no textures, no shadows	93.2 (166.7)	91.7 (166.7)	11.9 (18.5)
1024 x 768, textures, shadows	36.9 (55.6)	25.4 (47.6)	3.5 (5.8)
Immersive Multi-Display System	**light load (1000 polyg.)**	**medium load (10000 p.)**	**heavy load (50000 p.)**
640 x 480, no textures, no shadows	68.2 (90.9)	51.1 (90.9)	26.9 (45.5)
640 x 480, textures, shadows	54.5 (90.9)	46.7 (90.9)	6.2 (10.8)

For single-window monitor tests used parameters are as follows : view angle 90°, field of vision 73.8° and medium draw distance. As can clearly be seen in the table, the performance is good as long as the complexity of the scene remains tolerable. The system as such, while usable, is not very user-friendly in very complex design situations, such as designing a large office with hundreds of complex desks and chairs.

5 Network Protocol

5.1 Overview

In sharing a virtual space through a network, coherency control, which means keeping consistency of the virtual space between multiple remote locations, is one of the most important subjects. Many different methods for coherency control have been proposed over time [10, 11, 12, 13, 14]. The methods can be categorized in two main types : methods using exclusion control and methods not using exclusion control.

Protocols utilizing exclusion control allow only one user to access the virtual environment at any one time. As locking and unlocking, before accessing the virtual environment and after the operation is finished, require a little time, exclusion control causes delays in system operation. Also, it is not a very user-friendly solution, as only one user can operate each locked object at a time, forcing others to wait. Protocols not using exclusion control result in a tag-of-war -situation when several users are accessing the same object at the same time [10].

In a trial-and-error situation like interior design, restrictions in both aforementioned types of concurrency control pose a problem. To address this problem, a protocol that realizes simultaneous and restriction-free access for multiple users has been developed. The base idea of the proposed protocol is not to prevent a conflict before it happens, but to resolve it afterwards by user's discussion. If separate users' operations for an object causes a conflict, the conflicting object is duplicated to tell the users there is a conflict to be resolved.

For example, a designer and a customer move the same piece of furniture at the same time. This causes the mentioned piece of furniture to be duplicated, indicating there has been a conflict. The designer and the customer can then discuss which choice is better, and either delete one or both, or leave them both as they are. Using this mechanism, user's operations are realized immediately locally, dependless of possible delays in the network.

5.2 Managing Virtual Environment

In the proposed protocol, virtual environment is expressed by a tree of objects. All objects, which will be called nodes in this context, in the tree have specific identifying information. The information in each node includes node identification number for identifying the node in the tree, data of the appearance and placement of the node, such as object name, position and orientation, and the information needed for coherency control, such as version number based on the value of the logical clock and name of the last modifier. See figure 9.

Node identification number is used by the 3D Engine for indicating the object being modified by a user. It is a unique number and is given to the node in creation. Position and orientation data are given as offset values from parent object's position or orientation. Version

number and last modifier of the node are updated when the node is modified by a local or remote user. Updating the version number and the last modifier is done not only to the modified node, but also all it's children and the node on the path from modified node to the root node. See figure 10.

Object name works as a link to the geometry data of the object. In the proposed protocol, all geometry data is stored in an object database and can be accessed using the object name.

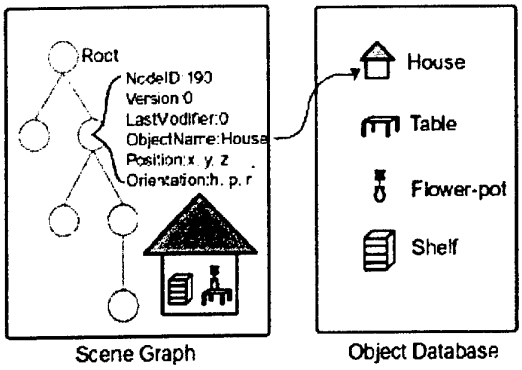


Fig 9. Expressing the virtual environment

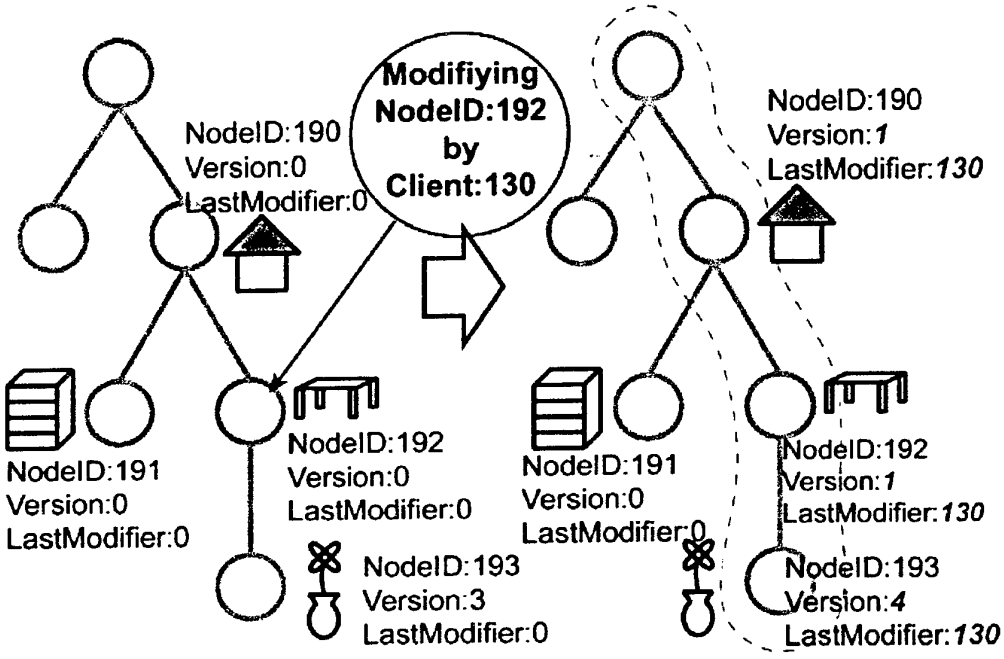


Fig 10. Updating version number and last modifier

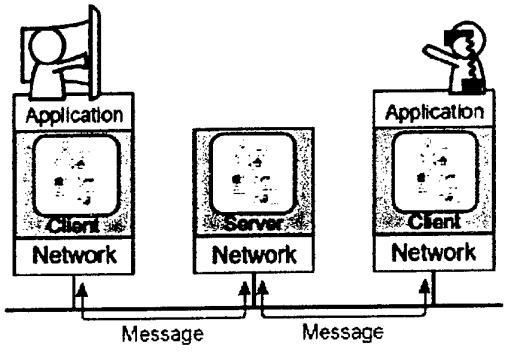


Fig 11. Network model

5.3 Network model

The protocol is based on client-server model, using one or more clients to provide interface for users and one server to manage the virtual environment, including coherency control. All clients and the server have information of the current virtual environment as a tree structure. TCP/IP is used for data transfer between clients and the server. See figure 11.

Clients provide interface through the 3D Engine for creating, moving, rotating, and deleting objects. When user modifies an object, the client updates current tree strcuture and sends the information concerning the modification as a message to the server. The message includes the type of operation, identification number of the modified node, modifying parameters, current version number and current last modifier.

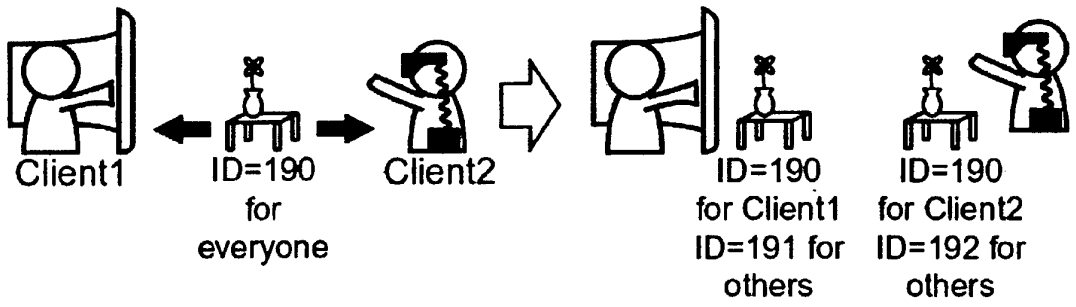


Fig 12. Node identification number translation

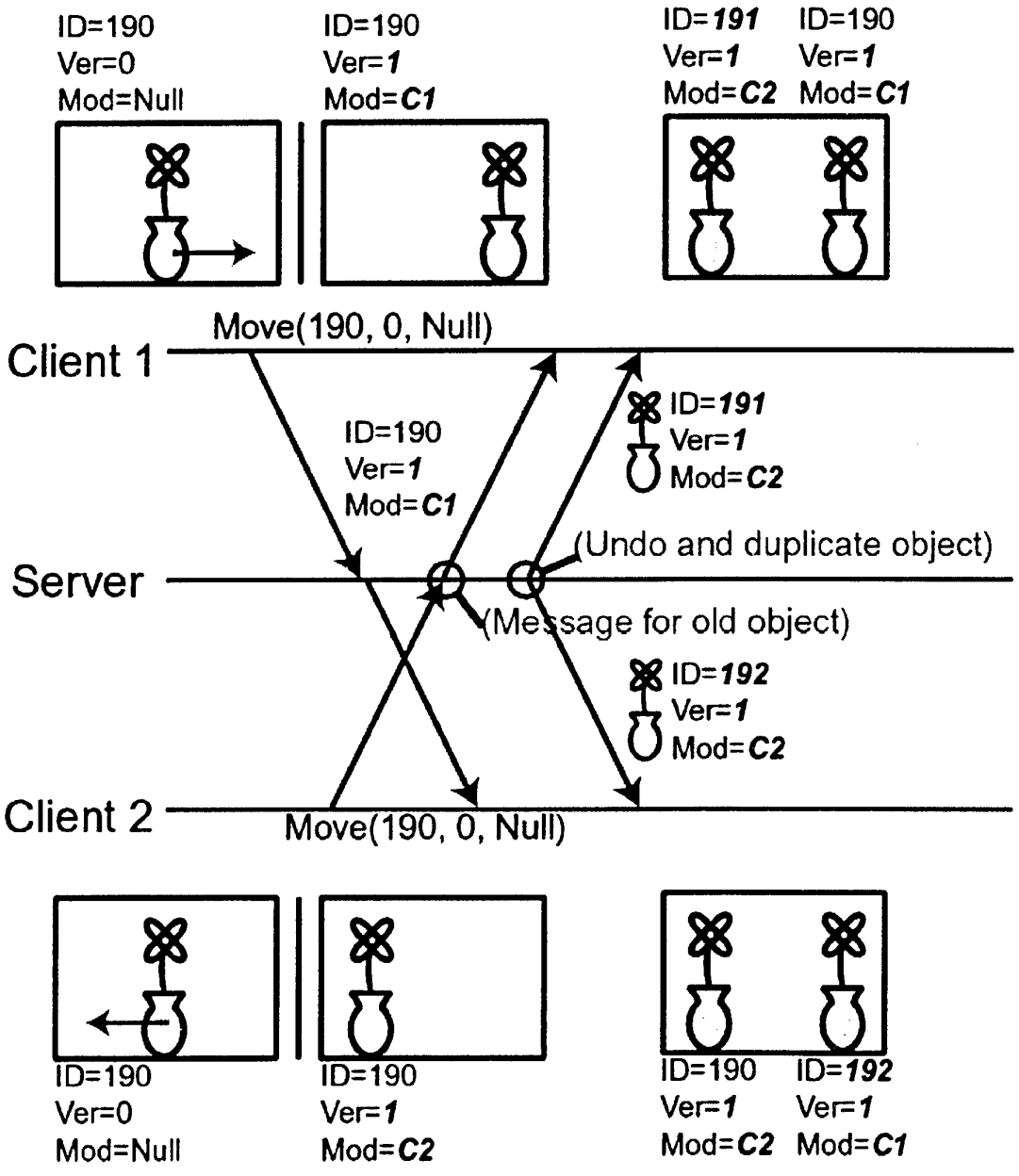


Fig 13. Conflict detection

Server manages connections from clients, data distribution to all clients, and coherency control. When the server receives a message from a client, it does the modifications specified in the message on its own tree structure before sending the message to other clients. In case of node duplicatioon, the node identification number is sometimes different between server and client, as the duplicated node gets a new identification number. See figure 12. For keeping track of different numbers, server has a node identification number translation table for each client, and it is used to translate the identification number whenever a message is sent or received. The translation table is updated when an object is duplicated. When the client receives a message from other clients via the server, it updates its own node.

5.4 Coherency Control

For detecting a conflict, all messages exchanged between a client and the server have a version number and information of last modifier. Information of last modifier is the name of the client who last modified the node. Upon receiving a message the version number and the last modifier in the message are compared with the version number and the last modifier of the corresponding node in the local tree structure. If the values in the message are different from corresponding node's values in the local tree, the node is known to have been modified between the time of creation and the time of having received the message, hence implicating a conflict. See figure 13.

When a client detects a conflict, it ignores the message and destroys it. When the server detects a conflict, it performs node duplication and starts undoing the entire tree until the required node with correct version number and last modifier is found. By this method the existance of the required node is proved. After undoing and locating the required node, the node is duplicated. Nodes to be duplicated are decided in the same way as updating version number and last modifier, as can be seen in figure 10. Duplicating multiple nodes is required for handling nodes having parent-children relation.When a node is duplicated, the server updates the node identification number translation table. Translating node identification number is done so that each client considers the locally modified object primary. After duplicating nodes, the server executes the modification required in a message. Then it redoes the whole tree excluding the duplicated nodes. At this point, the tree has two sets of nodes, one for each user's requirements. Last, the server sends duplicated nodes to the clients as messages. The clients receiving this message add the duplicated nodes into their own tree structures.

5.5 Advantage of the protocol

The proposed protocol provides a new kind of design procedure. The designer and the client can move the furniture around with no limitations from the protocol. After a conflict happens, the designer and the client can discuss which solution is better and modify the scene accordingly. The new procedure can reduce the number of required discussions compared to a case where traditional coherency control method is used. Discussion, in this context, is thought as a sequence of voice communications during design process.

Using a traditional coherency control method only one solution in a conflict situation can be displayed at a time. If either party wants to see the result after both possible modifications, two separate discussions are needed, one after each modification. Using the proposed protocol, both parties can make their own modification at the same time and only one discussion is needed in the end of the modifications.

6 Conclusion

To realize useful interior design system between heterogeneous virtual reality platforms, a portable 3D Engine and a network protocol are presented. The 3D Engine realizes the platform-independency among different platforms, such as a CYLINDRA and a PC-based immersive multi-display system. The network protocol realizes sharing a virtual space without locking while maintaining coherency, making the communication uninterrupted and smooth.

Possible future step in developing the system is changing one end of the system to a portable see-through head-mounted display system. This portable system allows client to stay at his home and see the design using augmented reality, adding virtual furniture into the real space making it even easier to imagine the final result.

References

1. http://www.solidray.co.jp
2. http://www.sgi.com/software/performer/
3. http://www.sense8.com/products/wtk.html
4. http://www.sgi.com/software/opengl/
5. Woo M., Neider J., Davis T., Shreiner D., "OpenGL Programming Guide, 3rd Edition" (1999)
6. http://reality.sgi.com/mjk_asd/glut3/glut3.html
7. Kilgard M., "The OpenGL Utility Toolkit (GLUT) Programming Interface, API Version 3", (1996)
8. http://www.sgi.com/software/opensource/glx/

9. Woo M., Neider J., Davis T., Shreiner D., "OpenGL Programming Guide, 3rd Edition", pp.632-633 (1999)
10. Leigh, J. et al., "CAVERN: A Distributed Architecture for Supporting Scalable Persistence and Interoperability in Collaborative Virtual Environment", Journal of Virtual Reality Research, Development and Applications, pp.217-237 (1997)
11. Singhal, S., Michael, Z., "Networked Virtual Environments", Addison Wesley (1999)
12. Lea, R. et al., "Scaling a shared virtual environment", http://www.csl.sony.co.jp/person/rodger/ICDCS/icdcs2.html (1996)
13. Katayama, A. et al., "Collaborative CyberMirage: A Shared Virtual Environment with High Photoreality and Mutual Awareness", Transaction of IPSJ, pp.1484-1492 (1998)
14. Broll, W., "Distributed Virtual Reality for Everyone", A Framework for Networked VR on the Internet", http://fit.gmd.de/pages/VRAIS.pdf (1997)

ICAT 2000

Image-based Building Shadow Generation Technique for Virtual Outdoor Scene

Xiaohui Zhang *Masayuki Nakajima*
Tokyo Institute of Technology
2-12-1 Ookayama Meguro-ku Tokyo Japan
{zhang, nakajima}@img.cs.titech.ac.jp

Abstract

Image-based rendering techniques are used by many virtual reality applications, especially in the outdoor scene generation applications. For image-based methods, how to deal with the problem of changing lighting situation, especially daylight is a big problem. Lighting affects image in two respects, shadow and color. Image-based shadow generation problem is one of the very important subjects in image-based methods. In this paper, we concentrate on the building shadow, and propose an approach to solve this image-based building shadow generation problem. The key point of this approach is to abstract a simple geometry model of a building by object matching method, and using this simple model to generate shadow under any novel lighting. In the object matching process, Genetic Algorithms (GA) is employed.

Keywords: Shadow generation, Object matching, Genetic algorithms, Virtual outdoor scene

1. Introduction

Generating virtual outdoor scene quickly is one of the big requirements in virtual reality. Because of the complexity of 3D model and daylight in outdoor scenery, image-based methods appear charming. Among the outdoor scene, building objects play a very import role, especially in flight, driving simulation systems, virtual traveling system, building design system. In the scenes with building objects, shadow provides strong clues about the shapes relative positions and surface characteristics of the objects. Besides these shadow can also indicates the approximate location, intensity, shape and size of the light source, and even time information.

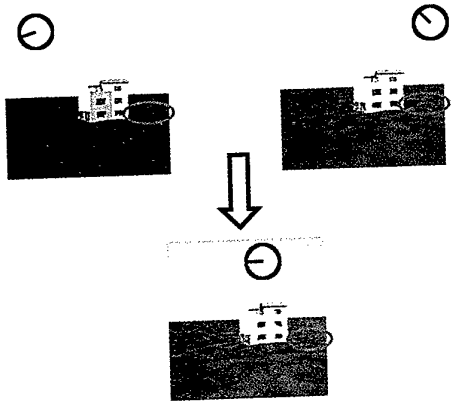


Figure 1 The objective of image-based building shadow generation

Shadow is the interaction of object 3D model and the lighting direction. Therefore, without 3D model information, it is hard to generate shadow. Modeling the building from photographs has been studied by Paul Ernest Debvec in [1], the contribution of this paper is to abstract an accurate model and its surface texture from photographs, and rendering it by CG. In order to get the model, the interaction of users is necessary in this work. Besides this work, there are a lot of researches on image-based modeling[2]. The purpose of image-based modeling is to abstract object model as accurate as possible for CG rendering technique. Most of this kind of works need user's interaction. While the aim of this paper is extract a simple model from several images taken from the same viewpoint but at different time to generate shadow easily and quickly. Therefore different from image-based modeling works, the contribution of this paper is to propose a method to generate shadow under different lighting conditions for image-based rendering and image-based lighting[3]. We call this problem as *Image-based Shadow Generation.*

Though image-based rendering methods have attracted a lot of attention recently, the image-based shadow problem has seldom been studied. This problem can be described as with several building shadow images taken from the same viewpoint but at different times to generate shadow at any arbitrary time, as Figure 1 shows

In our previous work [4], an image-based tree shadow morphing technique is proposed to deal with image-based tree shadow generation problem. The method proposed in [4] employs the abstract geometry model of trees to define the key points of shadows and then uses them as correspondence features. Different from traditional morphing techniques, the key points of new shadows are not determined just by interpolation of the source features and target features, but are calculated on considering the influence of the moving sun. Line segments connected the key points sequentially are used as multiple line pairs and then use the field morphing [5] to establish the transformation. The features of tree shadow and building shadow are quite different. Tree

shadow is consisted of irregular lines, with rich details and having holes in it; while building shadow is consisted of regular lines and generally no holes in it. Therefore, though the tree shadow morphing method solve the tree shadow problem very well, it is not suitable for building shadows.

A *shape from shadow silhouette* (**SFSS**) is used in our previous research[6] to produce building shadows on the ground surface. A 3D model called object shadow shape is reconstructed from several shadow silhouettes by **SFSS** first. The object shadow shape is a 3D model, which can cast the same shadow on ground surface as the real building shadow. Thus shadow can be generated by projecting this shadow shape easily. Shadow on the ground surface is dealt very well by this method, but for the shadow casting on its own building surface, this method does not work because the 3D shadow shape abstracted is not the exact 3D model of building.

In this paper, we propose a new approach which abstract the simple 3D model of building from several reference images by object matching method. Genetic Algorithms (GAs) is used in the 3D model optimization process. With the simple extracted 3D model, both the shadow on the ground surface or other building surface can be generated easily and quickly.

The paper is partitioned into the following sections. The features of a building and its shadow are described in Section 2. The method of abstracting simple 3D model from shadow images by object matching method is proposed in Section 3. The whole process of image-based shadow generation approach is described in Section 4. The conclusion of this paper and future work are given in Section 5.

2. Features of a Building and Its Shadow

For outdoor scene, shadow is generated by the sunlight. Therefore, the sun movement law is necessary to describe briefly.

2.1 Solar Geometry

The earth rotates about the sun approximately once every $365\frac{1}{4}$ days in an almost circular path. The earth also spins about its axis every 24 hours giving diurnal variation in solar intensity. The earth's axis of rotation is tilted by 23.45 degree relative to its plane of motion and this causes seasonal variation in sun position. Therefore, the position of the sun in the sky hemisphere, and, as a result, solar intensity, are determined by date, time and global location.

The location of the sun can be given out with the following equation[7]:

$$\cos\theta_s = \sin d \sin L + \cos d \cos L \cos h$$
$$\sin\phi_s = \frac{\sin d - \sin L \cos\theta}{\cos L \sin\theta} \quad (1)$$

$$\sin d = -\cos[(D_s - 1)\frac{180}{182.6}]\sin(23.45)$$

$$h = (LST - 12) * 15$$

where, as shown in figure 2,

θ_s is the solar zenith,

ϕ_s is the solar azimuth,

LST is the local solar time,

L is the latitude, and,

D_s represents the index of the day in one year. It equals to 1 on December 21, and 365 on December 20.

2.2 Shadows of Building

Building is a man-made object, its outline is generally regular line, and its 3D model can be represented by blocks. Each block has a small set of scalar parameters which serve to define its size and shape. The block is usually geometry primitive, such as cube, cylinder, hemisphere, cone and so on. By these primitives, a simple model of building can be represented as the model shown in Figure 3.

As described above, several geometry primitives could constitute the basic model of a building. Let's see the shadow of these primitives model. Figure 4 illustrates a point located in (X,Y,Z), and its shadow (x_s, y_s) caused by the sun in direction of (θ_s,ϕ_s) on surface $Z=0$. Shadow (x_s, y_s) can be calculated as

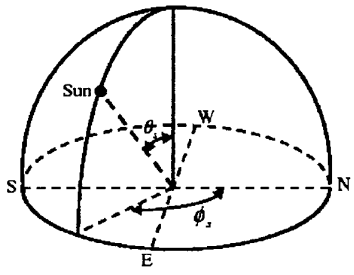


Figure 2 Solar Geometry

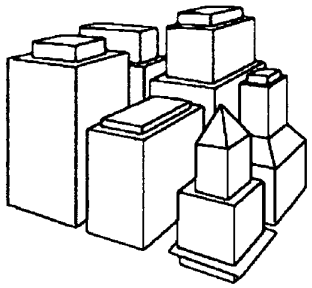


Figure 3 A Building represented by simple 3D

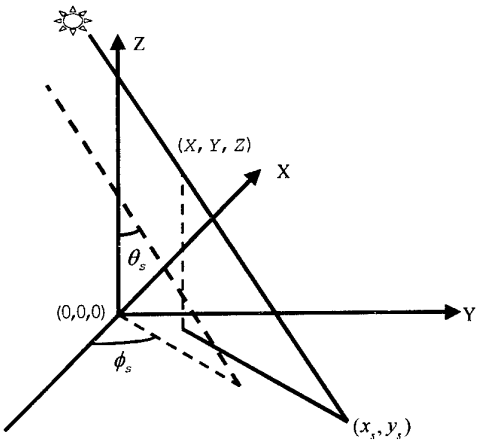


Figure 4 Shadow of a 3D point on surface

$Z = 0$

$$x_s = X + Z\tan\theta_s\cos\phi_s$$
$$y_s = Y - Z\tan\theta_s\sin\phi_s \quad (2)$$

The feature of building decides its shadow has the following features

1. Generally with regular outlines
2. Usually without holes in shadow region

There are mainly three kinds of building shadows. The shadow cast on the ground, the block shade, and the shadow cast on block surface by other blocks. Because of the building shadow features, a simple 3D model of buildings can generate these three kinds of shadows.

The basic idea of this paper is to abstract a simple model consisting of several primitives from several reference shadow images. Having the simple model of a building, its shadow casting on the ground surface or on its own body can be generated quickly. In the following sections, the method of abstracting simple model of building and generating new shadow will be described.

3. Abstraction of Simple Building 3D Model

As analyzing in the last section, a simple 3D model of building can give all of the three kinds of shadows, the simple 3D model should be abstracted first. To abstract the simple model, object matching is employed in this paper.

3.1 Object Matching

Object matching is a technique to recover 3D model from 2D images by projecting 3D model to 2D plane and matching with the 2D image. First a initial 3D model is given, then project it to image plane and compare with the reference image. If they are matched, the 3D model is abstracted. If not matched, adjust the 3D model parameters and do the process again. Different from general object matching method, instead of using images taken from different viewpoint, shadow images taken under various sun positions are used instead in this paper

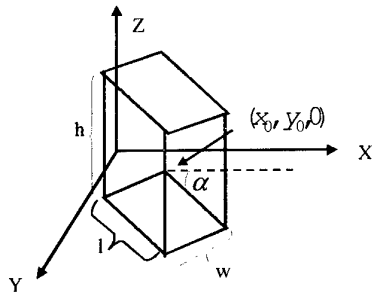


Figure 5 Parameters of a cube

The reference images used in this paper are images taken at different times of a day from the same viewpoint. Camera position, camera parameters, and photographing time are supposed to be known. Thus the shadow images of the initial simple model at each reference time can be produced quickly by CG shadow generation techniques[8]. Compared the generated shadow with the reference shadow images, the error is used to adjust this simple 3D model.

The number and types of primitives constituted the building in the reference images are assigned by user. Though this work is done manually, it would not bring too much burden for users, because for human being it is easy to judge what kind of primitives constituted the building from an image. After appointed how many and what kind of primitives the building consisted, parameters and position of those primitives are extracted by object matching automatically.

The geometry primitives can usually be defined by a few parameters. Take the cube as an example. As Figure 5 shows, a cube on the ground surface can be uniquely specified by only six parameters. These parameters are one corner point coordination (x_0, y_0), length l, width w, height h, and the angle α specified the direction of the cube with the direction of X-axis. These six parameters definitely define the cube 3D model.

Though six parameters can defined a box, a pixel in the image plane is a shadow or not is determined by the six parameters. If there are more than one primitives, it will become much more complexity. Therefore, it could be found that the object matching problem here is a hypersurface optimization problem. Along with the object number increasing, the complexity will increase dramatically. For such a problem, general optimization algorithms are not proper. We employ Genetic Algorithms (GAs) in the object matching process in this paper.

3.2 Genetic Algorithms

GAs[9] are adaptive methods that may be used to solve search and optimization problems. They are based on the genetic processes of biological organisms. They work with a population of "individuals", each representing a possible solution to a given problem. Each individual is assigned a "fitness score" according to how good a solution to the problem it is. For example, the fitness score might be the strength/weight ratio for the problem. The highly fit individuals are given opportunities to "reproduce", by "cross breeding" with other individuals in the population. This produces new individuals as "offspring", which share some features taken from each "parent". The least fit members of the population are less likely to get selected for reproduction, and so "die out". A whole new population of possible solutions is thus produce a new set of individuals. This new generation contains a higher proportion of the characteristics possessed by the good members of the previous generation contains a higher proportion of the characteristics possessed by the good members of the previous generation. In this way, over many generations, good characteristics are spread throughout the population, being mixed and exchanged with other good characteristics as they go. By favoring the mating of the more fit individuals, the most promising areas of the search space are explored. If the GA has been designed well, the population will converge to an optimal solution to the problem. Figure 6 illustrates GAs process.

Take the cube as an example again. The parameters of a cube could be coded as the following string:

x_0	y_0	l	w	h	α

The fitness function of GAs used in this paper is

$$F = \sum_{i=1}^{N} (R_{match}(i) - R_{unmatch}(i))$$

Here R_{match} is the ratio of matched shadow area; $R_{unmatch}$ is the ratio of unmatched shadow; N is the reference image number.

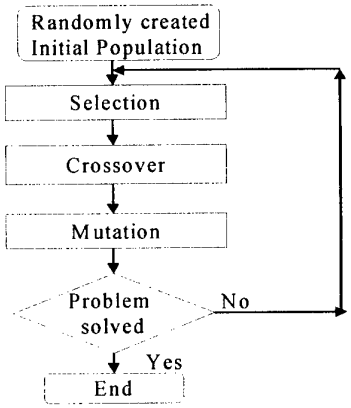


Figure 6 The iteration loop of Basic Genetic Algorithm

4. Image-based Building Shadow Generation

The whole procedure of our image-based building shadow generation is shown in Figure 7. There are three steps, *shadow extraction*, *simple model reconstruction*, and *new shadow generation*.

4.1 Shadow Extraction

Shadows are first abstracted from reference image. In this paper, only building shadow of an outdoor scene is considered. For outdoor scene, the sun is the only light source which can cause shadow, in addition, since daylight can be treated as white light, so the shadow caused by it, could be thought as black color.

Since skylight can be thought as emitting from sky dome that surrounds the earth, we can assume shadows are caused only by direct sunlight. The illuminance of a clear day is greater than that of an overcast day. Consequently, the intensity of pixel in a clear day must be greater than that of its corresponding pixel in an overcast day, except for the shadow region. If the intensity of a pixel in a clear day is less than that of an overcast day image, this pixel must fall into the shadow region. In this way, shadow region can be discriminated simply.

Figure 8(b) shows the extracted shadow silhouettes from the reference images of a box shown in Figure 8(a). The

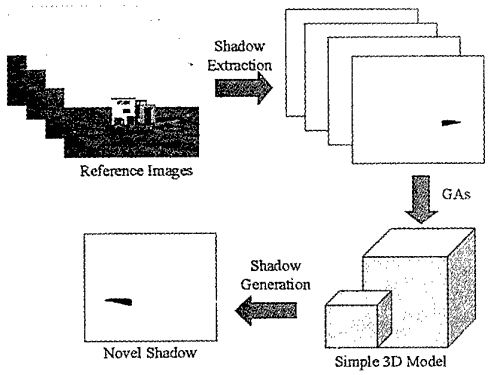


Figure 7 The flow chart of image-based building shadow generation

reference images Figure 8(a) are generated by Radiance[10], and the camera parameters and view position are known.

4.2 Simple Model Reconstruction

The number and type of primitives, which construct the basic geometry model of building, are assigned by user manually first. Then by using object matching method, the simple 3D model of buildings are optimized by GAs.

Table 1 GA parameters

Population Number	500
Crossover Probability	0.8
Mutation Probability	0.1
Generation Number	54

The speed of optimization by GAs is determined by the population number, crossover probability, mutation probability, and the coding length. Table 1 lists the GAs parameters we used to extract the simple 3D model of a box shown in Figure 8(a). The terminal condition is the fitness of the best individual reaches 0.9. The extracted model parameters are shown in table 2, its view on the image plane is shown in Figure 9(a).

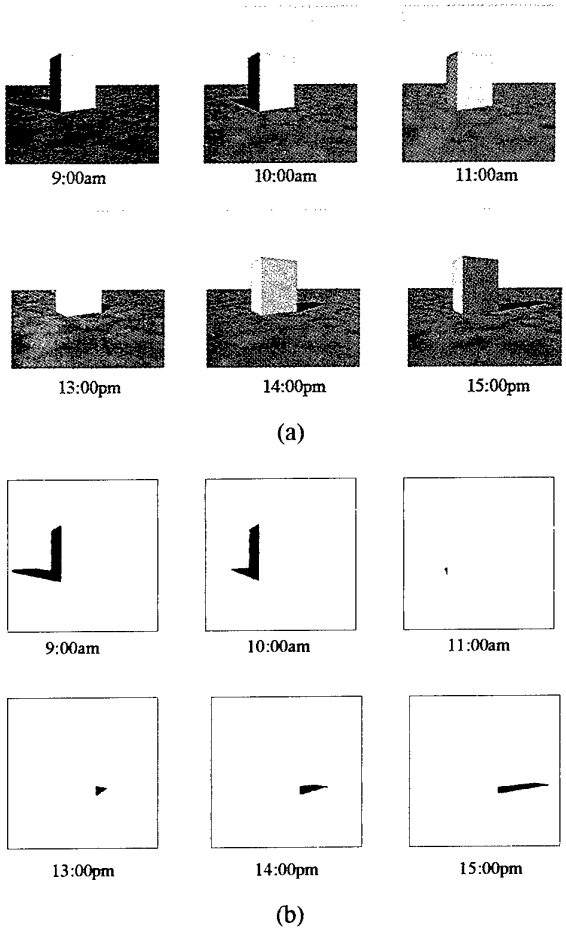


Figure 8 (a) Reference images of a box; (b) Shadow images

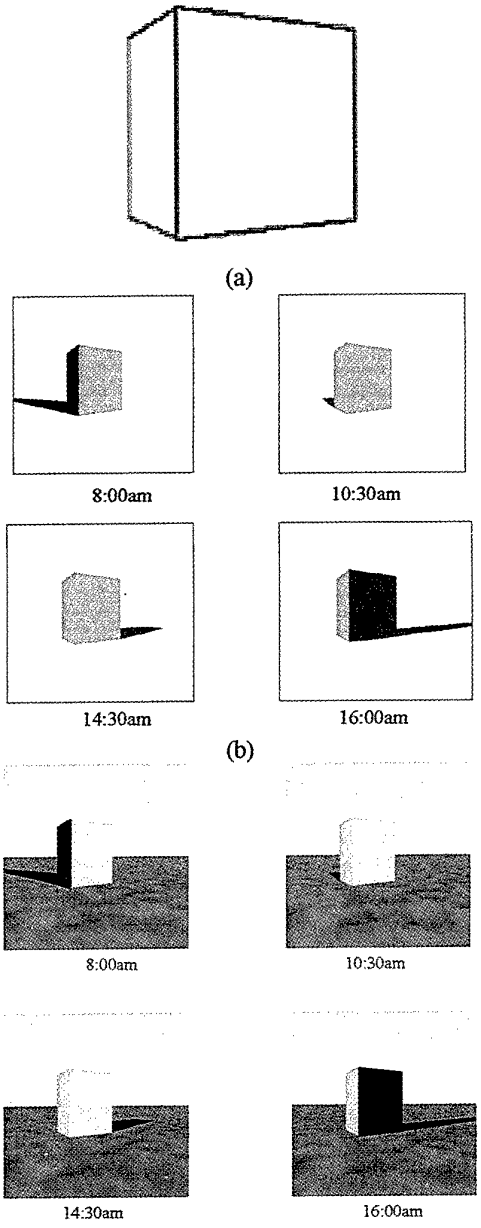


Figure 9 (a)Abstracted box model; (b) Generated shadow by the abstracted box model; (c) New generated shadow combined with other scene

Though the model abstraction process using GAs is a little time consuming, it is a offline process, and would not affect the rendering speed.

4.3 New Shadow Generation

Finally, after the simple model of a building being extracted, new shadow under any sun position can be generated quickly by CG shadow generation method.

From table 2, we can realize that the extracted parameters are a little different from the original one, moreover the model extracted is only a simple model of building, and therefore shadow generated by this simple extracted model maybe a little shift from the real shadow. For the shadow on the ground, a little shift could not affect the visual effect, but for the building self-shadow, even small shift will deteriorate the quality of the synthesized image. To overcome this problem, the following approach is proposed.

In order to let the new building self shadow align with the outline of building surface, the outline of the

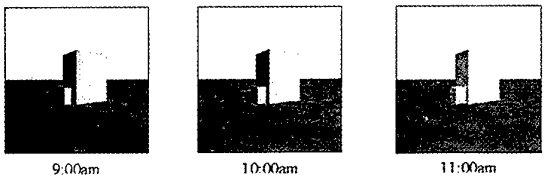


Table 2 Abstracted parameters

	x_0	y_0	l	w	h	α
Abstracted parameters	0.12	0.02	9.97	8.10	13.03	29.7
Real parameters	0	0	10	8	13	30

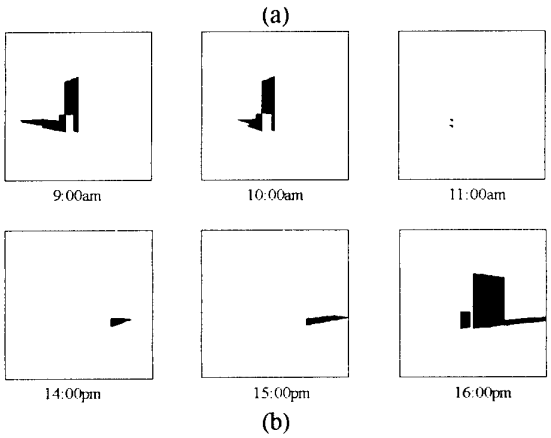


Figure 10 (a) Reference images of two boxes joined by common side; (b) Shadow images

building is extracted by canny edge detect method [11]. Then if there is a building surface line matches the line of new self shadow, take the building surface line to replace the new self shadow line. If there is no such build line matched, keep the new self shadow line unchanged.

Combine the new shadow with the original scene, the image-based building shadow generation problem has been solved. Figure 9(b) shows the new shadow generated by our approach, Figure 9(c) are new images by combining the new generated shadow with other scene. The result illustrates that the shadow on ground and self-shadow can be generated correctly. The shadows at 8:00am and 16:00pm are generated correctly, though they are the shadows beyond the range of reference images. This means that new shadow image at the time beyond the reference images time arrange can also be generated correctly, while this is can not be solved by previous work [6].

Figure 10 is another example of two boxes standing on the ground surface Z=0, and joined by common side. Figure 10(a) shows the reference images, Figure 10(b) shows the extracted shadows. The parameters which define these two connected boxes are described in Figure 11. l_1. Here w_1, h_1, l_2, w_2, h_2 are the length, width and height of box1 and box2 respectively. Here assume the two boxes have the same direction, and specified by angle α. The position of box1 and box2 is specified by a point (x,y), which located on the common line of these

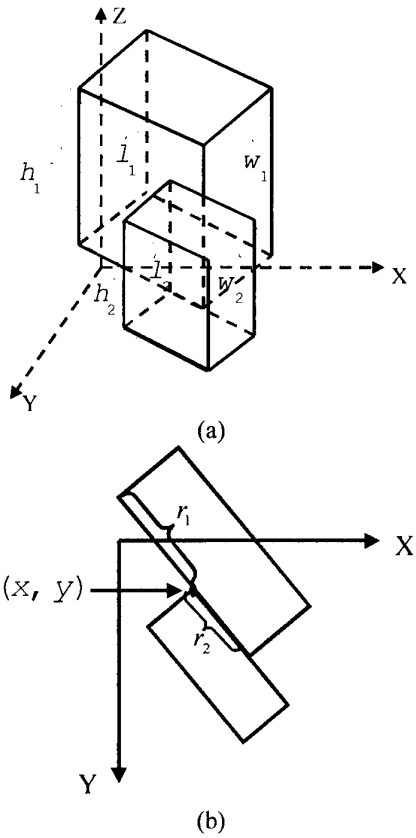


Figure 11 Parameters of two boxes standing on the ground surface and joined by common side. (a) side view; (b) virtical view

two boxes, and the two ratio parameters as shown in Figure 11(b).

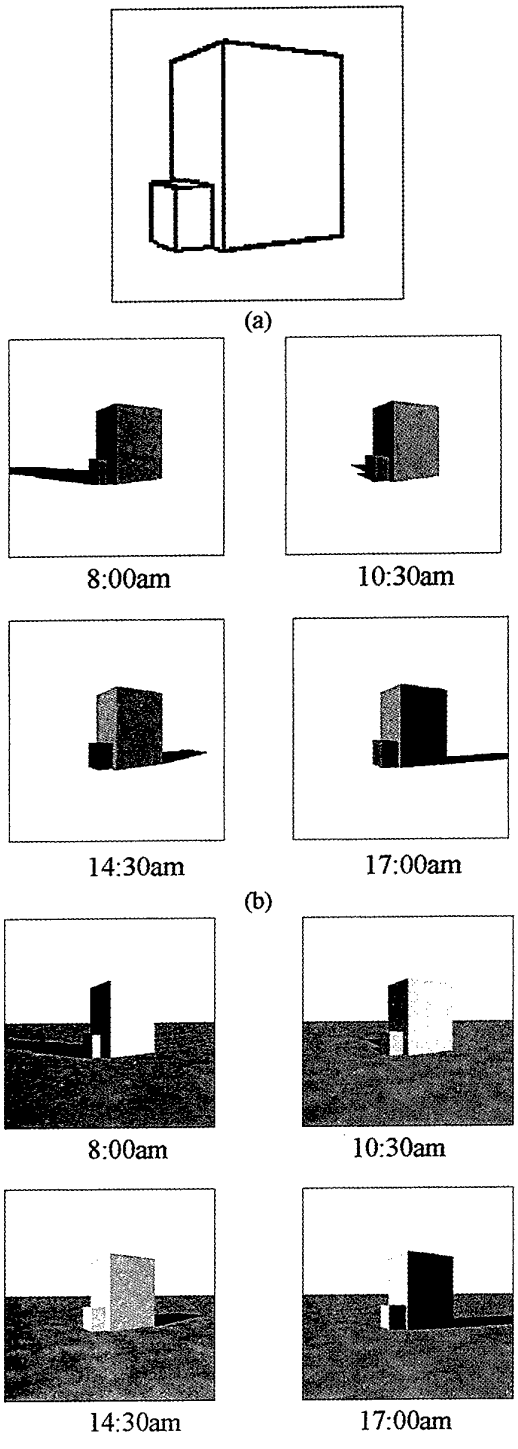


Figure 12 (a) Extracted model: (b) Generated shadow by the abstracted box model: (c) New generated shadow combined with other scene

Figure 12(a) is the simple model extracted by our method. The resulted new shadows are shown in Figure 12(b). The results of combined with other scene are shown in Figure 12(c). The new shadow image of 7:30am and 16:30 in Figure 12(b) shows very clear that the shadow on the small box cast by the bigger box is generated. The kind of shadow also appears on the new shadow image of 16:30pm in Figure 12(b). It demonstrates that the shadow cast by other block is well solved.

For this example, the reference images which taken at time 9:00am, 10:00am, and 16:00pm are very important for extract the smaller box because these shadow images include the shadow information of the smaller box. If there is no information of the small box, the model abstracted by GAs would not be correct. Only using one shadow references image, there will be many specified 3D model, which can generate the shadow. For a box, the minimum shadow reference shadows should include the information, which uniquely determine its height, length, and width. For the relationship between the reference number of shadow image and the correctness of abstracted model are needed further study.

5. Conclusion

The image-based shadow generation problem is a very important problem of image-based outdoor scene generation in virtual reality. This paper as a beginning work of image-based shadow generation, studies the building shadow features. According the features that general buildings are constructed by several simple primitives, we propose an approach to build a very simple building model from its shadow reference images by object matching method. Since the parameter optimization is a hyper surface optimization problem, GAs is employed in the object matching process. Having the simple model of building, its shadow caused by the sun at time can be generated very quickly.

As this paper is only a beginning, there are a lot of problem should be solved. The experiments are only the very simple model, in the future, we will exam this approach to the complex building scene. Besides, the user interface and how to improve the GAs speed will also be considered.

Reference

1 Paul Ernest Debevec, Modeling and Rendering Architecture from Photographs, doctor thesis of University of California at Berkeley. 1996

2 Paul Debevec, What Image Image-based Modeling and Rendering, and What Is Image-based Lighting, SIGGRAPH'99, course39.

3 Paul Debevec, Image-based Lighting, http://www.ee.oulu.fi/~kapu/cg_course/lectures/b-image-based-lighting.pdf

4 Xiaohui ZHANG and Masayuki Nakajima, Image-based Tree Shadow Morphing Technique, The Journal of the Institute of Image Electronics Engineers of Japan, Vol.28, No.5, 2000.

5 T. Beier ans S. Neely, Feature-Based Image Metamorphosis, Computer Graphics(Proc. Siggraph 92), Vol. 26, No. 2, pp. 35-42, 1992

6 Xiaohui ZHANG and Masayuki Nakajima, A study on Image-based Scene Modeling and Rendering Par IV—Building Shadow Generation by Shape from Shadow Silhouette, Proceedings of the 2000 Information and Systems Society Conference of IEICE, D-12-63

7 D.Rapp, "Solar Energy", Prentice-Hall, 1981.

8 Andre Woo, Pierrre Poulin, and Alain Fournier, A Survey of Shadow Algorithms, IEE Computer Graphics and Applications, vol. 10, no. 6, November 1990

9 D.E.Goldberg, Genetic Algorithms in search, optimization and machine learning, Addison-Wesley, 1989.

10 G. Ward, "The RADIANCE Lighting Simulation and Rendering System", SIGGRAPH'94 Proceedings", pp.459-472, 1994

11 Canny, J. (1986). A computational approach to edge detection. PAMI, 8(6), pp.679-698.

ICAT 2000

Precise surface model generation from slice images for Medicine and Archeology

Yasuhiro WATANABE, Kazuaki TANAKA, Norihiro ABE, Hirokazu TAKI, Yoshimasa KINOSHITA, Akira YOKOTA

Kyushu Institute of Technology, 680-4 Kawazu, Iizuka, Fukuoka, Japan

Contact to abe@mse.kyutech.ac.jp

Abstract

The research of a relic excavated from remains has become popular. In late years the research of a relic restoration using CG is also examined. But a laser measurement device is mainly used for measuring shapes of fragments and can't measure uneven complicated shapes. So the X-ray computed tomography to make 3-dimensional measurement possible has begun to be used as a measurement device, but the model generation needs the hand of man still more. In this research, we propose a procedure to automatically recover surface models of fragments with complicated shapes from slice images measured with an X-ray computed tomography. We have already reported a basic restoration system with MRI [1], and models restored with the system are useful to visualization or simulation of relic restoration. Regrettably, the models are not enough precise for experts such as archeologists to make detailed investigation possible. Much more precise models are needed to match the aim of experts.

To get a surface model, corresponding points of contours of two slice images must be found, but this is difficult without manual interposition of man. The surface model of a complicated shape is automatically formed by setting up a surface patch on each grid by interpolating intermediate points between the 2 corresponding contours. Further determination of the joining angle between two fragments became easy because re-construction of the thickness of each fragment is easily attained that is quite difficult to get using a laser measurement.

1. Introduction

A relic excavated from remains appears as a collection of smaller fragments. For the research of the culture or technique of the age when the relic was produced or the exhibition of the original shape

Re-constructing task is necessary to have these fragments joined together. Such a restoration task is taken place using excavated fragments directly up to now. But this restoration task is very complicated generally, and there are many cases that the restoration succeeds as a result of thinking error. Further there is the problem that fragments can't be returned to the original states after the restoration because they are adhered together with glue. Consequently, a re-constructed relic will fairly receive breakdowns compared with the original one. Further we can't examine an individual fragment in excavation after the restoration task. On the other hand, the development of 3-dimensional measurement technique makes it possible to measure correct 3-dimensional shapes of fragments. Further, the development of computers makes it possible to display data of high capacity. So we can measure the shape of each fragment in excavation, and practice restoration without using genuine fragments because a computer successfully reproduce fragments using computer graphics.

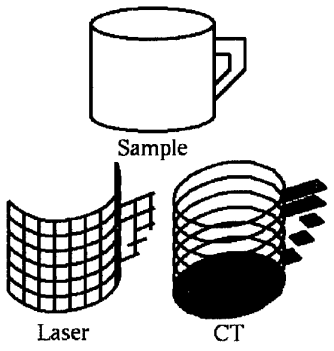


Figure 1. Difference in measurement methods

So far a laser measurement device has been principally used for measuring each fragment, but it is difficult to get the backside and thickness of a fragment although the device can get the close shape and color information of each face. So an X-ray computed tomography scanner is began to use for measuring the internal shape of an object by acquiring a slice image (profile image) as shown in the Figure 1. Further because it has the transitivity, research on a relic or remains will have the broad possibility. Besides, for the restoration of a sophisticated model with a computer, a measurement with an X-ray computed tomography scanner is indispensable. Though a measurement with an X-ray computed tomography scanner can get a close internal shape, it becomes a problem that a connection between slice images becomes discontinuous. The image measured with the computed tomography is modeled with voxels, but the data volume becomes so big that a strong machine power is necessary. So the surface model

making the data volume comparatively small becomes necessary.

A surface model consists of a set of surfaces or boundary surfaces. Any surfaces of a 3-dimensional object completely separate the outside from the inside of it, and must intersect with neither it nor any other surfaces. Besides, it is a very complicated problem to decide the surface including an arbitrary 3-dimensional object from voxel data of the object with a computer instead of the data of surfaces.

Because various interpretation in determining a surface is possible, many different surface construction algorithms are proposed, but needs to intervene with a man' hand for complicated shapes. So the aim of this research is to generate automatically a complete surface model from slice images of very complicated shape measured with an X-ray computed tomography

2. Model generation from slice images

2.1 As a traditional procedure

A voxel model is an aggregation of cells obtained by dividing 3-dimensional space into small unit cells. We can make a model easily by applying the unit cell to fill the interval between slices. Because voxel model just uses obtained CT values, a sophisticated model can be got. Further without forming any surfaces, a model can be provided whatever the shape is complicated. But on the other hand, a data volume increases so much that it becomes difficult to restore or display more than one fragment at a time. A method is called marching cube method that replaces with sooth surfaces the unevenness that is occurred with a set of unit cells when a surface model is generated from a voxel one. The method forms a triangular polygon based on the pattern of picture elements that are within eight neighborhoods of an element on the contour of an image. A surface model of the high quality can be generated with the method. There is, however, the danger that a different shape may be formed if several polygons are erroneously set up. If a shape includes intense changes between two levels of slice, wrong faces are patched there. As a result, the resultant shape is wrong because portions to be originally connected one another are torn to pieces.

2.2 Procedure of this research

The sophisticated model closely resembling the real object is got by using a voxel model, but the data volume increases and visualization or restoration of more than one fragment becomes difficult. A help of man becomes necessary for complicated and non-continuous shapes that can't be handled by the above-mentioned procedure. A purpose of this research is to propose a method that makes it possible to cope with such complicated shapes. It is that salient merit of this research is to introduce intermediary points that make it unnecessary to find the correspondence between two levels of slice image. We show the procedure in the followings.

3 Preprocessing

An X-ray CT image is processed before setting up faces. The image that is provided with an X-ray computed tomography scanner for each slice image is expressed with gray shaded picture elements of monochrome, each of which a value is calculated from attenuation at transmitting an object. Figure 2 is slice image taken with X-ray computed tomography scanner. This image sequence is slice images of 1-mm interval but is actually measured in 0.2-mm interval.

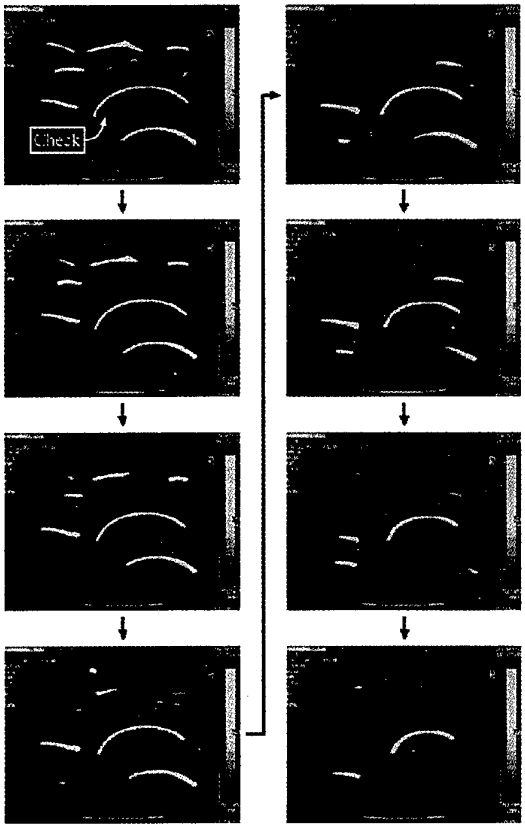


Figure 2. A sequence of cross sections of fragments measured with CT

Binary

Because gray shaded images cannot be expressed in polygons, they must be binaries. It is called threshold process. A threshold value is set at an intense place of alteration. Figure 3 is a binary image of the fragment checked in the Figure 2.

Interpolation of an image

Complicated images may include a thin portion consisting of a single picture element. Filling the portion with faces will result in a face without thickness. That is, it will cause a problem because no hollow surface model

is permitted. So, as a very easy but effective method an image of 3 times is generated. However, because we enlarge an image in length and breadth, the area becomes 9 times in substance. In addition, interpolation is performed. For every image, a hollow surface model can be generated for the portion consisting of a single picture element as shown in the Figure 4.

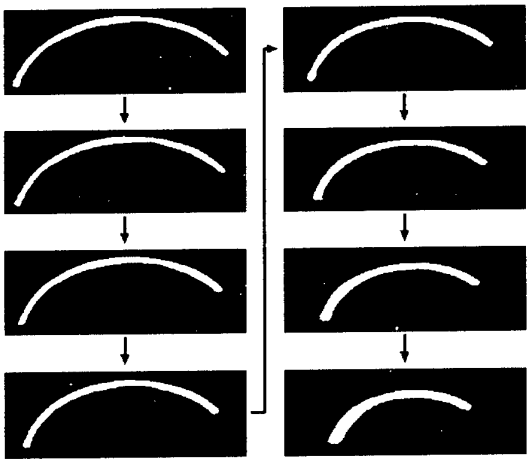


Figure 3. A sequence of cross sections of the fragment checked in the Figure 2

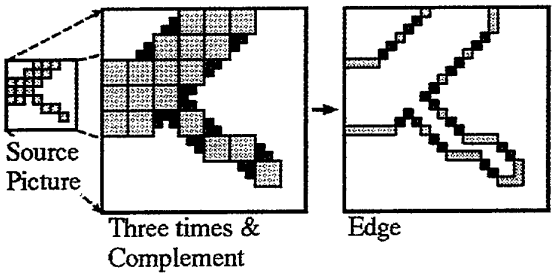


Figure 4. Enlarging and interpolating an image

Contour (an edge) extraction

Setting up a face needs to extract a contour. This is realized by using a brief patch.

3.1 Surface normal (a direction of a contour)

Each surface has a surface normal according to the contour enclosing the surface. So before extracting a contour, the direction of a face can be got by stepping on steps as shown in the Figure 5. This divides areas enclosed by consecutive two contours starting from the external frame of the image. In other words the first contour has an outward direction, and the next one in the opposite direction.

Contour tracing

Though details will be mentioned in the following chapter, a face is set up for every grid unit in the method, a list structure of contours becomes necessary. The list structure can be got by tracing each contour referring to the direction of the face related to the contour.

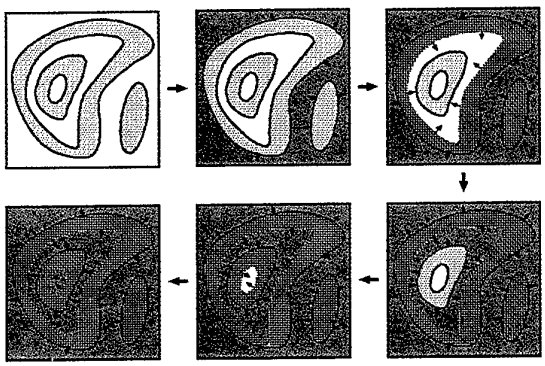


Figure 5. How to find the direction of each contour

3.2 Approximation of a contour using a set of grid points

A salient characteristic of this research is a face tension with a grid unit. The finer a grid unit becomes, the more precise the approximation is. Points that a contour and the grid cross are selected to approximate the contour as shown in the Figure 6.

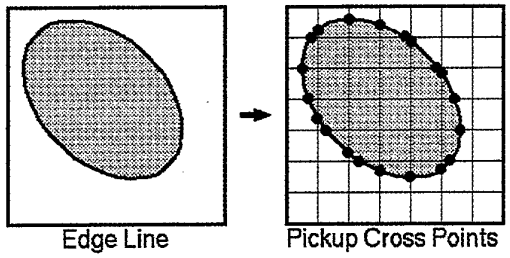


Figure 6. Extraction of a contour by using grid points

3.3 An intermediate point

A salient characteristic of this research is an intermediate point. An intermediate point is a point on the image obtained by taking difference of one slice image and another one. A detailed procedure is described using the Figure 7 as an example.

AND information

An AND collection is an intersection of two pieces of slice image. This portion is the region which polygon isn't set. Using this information, wrong selection of points nearby is avoidable even if the gap between two slice images with intense changes is interpolated.

Intermediate information

The difference information obtained by subtracting the AND information from the OR one mediates between a contour of lower slice from that of an upper one. In other

words we don't need to look for corresponding points between adjacent contours. Intermediary points is obtained by taking grid points included this difference information. The intermediate points are completely separated from the list structure mentioned above. And, the height of the intermediate points is not fixed. It is any decided by a distance of upper and lower slices.

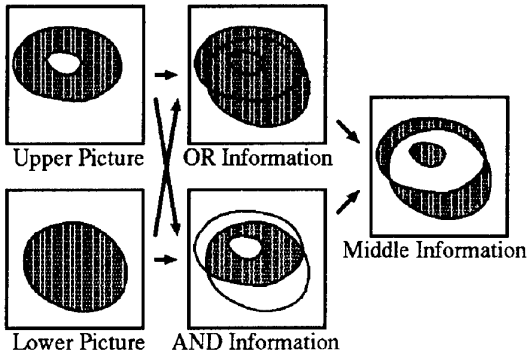


Figure 7. Procedure for generating intermediate points

4 Surface model generation

A surface model of an object is generated from the binary slice image as shown in the Figure 8. Connecting intermediate points and two levels of contour data provided with procedure shown previously, a set of surfaces connecting two levels of contour is generated. Repeating this process over the consecutive pair of contours, a surface model is completed.

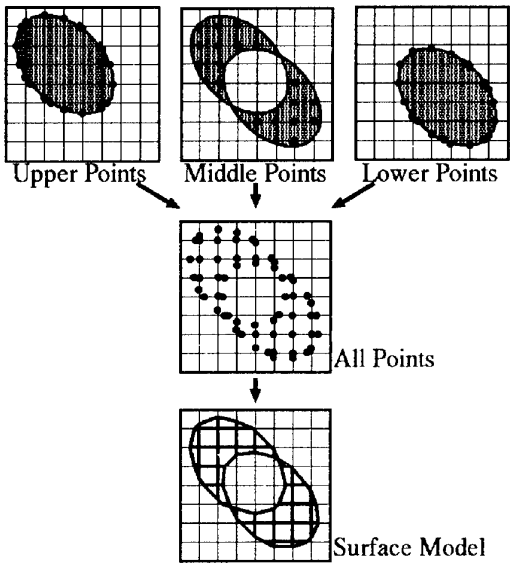


Figure 8. Flow graph of model generation

4.1 Process of grid unit

Because making correspondence between contours is difficult, intermediate points are exploited in this research. Without the search of corresponding points between contour, faces filling a gap between adjacent slice images are successfully set as shown in the Figure 9 using intermediate points. The Figure 10 is finally obtained. Now, the intermediate points express the fixed height. But in fact, the height is decided by a distance of upper and lower slices.

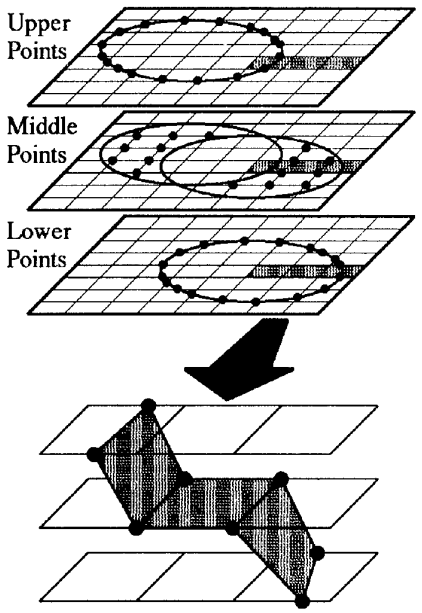


Figure 9. Face extension for each unit grid

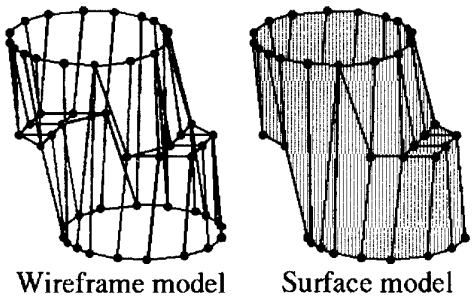


Figure 10. Face extension between two layers of slice image

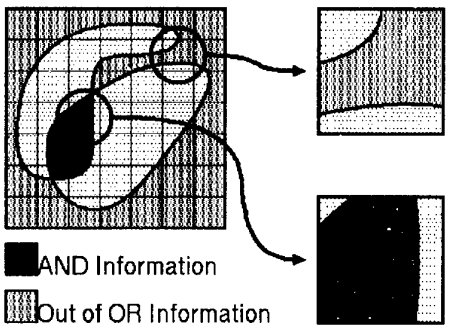


Figure 11. Areas in which face extension is inhibited

Labeling

There is the face that should be distinguished as shown in the Figure 11 when faces are dealt with grid unit. In other words it is a remaining portion obtained by removing both the AND portion and OR portion. We don't set up face on this portion. We have only to perform face tension particularly.

4.2 Face tension algorithm

The face tension is performed with respect to both grid unit and label unit. Tracing picture elements according to the direction of list structure obtained in the previous chapter, surfaces are set up as shown in the Figure 12. Note here that picture elements must be traced according to the opposite direction of the list structure in the next slice image. This allows every surface to be set up smoothly. A twisted portion such as "e, f, g, h" in the Figure 12 can be patched up without any problem. Further for a set of grids "a, b, c, d" in the Figure 12 where only intermediate points exist, the direction of a face can be easily determined from relationship between the top and bottom image.

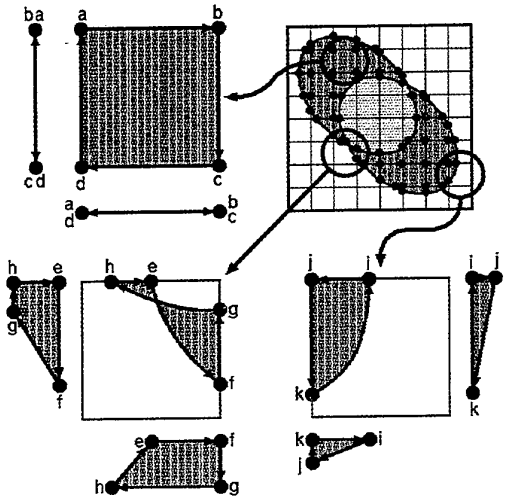


Figure 12. Face extension for a unit grid

5 Generation example of surface model

5.1 The restoring sample models

The restoring sphere

Surface models are generated from the given sphere using this algorithm. The sphere taken from the front and slant is shown in the Figure 13. The wire frame models in Figure 13 are shown in the Figure 14. It is characteristic of this algorithm that the wire frame model is grid.

The restoring holed pot

The model generated the holed pot in the Figure 15 is shown in the Figure 16. We can see that the inside and the hole are clearly expressed. It is impossible to express this with a laser measurement.

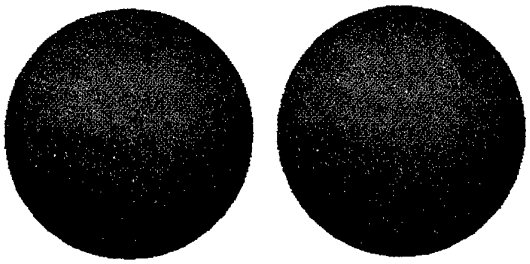


Figure 13. Model of sphere

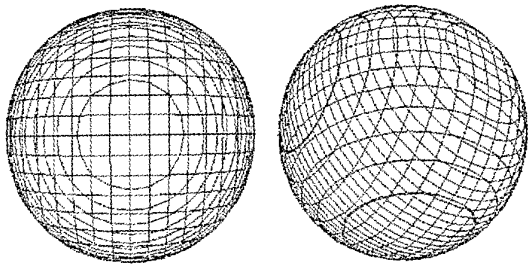


Figure 14. Wire frame model of sphere

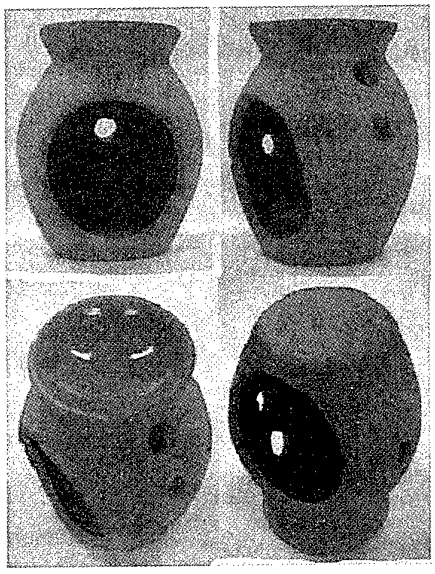


Figure 15. Original model of holed pot

5.2 The restoring relic fragments

Surface models generated from the given relic fragments (see Figure 17 and 18) using this algorithm are shown in Figure 19 and 20. We can see that the thickness of each fragment is clearly expressed, which is difficult to get with a laser measurement. And it is easy to catch characteristic of their shapes. The model magnified the turning point in the flagment in the Figure 19 is shown in the Figure 21. The wire frame models in Figure 21 are shown in the Figure 22.

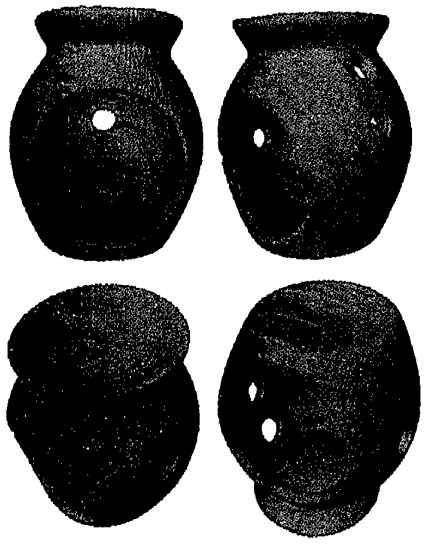


Figure 16. Model of holed pot

5.3 The restoration of relics

The original shape of a relic is restored using fragments restored with this algorithm. In the restoration task, the original tool we developed [1] is used. Figure 23 shows a result of restoration.

Owing to the lack of some fragments, the restored relics include holes. Generating models with CT prove that the one restored with the proposed method is easy to catch the characteristics of the original relic. The restoration task is extremely improved by referring to the thickness of fragments to be joined.

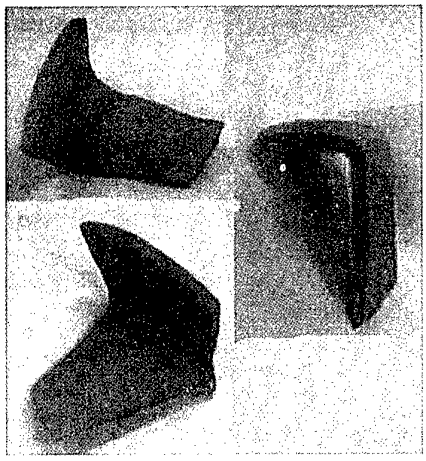


Figure 17. Original model of fragment

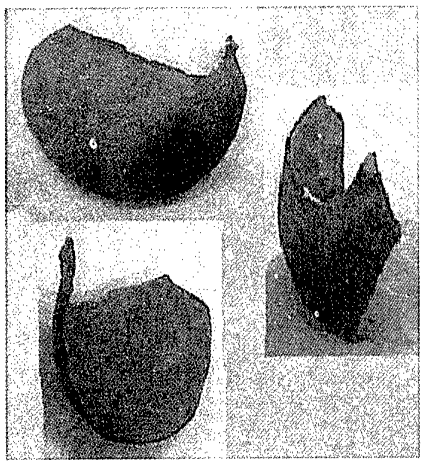


Figure 18. Original model of another fragment

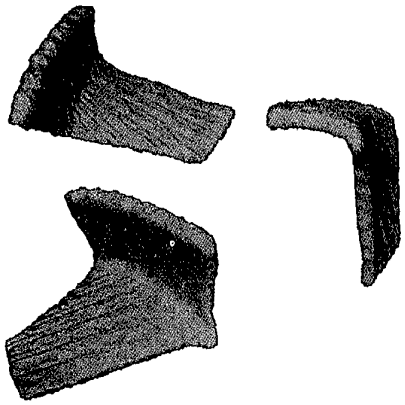


Figure 19. Model of fragment

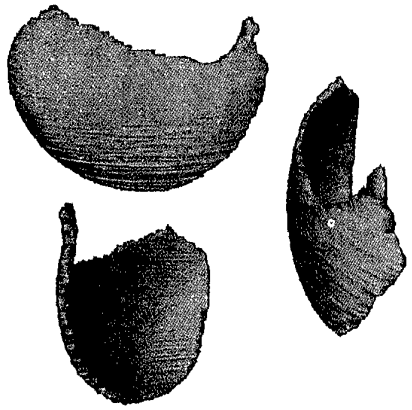


Figure 20. Model of another fragment

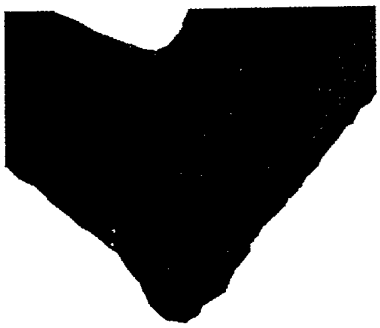


Figure 21. Model magnified the turning point in the Figure 19

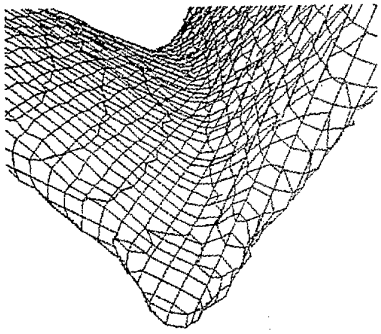


Figure 22. Wire frame model in Figure 21

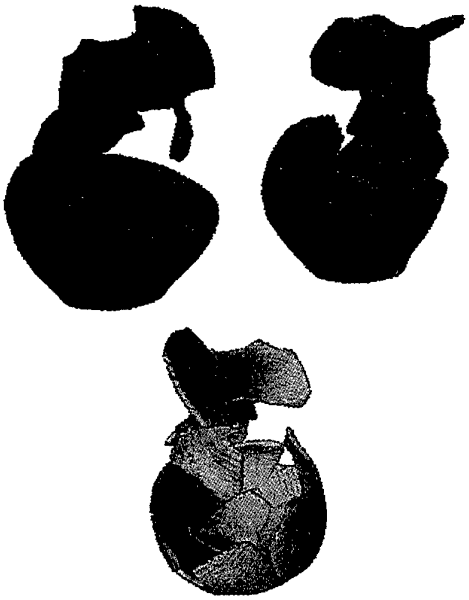


Figure 23. A relic restored with proposed method

6 Conclusion

A new approach is proposed in this paper that automatically restores a surface model of an object with a complicated shape from the CT slice images. Model generation from CT images so far requires not only complicated CAD operation but also Intervention of man. The method proposed makes it possible to automatically restore surface models of objects with complicated shapes. Compared with the thin model restored with a laser measurement, it becomes easy to catch the shape of a fragment by leaps and bounds. Further efficiency of a restoration task is improved by using the thickness of each fragment.

In regard to the future prospect, it is expected that the procedure proposed in this research can be applied to the medical images including very complicated shapes as shown in the Figure 24 as it can cope with a slice image consisting of complicated shapes with intense changes.

Problems to be solved include the improvement in the smoothness of a curved surface and the reduction of data volume. There are often cases where the unevenness is conspicuous because all shading are currently set to the same value. Taking a proper normal vector can be more smooth model. Even for the portion of little inclination the size of a grid is established in the same value. This is the cause that data volume increase idly. This is also solvable if different values are given to grids included in the areas of intense changes.

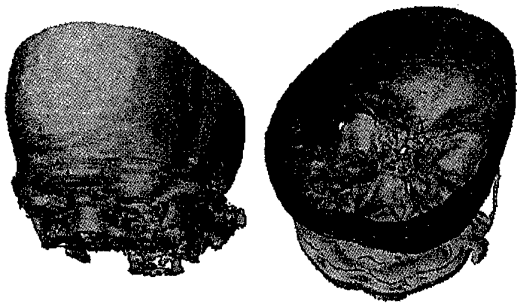


Figure 24. An example of restored skull

Reference

[1] Yasuhiro Watanabe, Kazuaki Tanaka, Norihiro Abe, Hirokazu Taki, Yoshimasa Kinoshita, Akira Yokota: Measurement of Fragments with MRI and Relic Restoration Using Virtual Reality Technologies, the Transactions of the Institute of Electronics, Information and Communications Enginners of Japan (D-II),31, 4, pp. 52-62, 2000.

[2] W.E.Lorensen, H.E.Cline: Marching Cubes: a high resolution 3D surface construction algorithm, Computer Graphics (ACM SIGGRAPH '87 conference proceedings), 21, 4, pp. 163-169, 1987.

[3] Kaneyama K, Chin K, Chihara K: Restoration of fragments using VR technology, Nara Institute of

Advanced Science and Technology Research Report, pp. 9-14, 1996.

[4] Norihiro Abe, Yasuhiro Watanabe, Kazuaki Tanaka, J.Y.Zheng, Shoujie He, Hirokazu Taki, Yoshimasa Kinoshita, Akira Yokota: Relics Restoration by Using Virtual Reality Technologies, International Conference on Computational Intelligence and Multimedia Applications, pp. 673-378, 1997.

[5] Norihiro Abe, Yasuhiro Watanabe, Kazuaki Tanaka, J.Y.Zheng, Shoujie He, Hirokazu Taki, Yoshimasa Kinoshita, Akira Yokota: Virtual Reality Based System for Relics Restoration, International Conference on Virtual Reality and Tele-Existence (ICAT), pp. 122-128, 1997.

Interaction with Medical Volume Data on a Projection Workbench

Ching-yao Lin[1], R. Bowen Loftin[2], Ioannis A. Kakadiaris[1], David T. Chen[1], Simon Su[1]

[1]Department of Computer Science
University of Houston
Houston, Texas 77204

[2]Virginia Modeling Analysis & Simulation Center
Old Dominion University
Suffolk, Virginia 23435

{chingyao, ioannisk, ssu}@cs.uh.edu, bloftin@odu.edu, dave@chen.net

Abstract

Interaction with volume data has often been difficult due to the large memory and processing power required. By taking advantage of current high-end graphics hardware, a volumetric virtual environment has been developed, which allows a user to interact with a volumetric visible human data set. The application enables the user to explore the interior of a virtual human body in a natural and intuitive way.

1. Introduction

In traditional data visualization, researchers visualize data on a two-dimensional screen and use a mouse and a keyboard to interact with the data. Recently, virtual reality (VR) techniques allow users to manipulate data naturally and intuitively in real-time. VR techniques have been applied to many areas of scientific visualization as well as training, one of the major areas is medicine. VR provides an intuitive way to visualize complex medical data, and can be used for medical education, surgery planning and training. In VR applications, the data are displayed in stereo mode, which allows users to better understand the spatial relationships between objects in the environment. In addition, the new generation of hardware allows interactive rates of interaction.

Traditional medical VR applications rendered scenes via surface graphics. Users build the anatomical models through modeling software or extract the surface information from volumetric data such as the ones obtained from magnetic resonance imaging (MRI), or computer tomography (CT) volumes through methods like the Marching Cube algorithm [1]. However, the heterogeneous inner structure of the human body cannot be displayed through surface graphics. Traditionally users studied their MRI or CT data as series of parallel slices, although the data are by their nature volumetric. The interior details of the human body can be presented using volume graphics. In the past, interaction in real-time with volume data was not practical due to the extensive computational power required. With the advent of fast graphics acceleration hardware, we are now able to create an interactive volumetric virtual environment.

Three-dimensional interaction is a more natural and intuitive way to manipulate data. People can "feel" the position and movement of their hands without looking at them. To perform a task, a user's perceptual system needs something to refer to, something to experience. Three-dimensional interaction uses a spatial reference to provide the perceptual experience [2]. Therefore, compared to a traditional keyboard and mouse interface, three-dimensional interaction provides an easier way to locate targets in a three-dimensional environment. For example, if we wish to select a clipping plane at an arbitrary angle in a three-dimensional environment, we can place the clipping plane at the desired location easily by just moving our hand. On the contrary when we use a mouse plus a keyboard, we have to adjust the plane's orientation in a slow and cumbersome manner.

Our goal is to develop an application with the ability to visualize volumetric medical data [2] in a virtual environment at interactive rates, which will allow users to explore the interior of the human body. The application is intended for use in surgical training and planning.

2. Related Work

2.1 Projects based on the Visible Human Data Set

The Visible Human Project™ [3] is a long-range plan of the National Library of Medicine (NLM) to provide data that would serve as a common reference point for the study of human anatomy [3]. NLM has created a complete, anatomically detailed three-dimensional representation of the normal male and female body. The data were obtained using CT, MRI, and digitized photographic images from cryosection. The male was sectioned at 1-millimeter intervals while the female at one-third of a millimeter intervals [4]. There are many applications and products built on the Visible Human data set [5]. Most of those applications are rendering two-dimensional images directly or reconstruct new cross-section images from the original data set [6, 7].

Parker [8] describes a system which uses ray tracing of large volume data (Visible Woman from NLM), multiple CPUs, and shared memory.

2.2 Virtual Reality in Medicine

In this paper, we discuss two major categories of VR applications in medicine. A detailed survey of such applications can be found at [9, 10].

Virtual endoscopy: Virtual endoscopy is used as an alternative to the uncomfortable endoscopy procedure [11]. The three-dimensional CT and MRI data are used to reconstruct the human body to provide a visualization of a patient's specific organs [12, 13 ls For example, in Virtual Colonoscopy [14, 15], the three-dimensional volumetric data are first reconstructed using a set of two-dimensional slice images, and then the surface information is extracted from the volume data by the Marching Cube algorithm. Parallel computation architectures and techniques have been used to improve the rendering rates.

Surgical simulation: The primary VR applications areas for surgical simulation include education, training, diagnosis, preoperative planning, rehearsal, and telemedicine [9, 11]. VR surgical simulators are designed to let young physicians examine the interior of a virtual body to learn anatomy and "practice" surgery [16]. Augmented reality (AR) also plays a very important role in surgical simulation [17].

3. Application Overview

3.1 Motivation

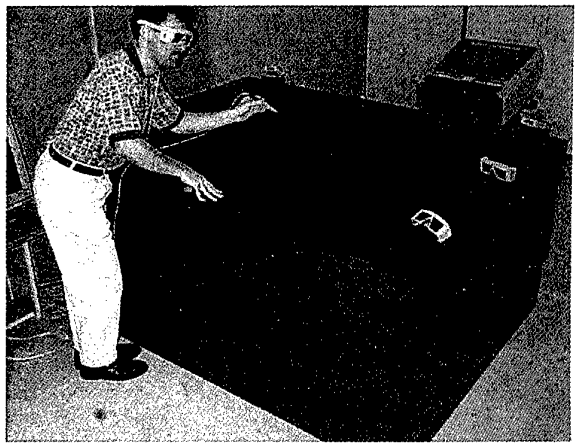
The benefit of volume rendering the Visible Human data directly from the volume data is the increased realism. Surface graphics only visualize surface information. That might lead to substantial loss of information. However, volume rendering provides information about inner structures also [18]. By taking advantage of the three-dimensional texture mapping hardware, one can interact with the volume data in real time.

3.2 System Setup

We built our environment on a custom-made immersive workbench with StereoGraphics®, CrystalEyes® shutter glasses, and a Polhemus FASTRAK® to track the movements of the user's hand (Fig. 1).

We are using the Visible Human male data sets in anatomical modes. These data include 1871 slices of 1760 x 1024 RGB tiff images. They also come with mask files, which contain the segmentation information of these data in different anatomical structures.

The basic idea of three-dimensional textures is to interpret the voxel array as a three-dimensional texture



defined over ([0, 1] x [0, 1] x [0, 1]) and three-dimensional texture mapping as the trilinear interpolation of the volume data set at an arbitrary point within these domains [19]. Two-dimensional texture mapping uses bilinear interpolation and creates the three-dimensional view by adding all two-dimensional slices together. But then, it is necessary to create three different data sets along X, Y, and Z axes to prevent seeing the "gap" between two s lgap'b betwee three-dimensional texture mapping we maintain only one data set.

SGI™ OpenGL Volumizer™ is a graphics API that allows graphics applications to treat volumetric and surface data in a similar way [20]. In addition, it utilizes the three-dimensional texture-mapping hardware to accelerate the performance of applications. One very important feature is the ability to mix volume objects and geometric objects within the same three-dimensional scene. This feature allows us to use traditional surface graphics to create the user interface and still render the data using volume graphics.

There are many different choices to display objects in a virtual environment. Examples include HMDs, workbenches, and CAVEs. On one end, building a CAVE is very expensive, and at the other end, the resolution of HMD displays is very low. A workbench, however, is quite suitable for medical VR applications because it looks like a surgical platform and can display data in real-life size.

The basic idea behind the workbench is to have computer generated stereoscopic images projected onto the surface of the workbench [21, 22]. One user operates in the virtual environment while others can observe that operator's activities through ls activ glasses. The "operator-observers" mode an mode and th to share the same scene provides a good teaching and training environment.

Our working platform is a Silicon Graphics® Onyx2™ workstation with 64 MB texture memory. Images with size 256 x 256 x 256 in RGBA format can fit in the

texture memory and can be displayed fast to achieve the interaction required for the VR application.

3.3 Method

The full size of the original data bank is more than 3 GB. If we tried to display the full data set at one time, the performance would be very slow and we could not achieve acceptable interaction. During the data preprocessing stage, we scale down the resolution of visible human data set to fit in the Onyx2's texture memory. Then we add an alpha channel onto the data.

Seven different data sets were created. The first data set contains full body data in low resolution (220 x 128 x 232). Users can learn the global relationships of biological structures and functions by examining the full body data. The other six data sets are related to: head, left chest, right chest, middle chest, middle body, and lower body. These data sets were created at resolution of 256 x 256 x 128. The user can select different regions of interest via a three-dimensional selection box and s/he can examine particular parts in detail (Figs. 2 and 3).

Using the Visible Human's segmentation masks, we divided the visible human data into eleven groups. (i.e., circulatory, muscular, respiratory, articulations, nervous, digestive, urinary, integumentary, reproductive, endocrine, and skeletal.) The data set also contains complete information on the relative position of different human structures.

We developed a special interface which we call "segmented plane" which combine which combines information with the anatomical data. In general, the segmented plane works like a clipping plane. However, instead of clipping all the parts away like a regular clip plane, the user can select whichever segments s/he would like to remove and examine the remaining parts (Fig. 4). We use the extra alpha channel to perform this task. By adjusting the values of the alpha channel of different segments, we are able to display different segments using different transparencies. Each time the user modifies a segment's alpha value, we re-scan the full set of segmentation information from the mask-file to find the corresponding position of target segments in the volume data. Then, we set the alpha value to the new values and update texture mapping.

The methods of interaction in our application include clipping planes and segment adjustable transparency. Interactive clip planes can remove unwanted parts completely and allow the user to see inside the volume. For example, adjusting the transparency level of different segment objects allows the user to see through skin or muscle.

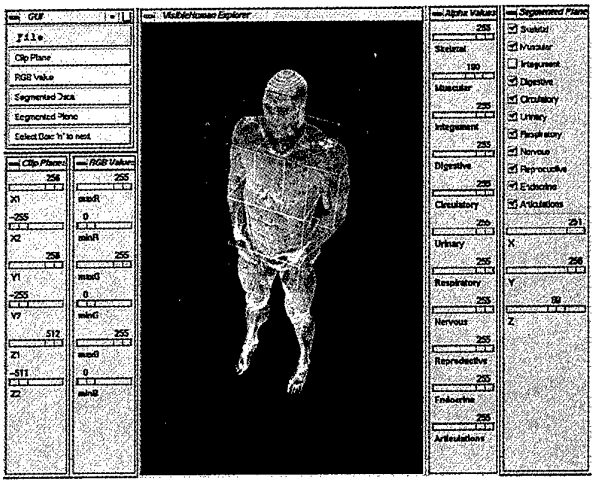


Fig. 2. Display of full body data with the selection box.

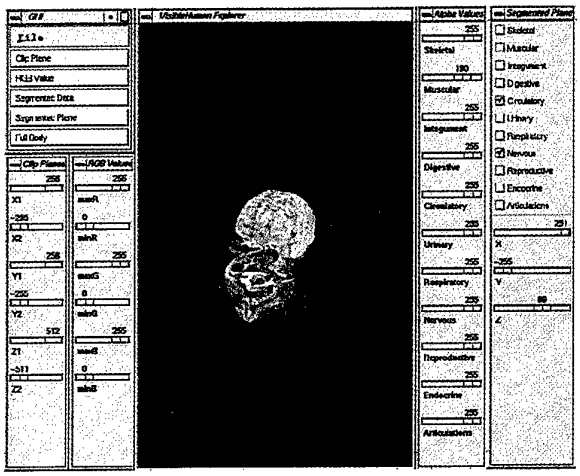


Fig 3. Display of the head in higher resolution

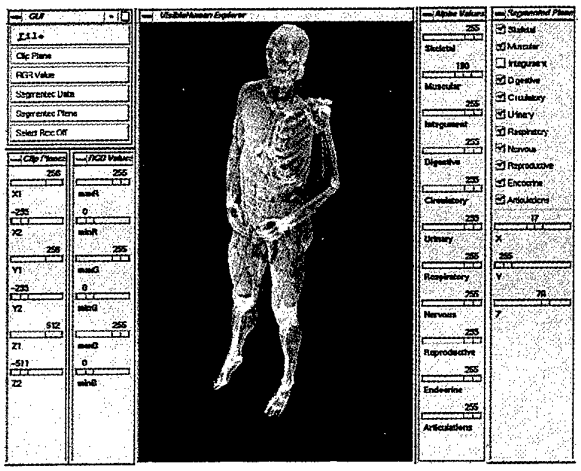


Fig 4. Display of the full body with different segments (skin and musculature of upper left body has been removed).

By rebuilding the virtual body through the Visible Human data set, we created a virtual human to allow students to study anatomy at their own pace for arbitrary lengths of time, since learning anatomy by dissecting cadavers limits a student's exposure to the information

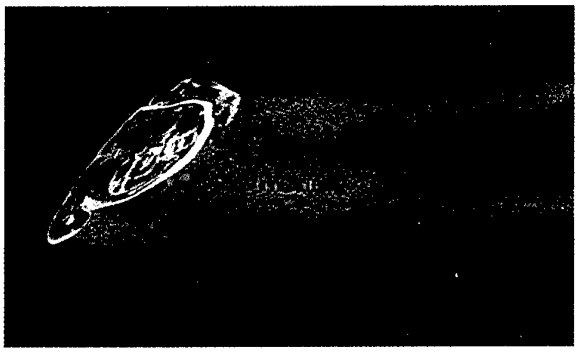


Fig 5. Display of full body on the table with a clipping plane selected

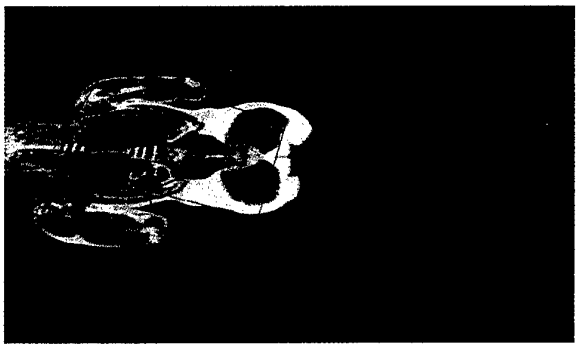


Fig 6. Another example demonstrating that we can select arbitrary clipping plane on the workbench

Fig. 7. Demonstration of our "segmented plane" concept

We have created two modes for our application. One mode displays the result on a monitor with mouse and keyboard input using a Motif GUI (Figs. 2 - 4). The other mode displays the result on the responsive workbench, and uses a Polhemus® Fastrak as tracking device combined with a 3D user interface to allow the user to interact with the data (Figs. 5 - 7).

The frame rate of our visible human explorer is 3.5 frames/sec for a 256x256x256 data volume, and a 800x500 viewport. It appears from statements of visiting doctors that this form would be acceptable to them. Because the data are stored on a remote file server instead of a local disk, it takes longer to load the data sets. Since the main memory of our Onyx2 is 256 MB, we cannot load the entire data set at one time. Our application needs roughly 500 MB of memory (64 MB/data-set * 7 data sets + segmentation information). The main bottleneck is the data loading time. It takes about 15 seconds each time we wish to load a different data set. We are currently experimenting with multi-resolution techniques to reduce loading time. Currently, the time needed to reconstruct data from low resolution to higher resolution is more than the I/O savings. Currently, we load the different data sets directly from disk.

4. Conclusions and Future Work

In comparison to traditional medical data visualization techniques, the main advantages of interaction with medical volume data using VR techniques are the following:

- Improving the understanding of spatial relations between objects in the scene by using stereoscopic mode.
- Providing the user an intuitive, convenient way to interact with volume data.
- Training routines can be performed repeatedly without the cost associated with actual dissections.

In this paper, we have presented a volumetric virtual environment, which provides a novel way to explore the inner structure of the human body. Although we are currently using a high-end workstation to achieve the interaction purpose, as new and cheaper workstations will become available, the interaction performance will be improved [23]. Our goal is to develop a volumetric virtual environment for education on the anatomy of the human body.

5. Acknowledgments

This material is based upon work supported in part by the National Science Foundation under Grant No. NEC95-55682, NASA grant NAG9-985, and funding from the Institute of Somatic Sciences. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and

do not necessarily reflect the views of the funding agencies.

References

1. W. E. Lorensen and H. Cline: "Marching Cubes: A High Resolution 3D Surface Construction Algorithm", *Computer Graphics*, vol.21, no.3, pp. 163-169, July 1987
2. K. Hinckley, R. Pausch, J. C. Goble, and N. F. Kassell: "A survey #As urvey of Desi gn Issues i Input", *ACM Symposium on User Interface Software and Technology*, pp. 213-222, 1994
3. The Visible Human Project®: http://www.nlm.nih.gov/research/visible/
4. M. J. Ackerman, "The Visible Hum 6The Visible *Proceedings of the IEEE*, vol. 86, pp. 504-511, March, 1998
5. NLM: Projects Based on the Visible Human Data Set, http://www.nlm.nih.gov/research/visible/applications.html
6. NPAC-3D Visible Human, http://www.npac.syr.edu/projects/3Dvisiblehuman/VRML/VRML2.0/MEDVIS/
7. S. Senger: "An Imm #An Immersive Environment Direct Visualization and Segmentation of Volumetric Data Sets", *Medicine Meets Virtual Reality*, IOS Press, 1998
8. S. Parker, M. Parker, Y. Livnat, P-P Sloan, C. Hansen: "Interac-Int eract ive Ray Tra c ing fo Visualization" *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 3, 1999
9. J. A. Waterworth: "Virtual Reality i 8Virtual Rea l survey of the State of the Art". http://www.informatik.umu.se/~jwworth/medpage.html
10. A. S. Pednekar, V. Zavaletta, and I. A. Kakadiaris: #Application of Virtual Reality In Surgery Simulation", To appe , To appear in Indian Con Computer Vision, Graphics, and Image Processing (ICVGIP 2000), Banglore, India, Dec. 20-22, 2000
11. Z. Soferman, D. Blythe, and N.W. John: #Advanced Graphic Behind Medical Virtual Reality: Evolution of Algorithms, Hardware, and Software Interface," *Proc. of the IEEE,* vol. 86, no. 3, pp. 531-554, March 1998.
12. R. Robb: "Virtual 8Virtual (Computed) E Development and Evaluation Using the Visible Human Datasets", *Proc. Visible Human Project Conference* ' 96, pp 2-230, 1996
13. R. A. Robb: "Virtual End 8Virtual Endoscopy Using the Visible Human Datasets and Comparison with Real Endoscopy in Patients", *Medicine Meets Virtual Reality*, IOS Press, 1997
14. S. You, L. Hong, M. Wan, K. Junyapraser, A. Kaufman, S. Muraki, Y. Zhou, M. Wax and Z. Liang: "Interactive Volume Rendering for Virtual Colonoscopy", *Proc. IEEE Visualization '97*, pp. 433-436, Oct. 1997.
15. M. Wan, Q. Tang, A. Kaufman, Z. Liang and M. Wax: "Volu 8Volume Rend ering Interactive Navigation within the Human Colon", *Proc. IEEE Visualization '99*, pp. 397-400, October, 1999
16. R. M. Satava: "Virtual Reali 8Virtua l Reali Application", *ITAB '97, 97, Proceedings of the Engineering in Medicine and Biology Society Region 8 Conference*, pp. 19-20. 1997
17. A. State, M. A. Livingston, W. F. Garrett, G. Hirota, M.C. Whitton, E. D. Pisano: 6Technd ogies for Aug mented Reality Syste ms: Realizing Ultrasound-Guided Needle Biopsies", http://www.cs.unc.edu/~us/uswww/good.html
18. A. Kaufman: "Volume Visualization: Principles and Advances", Course n , Course not es 24,
19. R. Westermann, T. Ertl: "Efficiently Using Graphics Hardware in Volume Rendering Applications", *Proceedings of SIGGRAPH 98*, pp. 169-177, July 1998.
20. OpenGL Volumizer, http://www.sgi.com/software/volumizer
21. W. Krueger, B. Froehlich: "The Responsive Workbench", *IEEE Computer Graphics and Applications,* vol 14, pp. 12-15, May 1994
22. G. Eckel, M. Gobel, F. Hasenbrink, W. Heiden, U. Lechner, H. Tramberend, G. Wesche, and J. Wind, $Benches andCaves", *ICON '98. 98. Proceedin the 24th Annual Conference of the IEEE*, vol. 4, pp 1996-1999, Sept. 1998
23. M. Haubner, C. Krapichler, A. Lösch, K. Englmeier, and W. V. Eimeren: "Virtual eality in Medicine-Computer Graphics and Interaction Technique", *IEEE Transactions on Information Technology in Biomedicine*, vol. 1, no. 1, pp. 61-72, March 1997

Control of Force Display Device by the Force Sensing System using EMG.

Hideo KITA, Keitaro NARUSE, Hiroshi YOKOI, Yukinori KAKAZU

Laboratory of Autonomous Systems Engineering,
Resarch Group of Complex Systems Engineering,
Graduate School of Engineering. Hokkaido Univaersity
Nihi-8, Kita-13, Kita-ku, Sapporo, 060-8628, JAPAN
{kita, naruse, yokoi, kakazu}@complex.eng.hokudai.ac.jp

Abstract

In recent years, the research area known as Artificial Reality, Virtual Reality and Tele-existence have been paid much attention from various application fields, including entertainment, medical engineering and computer aided instruction. The goal of these areas is to create virtual spaces that give natural feelings to human users, or operators.

To create good virtual spaces, it is indispensable to deal with and integrate various information from various senses that human being has. We, however, concentrate on dealing with one of those senses. It is the sense of force. The sense of force is necessary to realize touch of objects and feelings of its weight.

Most of researches on force display depend on dynamics models of objects to be operated, and control force feedback devices by using the models. On the other hand, for controlling force the feedback devices, we employ information about physiological conditions of operators rather than the dynamics models of the objects. Our method is based on the idea that a control scheme of the force display can be realized by an operator side as complemental way to the conventional methods. We call the system "Personality adaptable type force feedback device system".

Key words: Virtual Reality, force feedback, EMG

1. Introduction

When a person operates an object, the resultant feeling of the object differs from other persons' feelings, even if they operate a same object. This is because what kind of characteristic of the object is important depends on subjectivity of each person. Ignoring this kind of differences in the processes of making dynamics models of the objects gives operators having incongruity feeling. However this kind of the differences is seldom taken into account.

To deal with this problem, we use not only the dynamics models of the objects, but also use recorded data of operators' physiological conditions collected when they operated those objects in the actual world. In the phase of controlling the force feedback devices, it is a goal of the system to close the operators' physiological conditions to the recorded conditions. This method enables us to cope with the changes of operators' impedance caused by fatigue of muscles and various physical conditions. The control flow is shown in Fig. 1.

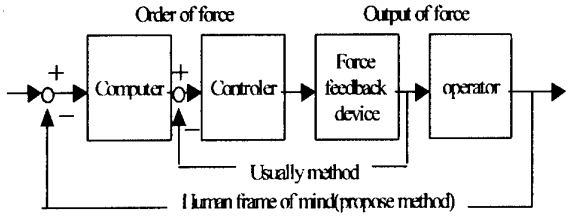


Fig. 1 Control flow

In this paper we use a surface electromyogram(EMG) signal which is one of bio-signals as physiological conditions. The surface EMG signal is active electric potential generated by contraction of muscles. It is measured by electrodes put on an operator's skin. Using the surface EMG signal matches our purpose, because some physiological conditions which concerns with force of the operators are necessary to deal with sense of force.

In order to control the force feedback device, we have to properly decide the magnitude of force the device displays by using the surface EMG signal. Here, we take the following assumption: if two patterns of the surface EMG signal taken from an operator at different points in time are similar to each other, the operator's subjective feelings at those points in time also similar to each other. With this assumption, in order to give operators certai feeling virtually, it becomes a goal of the device to control it so as to close the pattern of operator's the surface EMG signal to those measured in the actual world. Building such devices rise need for attention to the following characteristics of the surface EMG signal:

1) Surface EMG signal relates to force generated by muscles.

2) Surface EMG signal is influenced by fatigue of muscles.

3) Surface EMG signal varies according to operator, and sometimes vary according to time, even if same operator.

And there are following problems:

1) From the viewpoint of signal-noise ratio, the surface EMG signal has undesirable characteristic. In other words, it contains much noise.

2) According to the position of the electrode, measured values largely vary.

To achieve the goal with coping with these characteristics and problems, we employed a feed-forward type neural network for mapping from the surface EMG signal to the magnitude of force to be displayed. Feed-forward type neural network is suit for clustering of data with nonlinearity, and the surface EMG signal have this property. Learning ability of feed-forward type neural network enables us to cope with the problems such as potions of an electrode, fatigue of muscles and individuality. We made prototype of "Personality adaptable type force feedback device system", and investigate operator's feeling about this system.

2. Design

The block diagram of the proposed model is shown in Fig. 2.We enter into details of the Fig. 2.

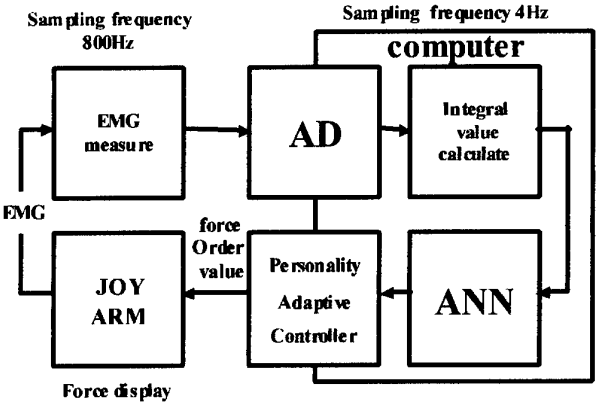


Fig. 2 Block diagram of the propesed model

Measurement of EMG

We deal with the two channels EMG signals. EMG electrode is shown in Fig. 3. The EMG electrode is a rectangle three centimeters by two centimeters. The position of EMG electrode is shown in Fig. 4. The electrode uses the product of DelSys Inc. The side of radius is channel zero and channel one, and the side of ulna is channel two and channel three. Reference is placed on the wrist near the hand. The sampling frequency is 800 Hz.

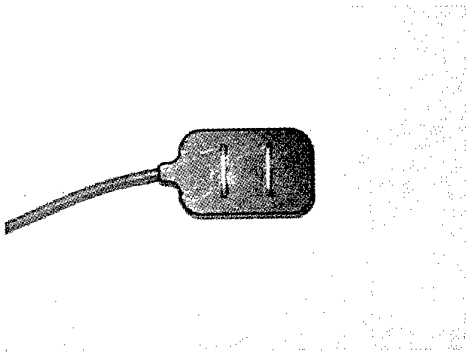


Fig. 3 EMG electrode

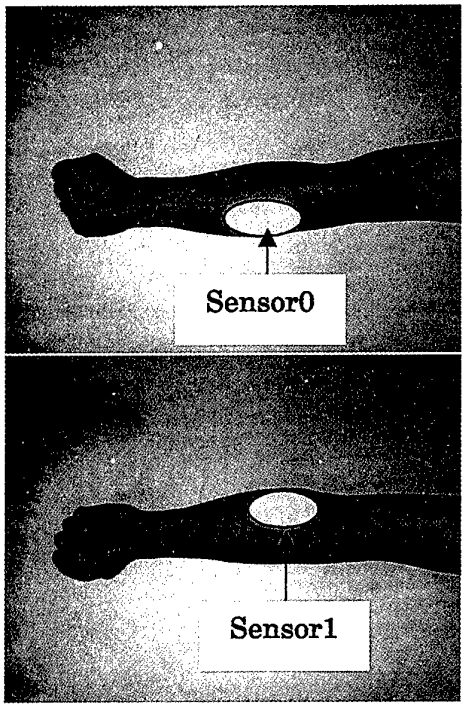


Fig. 4 sensor positions

Pre process

EMG measured by the surface electrode is that the temporal and spatiality addition value of the electric pulse which occurred from the tip of the nerve fiber. In other words, muscular impedance around the electrode can be estimated to integrate EMG in the constant time. Only a difference in electric potential from the ground is important here. Therefore, the absolute value of EMG is calculated, and integral value (1) is found after that.

$$I=\sum_{t}|e(t)|\bullet\Delta t \quad (1)$$

I : Integral value
e(t) : EMG signal at time t
Δ t : 1/(sampling frequency)

Next, it needs to be considered that the dispersion of the integral value is reduced. It is done concretely in the following method. The method prepares for the integral value of eight sections where time continues, and it excludes two the biggest integral values and two of the smallest ones. Then, it averages four left integral values. This value is handled as one normalized integral value.

Artificial Neural Network (ANN)

We employed a feed-forward type neural network for a mapping from the surface EMG signal to the magnitude of force to be displayed. In this network, It is made to record that operators' surface EMG signal collected when they operated those objects in an actual world. And, when a certain feeling is given virtually to the operator, ANN degree of resemblance of recorded EMG and present EMG is outputted. The network uses (2).

$$u_i^m=\sum_{j=1}^{n_{m-1}} w_{ij}^m \bullet x_j^{m-1} \quad (2)$$
$$x_i^m=f\left(u_i^m\right)$$
$$f(u)=1/(1+\exp(-u))$$
$$(m=2,3,i=1,\cdots,n_m)$$

n_m : The number of the cells of the m layer

x_i^m The cellular output of the i turn

w_{ij}^m The cellular weight of the j turn of one previous layer

$f(u)$ The function of sigmoid

$$\frac{\partial E}{\partial x_i^3}=x_i^3-d_i$$
$$\frac{\partial E}{\partial x_i^2}=\sum_{k=1}^{n_3}\frac{\partial E}{\partial x_k^3}f\left(u_k^3\right)w_{ki}^3$$
$$\Delta w_{ij}^m=-\mu\frac{\partial E}{\partial x_i^m}f\left(u_i^m\right)x_j^{m-1}$$

d_i The training data of i turn

μ Learning rate

It was made three layers structure of ANN, and the following is the number of neuron of each layer. The input layer is two neurons which is the number of the sensors. The middle layer is ten neurons. The output layer is three neurons which is the number of the loads

Personality Adaptable controller

In the phase of controlling force feedback devices, the goal of a system is to close the operators' surface EMG signal to the recorded surface EMG signal. And in order to control the force feedback device, we have to properly decide the magnitude of force which the device displays

by using the surface EMG signal. Personality Adaptable controller calculates that force. As for the details, it is stated in section 3.3.

JOYARM

JOYARM manufactured by Mitsui Engineering & Shipbuiding Co.Ltd display force. Force of six shaft can be displayed JOYARM by DC motor, and it can monitor a three-dimensional position. JOYARM display the load0 (0kg), load1 (83.68kgmm) and load2 (167.29kgmm) as a torque load toward the rotation movement of the arm.

Fig. 5 JOYARM

3. Experiments

3.1 Experiment 1

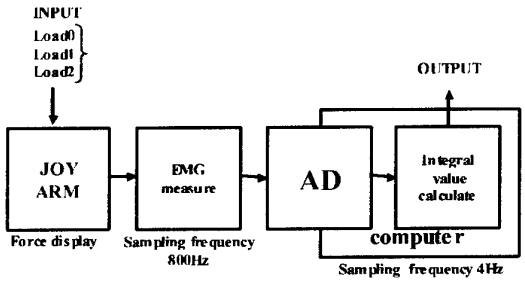


Fig. 6 Control flow of Experiment of measure EMG

The procedure to measure the surface EMG signal is explained in this section. And it is examined what kind of characteristics there are in surface EMG signal. Then, concrete experiment process is shown.

The detail condition was shown in section 2. The posture of measurement is shown in Fig. 7. We show the sequence of the loads which JOYARM displays. First, displayed load0. Secondly, displayed load1. Thirdly, displayed load2. Interval of the experiments is long enough, because we want to refresh operator. We do not determine special interval time. Each operators determined interval time. The time of measurement is 1.5-second par one experiment. We get six integral values from one experiment. Because there are three loads, we get 18 integral values. The experiment went toward one operator. The operator is a man 24 years.

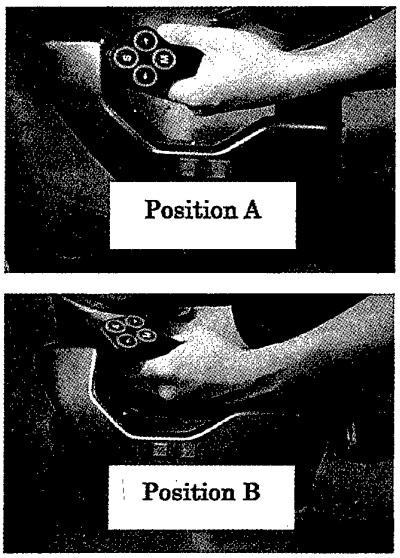


Fig. 7 Measured Positions

3.2 Result of experiment 1

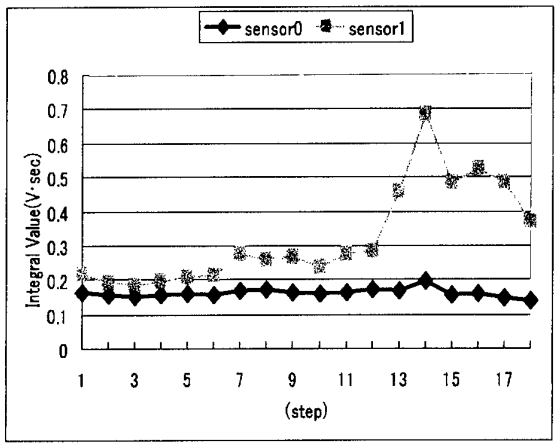


Fig. 8 result of experiment 1

The experiment result is shown in Fig. 8. The horizontal axis of ordinates is step, and the vertical axis is the integral value (V·sec).The diamond mark represents the integral value of sensor0 and foursquare mark represents the integral value of sensor1. JOYARM displays the load0 from first to sixth steps, displays the load1 form seventh to twelfth, and displays the load2 thirteenth to eighteenth. The graph indicates, the integral values of seventh-twelfth larger than the integral values of first-sixth, and the integral values of thirteenth-eighteenth larger than the integral values of seventh-twelfth. This result is corresponding references [1]-[4]. References have described that when a load becomes big, the integral value becomes big, too. And, when a load becomes big, dispersion is big, too.

3.3 Experiment 2

The purpose of this experiment is the movement confirmation of the system which showed it in Fig. 2. The outline of this system is as follows.

1) We measure operators' surface EMG signal collected when they operated those objects in the actual world, and it is recorded in ANN. In this experiment. Three kinds of loads that were displayed by JOYARM of the experiment 1 are recorded in ANN.

2) As for a goal for control of JOYARM is to measure recorded the surface EMG signal. As a result, the operator was given certain feeling of 1) virtually. In this experiment, it is tried to display the feeling of load1 of the experiment 1 to the operator.

3) The output values which got it in 2) is inputted to Personality Adaptable controller, and the controller decide the magnitude of the force which JOYARM displays

Next, the details of Personality Adaptable controller are shown. First, the purpose of the controller is stated. In order to give operators a certain feeling virtually, it becomes a goal of JOYARM to control it so as to close the pattern of operator's the surface EMG signal to those measured in the actual world. Here, ANN degree of the resemblance of recorded EMG and present EMG is outputted. Therefore, it is decided that the purpose of the controller increases the degree of a resemblance. In other words, in this experiment, in order to give the operators load1 feeling virtually, it becomes a goal of JOYARM to control it so as to close the output layer of the three neurons of the ANN are outputted to (load0,load1,load2)=(0,1,0).to those measured. So, it is represented the following algorithm. It is shown about the case that load1 feeling is given virtually to the operator. The flow chart is shown in Fig. 9. Input of the flow char is output of the ANN and the current order value of JOYARM, and the output is the next order value of JOYARM. The coefficient was decided as follows. α =0.8, β =1.

Next, detailed experiment condition is stated. The posture shown in Fig. 7, and measuring time is three seconds. And, The operator is the same person as the experiment 1. Then, a sensor isn't re-covered. The initial value of JOYARM was made load1. This experiment went twice on the same condition.

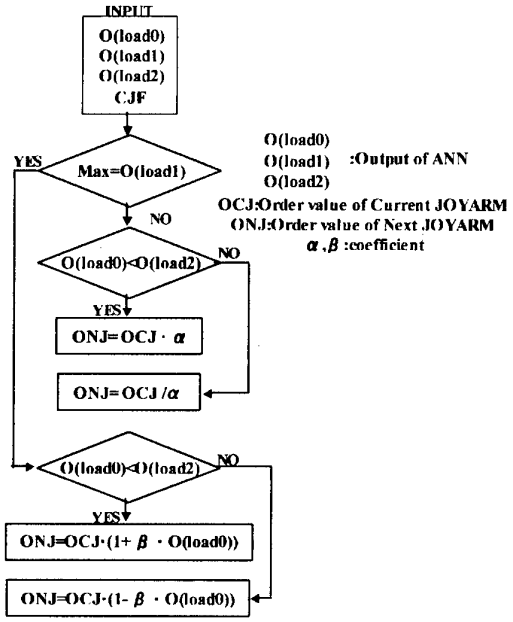


Fig. 9 flow chat

3.4 The result of the experiment 2

The experiment result is shown in six Figures. The resultA is the first experiment result. The resultA is shown in the right. The resultB is the second experiment result. The resultB is shown in the left. The horizontal axis is time with six Figures as well. Fig. 10 and Fig. 13 are the graphs which show two integral values (V·sec) of sensor0-sensor1. Fig. 11 and Fig. 14 are the graphs which

show the output value of the ANN. Fig. 12 and Fig. 15 are the graphs which show the values (kg·mm) displayed by JOYARM.

1) resultA

First, see to the integral value of the sensor1 in Fig 10. the minimum integral value is about 0.3(V·sec). The maximum integral value is about 0.45(V·sec). These are equivalent to the output value of the load1 in Fig. 8. As this result, The output of load1-neuron is always close to 1 in Fig. 11.

Second, it turns to Fig 11. It pays attention to about 2.5 seconds and 7.2 seconds. Here, load0-neuron output a big value, and load1-neuro output a small value. It is a cause by operator's physiology condition change. The integral value becomes small, and the output of neuron changes. The output of JOYARM is bigger than the one before in Fig. 12. This is because Personality Adaptable controller worked suitably.

2) resultB

First, it looks at to the integral value of the sensor1 in Fig. 13. The minimum integral value is about 0.3(V·sec). The maximum integral value is about 0.5(V·sec). These are equivalent to the output value of load1 in Fig. 8, too. As a result, the output of load1-neuron is always close to 1 in Fig. 14, too.

Second, it refer to Fig 14. It gives a closer look about 3.8 seconds and about 8 seconds. Here, load2-neuron output a big value, and load1-neuro output a small value. This result didn't happen in resultA. The output of JOYARM is bigger than the one before in Fig. 15. Personality Adaptable controller works suitably here, too.

3) resultA and resultB

The resultA is compared with the resultB. In the resultA, the integral value at the start of the experiment is small. Then, it increases. In the resultB, the integral value at the start of the experiment is big. Then, it decreases. The tendencies are different, because operator's physiology condition changed. This system ran normally in such a case as well.

4. Conclusions

In this paper, The following were confirmed.

1) Surface EMG signal relates to generated force.

2) Surface EMG signal sometimes vary according to time.

3) Personality Adaptable controller was made, and a performance was confirmed.

4) Personality adaptable type force feedback device system was made, and a performance was confirmed.

References

[1]Carlo J. De Luca: The Use of Surface Electromyography in Biomechanics, *Journal of Applied Biomechanics, volume 13, number 2, May 1997.*

[2]Carlo J. De Luca: Surface Electromyography: Detection and Recording, http://www.delsys.com(1996).

[3]Carlo J. De Luca: Physiology and mathematics of myoelectric signals, *IEEE Transaction on Biomedical Engineering, 26: 313-325, 1979.*

[4]Carlo J. De Luca: Myoelectric manifestations of localized muscular fatigue, *CRC Critical Reviews in Biomedical Engineering, 11: 251-279, 1984.*

[5]Funase, A., Yagi, T., Kuno, Y., Uchikawa, Y., Fundamental research for EEG interface: EEG and auditory/visual stimulus during eye-movements, Proc. of 1999 IEEE Int. Conf. on systems, Man and Cybernetics, Vol.II, 413-417, 1999.

[6]Kuno, Y., Yagi, T., Uchikawa, Y., A Study toward EEG Applications: Event -related Potential from Posterior Parietal Cortex, Proc.19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 1551-1553, 1997.

[7]Kuno, Y., Yagi, T., Uchikawa, Y., Development of Fish-eye VR system with Human visual function and Biological signal, Proc. of IEEE International Conference on Multi-sensor Fusion and Integration for Intelligent Systems, 389-394, 1996.

[8]O.Fukuda, T.Tsuji and M.Kaneko, Pattern Classification of EEG Signals Using a Log-Linearlized Gaussian Mixture Neural Network, Proceedings of 1995 International Conference on Neural Networks, 2479-2484, 1995.

[9]T.Tsuji, H.Ichinobe, O.Fukuda and M.Kaneko, A Maximum Likelihood Neural Network Based on a Log-Linearlized Gaussian Mixture Model, Proceedings of 1995 International Conference on Neural Networks, 1293-1298, 1995.

[10]K.Ito, T.Tsuji, A.Kato and M.Ito, EMG Pattern Classification for a Prosthetic Forearm with Three Degrees of Freedom, Proceedings of the IEEE International Workshop on Robot and Human Communication, 205-210, 1992.

[11]Akihito SANO, Hideo FUJIMOTO, Akihiro SAKAKIBARA and Sique YANG, Auxiliary Operation Suited to Intentional Positioning Accuracy, Robotics society of JAPAN, 1998

[12]T.Yoshikawa, Dynamics Shaping in Robot Force Control and Artificial Reality, Proc. of the 6th Int. Conf. on Advanced Robotics ('93 ICAR), pp.3-8, 1993

[13] T.Yoshikawa and H.Ueda, Haptic Virtual Reality: Display of Operating Feel of Virtual Objects, Preprints of the Int. Symp. of Robotics Research, 1995

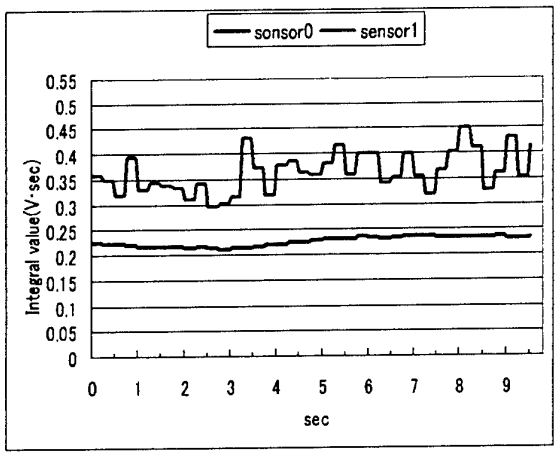


Fig. 10 The integral of EMG experiment 2(resultA)

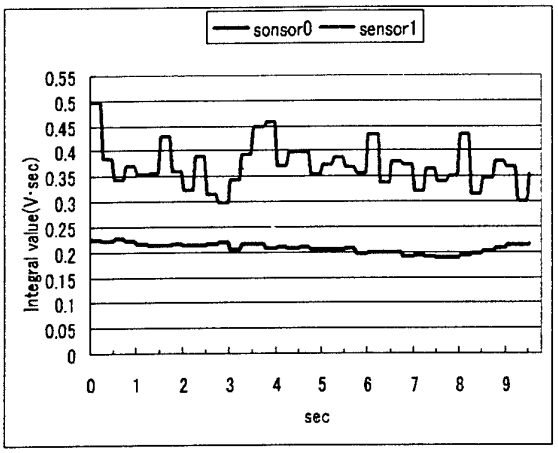


Fig. 13 The integral of EMG experiment 2(resultB)

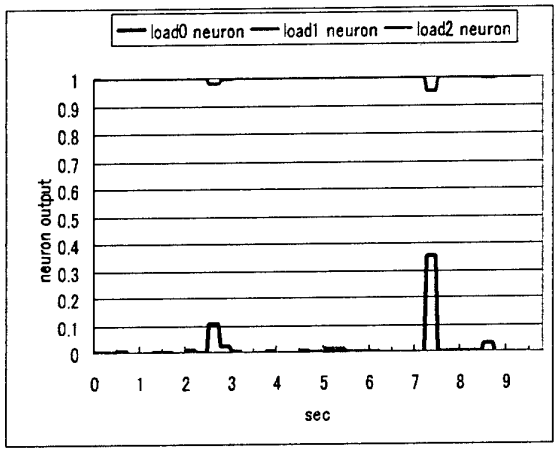


Fig. 11 The output of ANN(resultA)

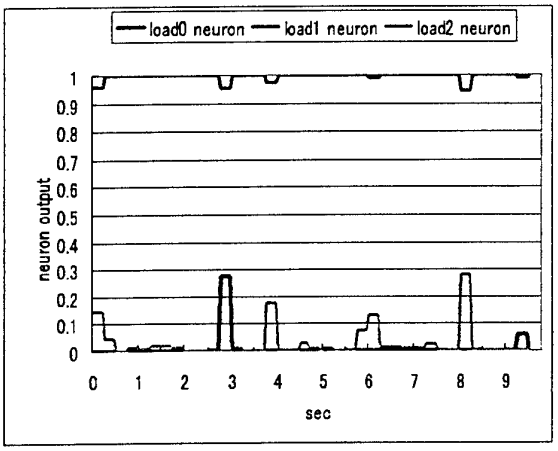


Fig. 14 The output of ANN(resultB)

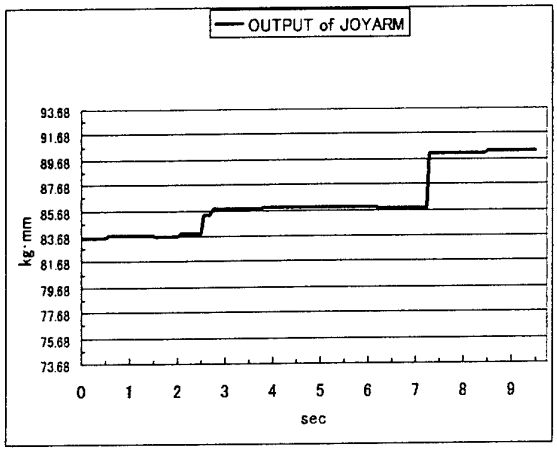


Fig. 12 The output of JOYARM(resultA)

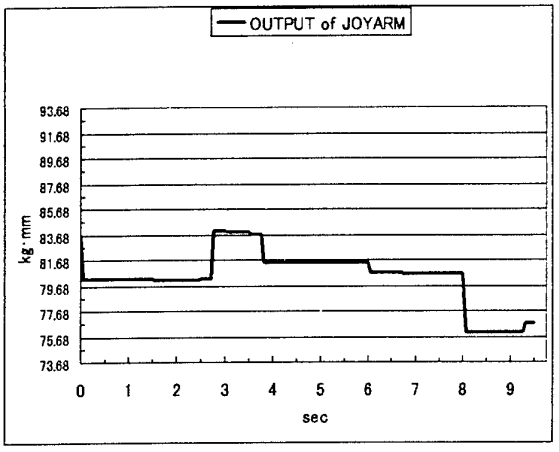


Fig. 15 The output of JOYARM(resultB)

Molecular Virtual Reality System with Force Feedback Device

Hiroshi Nagata[1,2], Eriko Tanaka[3], Masaaki Hatsuta[3], Hiroshi Mizushima[1], and Hiroshi Tanaka[2]

National Cancer Center Research Institute[1]
5-1-1, Tsukiji, Chuo-ku, Tokyo 104-0045, Japan
Tokyo Medical and Dental University[2]
1-5-45, Yushima, Bunkyo-ku Tokyo 113-8549, Japan
Mitsubishi Space Software Co. Ltd. [3]

hnagata@ncc.go.jp, erico@tkb.mss.co.jp, max@tkb.mss.co.jp, hmizushi@ncc.go.jp, tanaka@cim.tmd.ac.jp

Abstract

We designed a novel concept of computer aided drug design system using virtual reality technologies, in particular the tactile sense technology, and developed a prototype. The most characteristic function of the system is enabling its user to "touch" and sense the electrostatic potential field of a protein molecule. The user can scan surface of a protein using a globular probe, which is given an electrostatic charge, controlled by a force feedback device. The electrostatic force between the protein and the probe is calculated in real time, and immediately fed back into the force feedback device. The user can easily search interactively for positions where the probe is strongly attracted to the force field. Such positions can be regarded as candidate sites where small chemical groups corresponding to the probe, functional parts of lead compounds, can bind to the target protein. Certain limitations remain, for example, only ten protein atoms can be used to generate the electrostatic field. Furthermore, the system can use only an globular probe, rather than drug molecules or small chemical groups. These limitations are due to our computer resources being insufficient. However, our prototype system has the potential to serve as a new application method as well as being applicable to conventional VR technologies, especially to force feedback technologies.

Key words: Force Feedback, Virtual Reality, Drug Design, PHANToM, Electrostatic Potential

1. Introduction

We developed a new drug design strategy utilizing virtual reality (VR) technologies, focusing especially on tactile sense technology. Then, we designed a molecular VR system for drug design according to this strategy, and developed a prototype. The prototype enables users to tactually sense electrostatic force fields surrounding proteins using a force feedback device. Users can scan protein molecules with various probes, which represent chemical groups and small molecules capable of becoming parts of drug molecules. Our concept and method are anticipated to be useful for designing new drugs in the post genome age.

Genome studies have advanced greatly over the past decade[1, 2]. Many genes related to various diseases have been elucidated. Because each gene encodes the design for a protein, if the sequence of a gene governing a certain disease is decoded, it becomes possible to predict and analyze the molecular structure of the protein encoded by the gene[3]. If the protein structure is determined, it may be possible to design new compounds, candidates for new drugs, which can specifically bind the protein[4]. This is one of the important strategies of drug design based on genome science. Realization of this scenario would be feasible if the advanced computer science and engineering now available are applied creatively. Many drug design support systems have been developed for use in universities and pharmaceutical companies[5,6,7].

However, such tools are not easy to use. In particular, these tools are not suitable for interactively manipulating molecules. Moreover, their functions are insufficient to express interactions between a drug molecule and a protein. Therefore, it is difficult to reflect the insights and experiences of the drug designer into new drug designs employing these conventional tools.

We speculated that these problems could be solved using VR technologies, because VR is suitable for achieving

excellent user interfaces. The incorporation of VR into molecular science has only just begun. Several types of software for displaying 3-D models of bio-molecules have been developed. Most of them are implemented using VRLM (Virtual Reality Modeling Language)[8,9,10]. However, drug design support using force feedback technology is still in its infancy. In this study, we attempted to use force feedback technology for molecular design, and succeeded in developing a prototype for a unique drug design system. We consider force feedback technology have enormous potential for improving the methodology of drug design.

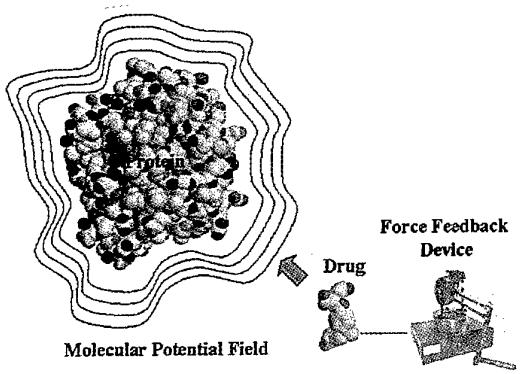


Fig. 1 Basic concept behind applying the force feedback technology to drug designing

2. The New Concept of The New Drug Design Method

The targets of drugs are protein molecules. Proteins are the only products of genes. These proteins have peculiar structures and functions necessary for maintaining life. A drug molecule binds only with a specific site of a specific protein, which is called the "target protein", and obstructs the function or changes the structure of the protein. The medicinal effect is the result.

Drug molecules approach the specific binding sites of target proteins guided by physical-chemical potential fields surrounding the proteins. The most important potentials are the electrostatic potential, the van der Waals potential, and the hydrogen bond potential. The van der Waals and hydrogen bond potentials, which operate only over short distances, work mainly to assure the final binding of drugs and proteins. In contrast, the electrostatic potential is thought to play major roll in attracting drug molecules to nearby the binding sites, because it can work over relatively long distances. Several experiments and simulations support this hypothesis[11,12].

The strength and sign of the electrostatic force produce drug molecule changes by altering molecular structure, the distance to the protein, and both posture and direction. Therefore, it is difficult to visually express the general electrostatic force potential commonly affecting all drugs.

However, it is feasible to make the electrostatic potential field available for tactile sensation using force feedback technology. Figure 1 illustrates this concept.

3. Potential Force Field

The molecular potential field surrounding a protein consists of electrostatic, van der Waals, and hydrogen bond potentials. The total potential energy between a drug molecule and a protein can be expressed by the following formula (1)[13,14,15].

$$E = \sum_{v=1}^{V}\sum_{w=1}^{W}(E_{lj}(v,w) + E_{el}(v,w) + E_{hb}(v,w)) \quad (1)$$

In the formula, V and W are the numbers of atoms in the drug and the protein, respectively. E_{lj}, E_{el}, and E_{hb} are the van der Waals potential energy, electrostatic potential energy, and hydrogen bond potential energy between the v'th atom of the drug and the w'th atom of the protein, which are given by formula (2) to (4)

$$E_{lj}(v,w) = \frac{A}{d_{vw}^{12}} - \frac{B}{d_{vw}^{6}} \quad (2)$$

$$E_{el}(v,w) = \frac{p_v q_w}{K}\left(\frac{1}{d_{vw}} + \frac{(-)/(+)}{\sqrt{d_{vw}^{2} + 4 s_p s_q}}\right) \quad (3)$$

$$E_{hb}(v,w) = \left(\frac{C}{d_{vw}^{6}} - \frac{D}{d_{vw}^{4}}\right)\cos^{k} \quad \cos^{2} \quad (4)$$

In these formulas, d_{vw} is the distance between the v'th and the w'th atoms; p_v and q_w are the electrostatic charges of the atoms;. K is the combination of a geometrical factor and natural constants; □ is the dielectric constant of the protein surface; □ is the dielectric constant of the solvent; s_p is the depth of the m'th atom of the drug molecule on the protein surface; s_q is the depth of the n'th atom of the protein molecule on its surface; dmn is the distance between the m'th atom of the drug molecule and the n'th atom of the protein molecule.

The forces of each potential exerted on the v'th atom by the protein are given by formulas (5) to (7) as summations of differentiations by the distance d_{vw} of these formulas.

$$F_{lj}(v) = \sum_{w=1}^{W}(-12\frac{A}{d_{vw}^{13}} + 6\frac{B}{d_{vw}^{7}}) \quad (5)$$

$$F_{el}(v) = \sum_{w=1}^{W} \left(-\frac{pq}{K} \left(\frac{1}{d_{vw}^{\ 2}} + \frac{d_{vw}(\ - \ )/(\ + \ )}{(d_{vw}^{\ 2} + 4 s_p s_q)^{3/2}} \right) \right) \tag{6}$$

$$F_{hb}(v) = \sum_{w=1}^{W} \left(\left(-6\frac{C}{d_{vw}^{\ 7}} + 4\frac{D}{d_{vw}^{\ 5}} \right) \cos^{k} \ \cos^{2} \ \right) \tag{7}$$

The total force exerted on the drug is given by formula (8).

$$F_{Total} = \sum_{v=1}^{V} (F_{lj}(v) + F_{el}(v) + F_{hb}(v)) \tag{8}$$

Our ultimate goals are to calculate and interactively feedback the F_{Total} to a force feedback device in real time. However, achieving these goals simultaneously appears to be very difficult. Therefore, we designed and implemented the simple prototype described below.

4. System Concept, Requirements, and Design of the Prototype

We designed and developed a prototype system by which the variations in the electrostatic force can be felt while scanning the surface of the protein with the drug molecule as a probe. The following are the system concepts for our prototype system.

(a) A protein molecule and a drug molecule are placed in a VR space. The molecules are displayed as 3-dimensional computer graphics.

(b) The drug molecule is moved and its posture controlled by using a force feedback device.

(c) The drug molecule is used as a probe, and with this probe, the user can scan the surface electrostatic potential field of the protein. Atoms of the molecules are assigned sizes based on their Van der Waals radii, and these are restricted to avoid adherence to each other.

(d) The electrostatic force, which the probe receives from the potential, is calculated and fed back to the force feedback device in real time.

The requirements for the prototype system are as follows.

Computers: A graphic workstation (OCTAINE MX1) and a Windows NT PC(MMX Pentium II).

Force Feedback Device: PHANToM™ Desktop

Force Feedback: Output 0 to 1.5 N(Newton). 1au (atomic unit) = 1N.

Force Potential: generated using a maximum of 10 protein atoms.

Molecular Graphics: Space filling model[16], Connolly model[17,18].

Probe: Spherical. The user can set the charge and the radius. Position is input by PHANToM™.

Manipulation of protein: Rotation by X-Y-Z axis.

A graphic workstation is used to provide a graphical user interface and display a VR space. A Windows NT personal computer (PC) is used to control the PHANToM™. Initially, we planned to draw real time molecular graphics using the same PC, but this was difficult because the CPU power was insufficient. Therefore, we divided the software between two computers. The position of the probe is input by the PHANToM™ interactively. The PC immediately calculates the electrostatic force working on the probe and then feeds this information back to the PHANToM™. The force exerted on the probe is fed back to the PHANToM™ according to a linear relationship based on 1AU(atomic unit) = 1N(Newton). However, we limited the maximum output of the PHANToM™ to 1.5N, to avoid breakdowns.

It is necessary to calculate the electrostatic force on the probe using all atoms of the protein, but we had to limit the number of atoms to 10 to calculate the power in real time. As mentioned above, the probe is a spherical object, and an arbitrary radius and charge can be assigned to it. Posture control of the probe was not attempted at this time.

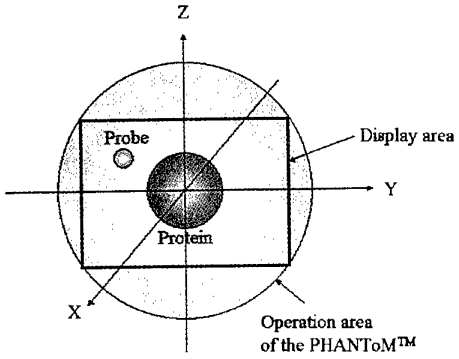


Fig. 2. VR space coordinates

We coordinated the VR space as shown in Figure 2. We defined the centers of the PHANToM™ and the protein

so as to be corresponding to the center of the VR space. The radius of the VR space was defined as being equal to the operation radius of the PHANToM™. To facilitate scanning, the protein should be rotatable to the X-Y-Z axis of the VR space.

The prototype system was designed as shown in Figure 3. The PC and the workstation are linked by a TCP/IP LAN, which inter-exchanges the data on the coordination of the probe and the protein.

The workstation software provides a graphical user interface, which consists of functions for drawing and rotating a protein molecule, transmitting data to the PC, and drawing and moving the probe in the VR space according to the coordinate data transmitted from the PC. It also provides the user interface needed to change the properties of the probe. Several kinds of probes can be registered, and the user can select and change them interactively. Protein 3D data can be obtained from the Brookheaven Protein Data Bank over the Internet ([On-line]. Available: http://www.rcsb.org/pdb/). The user selects ten atoms to calculate the electrostatic force. The MOPAC, among the most popular software for computational chemistry, is used to calculate the charges of the selected atoms.

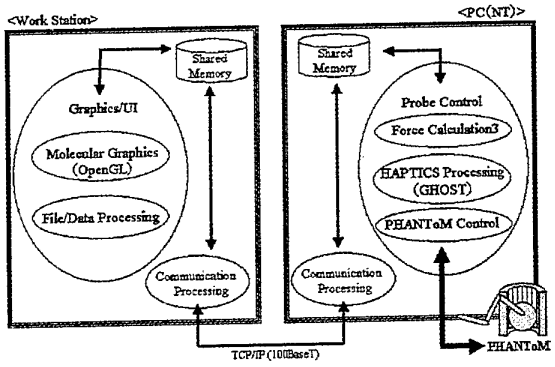


Fig.3 System design of the prototype

The properties of the probe, the protein structure data, and information on the rotation of the protein and the 10 atoms selected by the user are stored in shared memory and sent to the PC. On the PC, the electrostatic force is calculated and fed back to the PHANToM™. The PC also calculates the coordinates of the probe in the VR space according to the movements of the PHANToM™, and sends the results to the workstation.

5. Implementation and Results

Figure 4 shows the user interface of the workstation software. The graphical user interface was implemented using Open GL, and the internal processing functions were implemented using C++ programming language. The left half of the user interface shows the VR space, in which a protein is displayed in it. Prominently displayed points are the atoms selected to generate the force field. Several control buttons are also implemented in this area. All of the buttons are implemented in VR, and they can be selected by using the PHANTOM™. The right half shows system status and the communication situation with the PC.

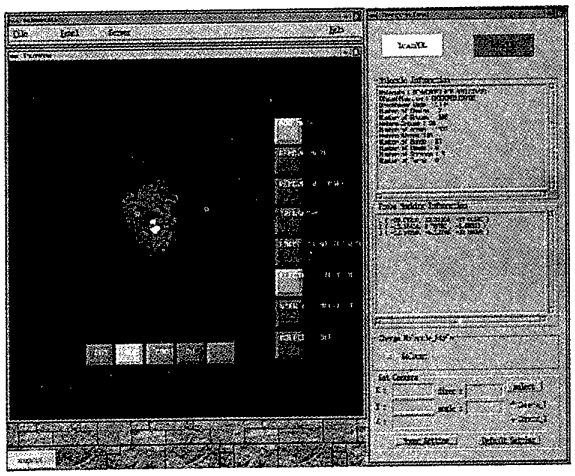


Fig. 4 User interface of the workstation software

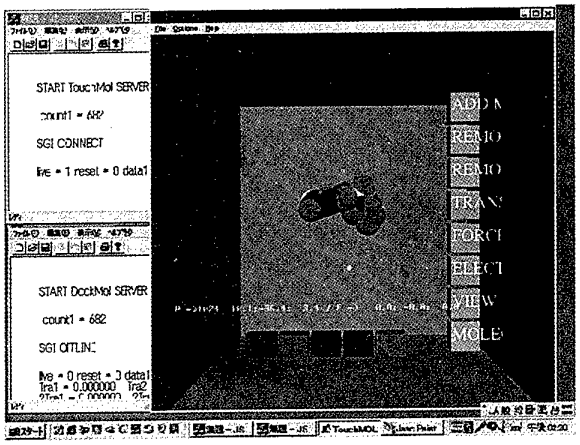


Fig.6 User interface of the PC software

The PC software was implemented using Visual C++ and the GHOST, the control function library for the PHANToM™. Figure 5 shows the user interface of the PC software. It has two windows, one is for displaying the same VR space as that of the workstation, and another is to monitor the communication status. The quality of the molecular graphics is much lower than that of the workstation, such that the VR space of the PC is mainly for debugging. However, it may be possible to generate and display high quality molecular graphics on the PC when its CPU power is greatly improved. The workstation will become unnecessary,

and the system design will be simpler.

We carried out various tests to confirm the system to be correctly implemented. For example, a virtual atom with a radius of 1 (approximately equal to that of an oxygen atom) was given a charge -1 and placed in the VR space. We then scanned it using a probe with a radius of 0.5 (approximately equal to that of a hydrogen atom) and a charge of +1. The probe was thereby confirmed to be strongly attracted to the virtual atom. The electrostatic force was in inverse proportion to the second power of the distance. It was also felt that the attractive force increased rapidly as the probe approached a virtual atom, and that the force was rapidly attenuated as the probe was moved away. We also increased the number of virtual atoms and repeated the tests, ultimately concluding that the system had been implemented correctly.

Next, we evaluated the system employing actual protein data. We used several enzymes which are known to be targets of anti-cancer drugs. For example, dihydrofolate reductase, one of the most important enzymes for cancer chemotherapy, was prepared. The binding sites of these drugs have been identified. From atoms that form the binding site, ten atoms on the exposed surface of the protein were selected, and the force field was generated. The charges of these atoms were calculated by MOPAC. We scanned the force field using the PHANToM™ (Figure 6), and succeeded in "feeling" the complex structure of the electrostatic force field. Because the probe was subjected to attractive and repulsive forces from the ten atoms simultaneously, the direction and the strength of the force changed even when the probe was moved only slightly. Moreover, it was possible to "feel" situations in which the probe was not able to easily escape local minimal points when captured by such points. It was confirmed that the force changed when the radius and charge of the probe were changed. These results confirmed the ability to scan and "feel" the electrostatic potential field of a protein using tactile sense technologies.

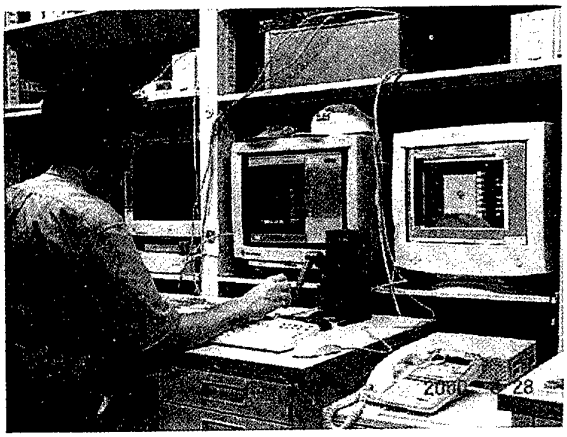


Fig. 6 Illustration of the scanning of a molecule using the PHANToM™

6. Discussion

We developed a new drug design concept based on force feedback VR technology, and implemented a prototype system which enables users to "feel" the electrostatic potential field surrounding a protein molecule. The prototype still has several limitations. It cannot account for all electrostatic interactions between the protein and the probe, but we believe that our system, with certain improvements, will be useful for molecular design in the future.

To improve the system sufficiently to allow practical use, we must overcome several technological hurdles. First, it is necessary to improve the computer power by 10 to 100 fold. Our prototype can use only 10 atoms to generate the electrostatic field, but the drug binding site of a protein is usually composed of several hundred atoms. Network parallel processing may be an effective means of to increasing the calculation power by tens of folds.

At this time, we used only the electrostatic potential. However, several other potentials including van der Waals potential and hydrogen bond potential are more important than the electrostatic potential in the final stage of creating chemical bonds between a drug and a protein. It is necessary to consider these potentials in searching for probe binding sites more accurately. To achieve this, far greater computer resources may be necessary.

It is also necessary to improve the probe. Herein, the probe was globular and had a single charge. However, our goal is to use a drug molecule as the probe. A drug molecule generally consists of 10 to 100 atoms, which have individual charges and various other chemical properties. Additionally, drug molecules have their unique shapes and structures. Therefore, it will necessary to use the posture control if drugs are to serve as probes. We plan to introduce the control theory of a robot arm into our system. We believe network parallel computing may also be needed to achieve all of the calculations in real time.

References

1. A.Kormberg: The Golden Helix. Univ. Science Books (1996)
2. J.Refkin: The Biotech Century : Harnessing the Gene and Remaking the World. *J P Tarcher* (1999)
3. BR.Glick and JJ.Pastermak,: Molecular Biotechnology. *Amer Society for Microbiology* (1998)
4. S.Wo-Pong and Y.Rojanasakul (Edited): Biopharmaceutical Drug Design and Development. *Humana Press* (1999)
5. F.Ooms: Molecular modeling and computer aided drug design. Examples of their applications in

medicinal chemistry. *Current Medicinal Chemistry* 7, pp141-58 (2000) Feb;7(2):141-58

6. JS.Dixon: Computer-aided drug design: getting the best results. *Trends in Biotechnology.* 10, pp357-63 (1992)

7. NC.Cohen, V.Tschinke: Generation of new-lead structures in computer-aided drug design. *Progress in Drug Research,* 45, pp205-43 (1995).

8. AL.Ames, DR Nadeau, and JL.Moreland: Vrml 2.0 Sourcebook. *John Wiley & Sons,* (1996)

9. J.Reichert, A.Jabs, P.Slickers, J.Suhnel: The IMB Jena Image Library of biological macromolecules. *Nucleic Acids Reseach,* 28, pp246-9(2000)

10. G.Moeckel, M.Keil, T.Exner, J.Brickmann: Molecular modeling information transfer with VRML: from small molecules to large systems in bioscience. *Pacific Symposium on Biocomputing* pp327-38 (1998)

11. RC.Wade, RR.Gabdoulline, SK.Ludemann, V.Lounnas: Electrostatic steering and ionic tethering in enzyme-ligand binding: insights from simulations. *Proceeding of the National Academy of Science. U S A,* 95, pp5942-9 (1998)

12. J.Antosiewicz, ST.Wlodek, & JAMcCammon: Acetylcholinesterase: role of the enzyme's charge distribution in steering charged ligands toward the active site. *Biopolymers,* 39, pp85-94(1996)

13. R.C.Wade, K.J.Clark & P.J.Goodford: Further development of hydrogen bond functions for use in determining energetically favorable binding sites on molecules of known structure. 1. Ligand probe groups with the ability to form two hydrogen bonds, *Journal of Medicinal Chemistry*, 36, pp.140-7 (1993)

14. R.C.Wade, R.R.Gabdoulline, S.K.Ludemann, & V.Lounnas: Electrostatic steering and ionic tethering in enzyme-ligand binding: insights from simulations, *Proceedings of the National Academy of Sciences of the United States of America*, 95, pp.5942-9 (1998)

15. C.A.Reynolds, R.C.Wade, & P.J.Goodfold: Identifying target for bioreductive agents: using GRID to predict selective binding regions of proteins, *Journal of Molecular Graphics*, 100, pp1-3-8 (1989)

16. J.Jimenez, A.Santisteban, JM.Carazo & JL.Carrascosa: Computer graphics display method for visualizing three-dimensional biological structures. *Science*, 211, pp661-6 (1981)

17. R.Langridge, TE.Ferrin, ID.Kuntz, ML.Connolly: Real-time color graphics in studies of molecular interactions. *Science* 211, pp661-6(1981)

18. **ML.Connolly: The molecular surface package.** *J Mol Graph.* 11, pp139-41(1993)

Controlling Two Remote Robot Arms with Direct Instruction using HapticMaster and Vision System

Takao HORIE, Norihiro ABE, Kazuaki TANAKA, Hirokazu TAKI
Faculty of Computer Science and System Engineering Kyushu Institute of Technology
Kawazu 680-4,Iizuka, Fukuoka, Japan, 820-8502
Faculty of System Engineering Wakayama University
Sakae, Wakayama, Japan
horie@sein.mse.kyutech.ac.jp

Abstract

. This paper proposes a system that helps an operator control a manipulator using a direct Tele-guidance method that allows him to grasp with a data glove and teach the real manipulator to assemble/disassemble mechanical parts. The direct Tele-guidance is realized by calculating joint angles from the position and orientation of the end effector, which are specified with his data glove. Task environment are captured from two pan tilt TV cameras which are controlled according to his head movement, and he can see the stereoscopic image of it through an HMD. When deciding the end effector has approached to an object to be grasped, or a part grasped has come up to a target object, he has only to use a haptic device to continue his operation. The device transmits to him the force and torque values added to the force torque sensor attached to the end effector. This information allows him to intuitively recognize the state of the effector together with the visual information and makes it possible to precisely control it

Key words: Robot Arm , Haptic Master , Data Glove

1. Introduction

The use of robot arm increases in various fields late years. And much research is done about the use of robot arms. As a method to control robot arms, joint instruction method is general which gives information such as a coordinate or each joint angle of a robot arm. When we want to give instruction to several robots in order to have them collaborate, we must give a sequence of instruction to them while considering the position relation between their arms. It is not easy for a beginner to give correct instruction to the robot arms.

On the other hand there is a direct instruction method as a simpler instruction method. This is a method to give robot arms their movement with a man grasping the arms. It is not an appropriate instruction method when there are more than one robot in the environment, or they are in the remote place or the environment is dangerous for a man.

We propose a system in a virtual reality environment that helps us to give several robot arms instructions using t simple instruction method and realizes collaboration with several robot arms which are located at a distant place. In the task of machine part assembly, as it is difficult for robot arms to decide with any computer vision system whether the parts to be assembled have been exactly mated, they may be not able to finish the operation. This system successfully gives an operator a feeling of fitting between parts to be assembled by using a force feedback device together with the direct instruction method.

2□GENERAL BASIC IDEA

2.1 Instruction

Table There are two ways that we use to give instruction to a robot. One is an indirect instruction method that gives the joint angle with a teaching box or robot language, and another is a direct instruction method that gives direct movement to a robot by grasping it with a hand. As for the indirect instruction, we can instruct correct movement to a robot arm by giving numerical value directly, but on the other hand, it is difficult to give complicated instruction,

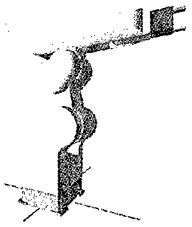


Figure 1. 6 DOF model.

Figure 2. 5 DOF model

and it will take a long time to finish the instruction process. The direct instruction method can let the instruction finish in short time as it helps us operate a robot by hand directly. On the other hand, there is the problem that instruction of minute movement is difficult with the direct instruction method, but there is an advantage that it permits us to intuitively operate a robot by devising scale conversion in the movement speed of an arm.

2.2 World Tool Kit

The World Tool Kit of SENSE8 Company is used to make it easy to construct the virtual reality environment. World Tool Kit is a library of the C language function and

offers several means for producing a complicated application which requires various objects in a virtual environment to works as well as in the real world.

2.3 Force feedback

It is considered that we rely on not only vision system but also the haptic sensation returned to our hand when we want to insert a peg in a hole. From this point of view, it is considered that haptic information should be given to an instructor of the virtual robot arm. In this study a force torque sensor is installed at the end effector of the robot arm and the force to be added to the effector is transmitted to the HAPTICMASTER, which is one of haptic feedback devices, by way of PC. When the part which the robot arm grasps touches other parts, the instructor can get the sense of touch from the HAPTIC MASTER and it helps him attain precise instruction.

3. Modeling

3.1 Modeling of robot arm

A robot arm with 5 degrees of freedom and the one with 6 the degrees of freedom are modeled in the virtual environment. The robot arm is decomposed into several joint parts in order to make it easy to construct the models. The model s of the robot arms shown in Figure 1 and Figure 2 are constructed by combining the individual part into hierarchical structure.

3.2 The model of a hand

To directly instruct robot arm in virtual space, a virtual hand equivalent to the real hand becomes necessary. So the model is defined, which is consisted of 19 parts. Movement of this model is decided by referring to the position data obtained from data glove with three-dimensional position sensors.

4. Superimposing Two Virtual Spaces

In this system a TD-300 of Intergraph company is used to build virtual environment. Further two video cameras and an HMD are used in order to get a virtual task environment from a real environment including real robot arms and objects (Figure 3).

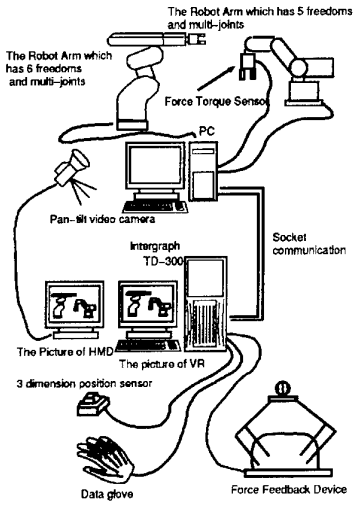
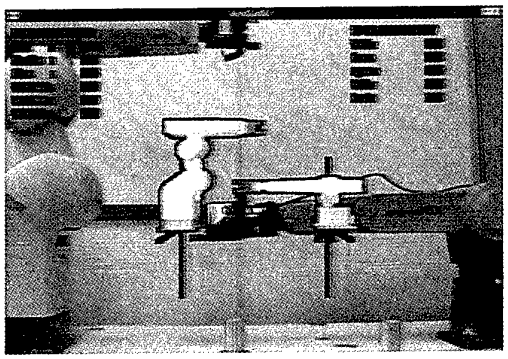
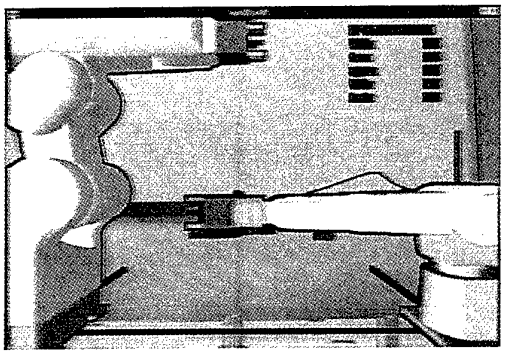


Figure 3. System organization

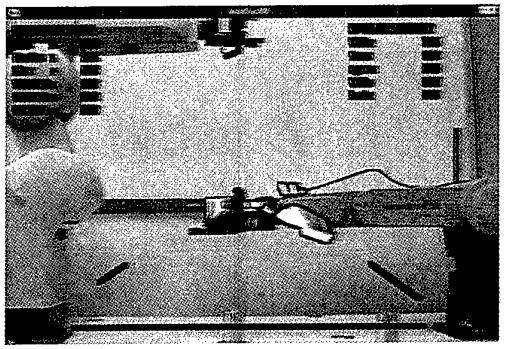
There are two virtual spaces; the first is the space including the virtual hand, and the second is the space constructed with stereoscopic images. In order to manipulate the virtual arm with the virtual hand, the two spaces must be coincided. The flow of work is shown in the following (see Figure 4).



(1)



(2)



(3)

Figure 4. superimposition of virtual space onto realspace

First, it makes the posture of the virtual robot arm and the posture of the fruit robot arm agree.After that, using the scan converter, it piles up the picture of the virtual robot onto the video camera picture.(Figure4-1) It judges the RGB value of the computer image in the threshold value and the scan converter displays a pixel with high RGB value on the video footage.

In other words, only the robot model can be piled up onto the video footage in making a background in the virtual space black.

However, only in this, there is a big difference in the display of the robot model and the display of the actual robot. In to adjust both of the virtual camera, the video camera positions, it becomes an approximately similar display (Figure 4-2).

Lastly, the robot model, too, changes to the black by the keyboard operation.With this operation, the robot model, too, becomes not displayed and only the model of the operator's hand is displayed (Figure 4-3).
4-3 of the figure is the picture that the hand of the operator is sinking into the actual robot arm.

5. Direct Instruction Method

We will describe a method to instruct a virtual robot arm with data glove directly. By referring to both information from 3 dimensional position sensors and that from the data glove, a virtual model of operator's hand is moved freely in the virtual space.

When grasping the virtual arm, which is the image of the real robot arm, and the operator successfully moves his hand to the point he intends, the system can lead the real arm to the corresponding point by transferring to the arm the value of each joint angle of the virtual arm. It is a fundamental problem in the direct instruction of robot arms that the system can correctly judge whether an instructor is going to move an arm or not. In this system, it does the interference judgment of the box which surrounds the whole model of the hand of the operator and the box which surrounds the fingertip (the black part) of the robot arm.
Then, when bent above above the angle of box fellow's contacting and moreover finger's there being which is constant, an operator is judged to try to move a robot arm about the system.
At the same time, does the system compute a distance between the model of the hand of the operator and two each of the robot fingertips, too, and which robot fingertip does the operator try to move or the judgment, too, goes.

And, it prepared the mode to do the collision detection of the robot and the robot to avoid the crash of two robots when two robot arms are intersecting work, too (Figure 5).

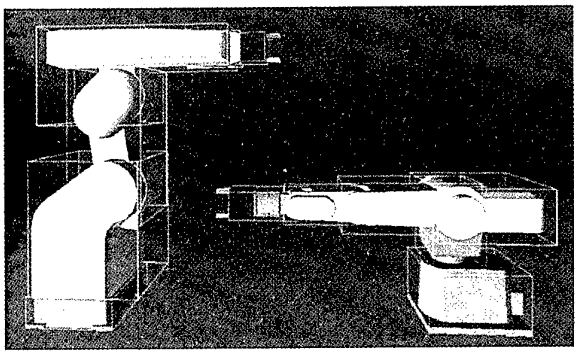


Figure 5. Bounding box of robot

5.1 Determining the posture of the arm

Each joint angle and the position of the robot are needed to compute the next posture of robot when an instructor leads the arm to the next position. In case of 5 DOF robots, the system can decide the posture from both the coordinate and the orientation of the arm end point and the constraint on the arm posture.
In case of the 6 DOF robot which was used by this research, on the structure, more than one answer exists. The candidacy of four answers is computed when the 6 DOF robot which was used by this research fixes the position and the posture of the fingertip in the relation of the design.

1: The value with positive $\theta 4$ and moreover the value with positive $\theta 5$
2: The value and moreover $\theta 5$ with positive $\theta 4$ have a negative value.
3: The negative value and moreover the value with positive $\theta 5$ by $\theta 4$
4: The negative value and moreover $\theta 5$ have a negative value in $\theta 4$.

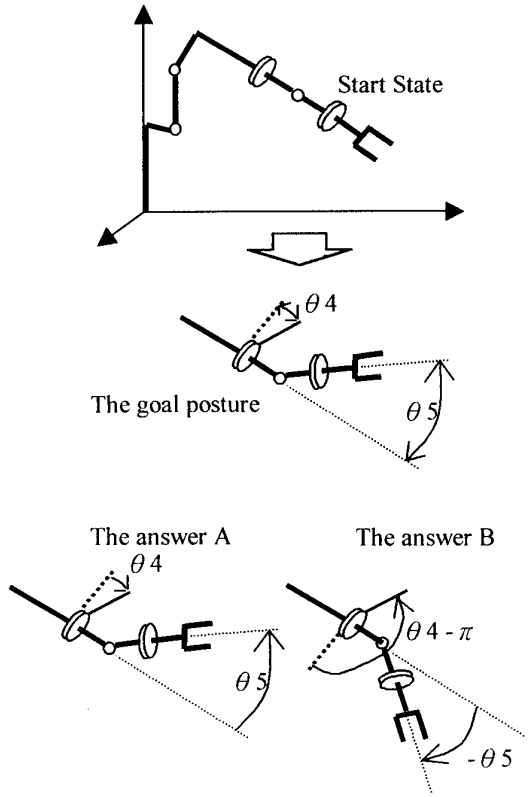


Figure 6. The choice of the operation of the 6 freedom degree arm

Both the direction of the positive and the direction of the negative are designed with 160 degrees by the operation range of $\theta 4$ of this robot arm.

It depends and the candidacy which is two that the absolute value of $\theta 4$ is within 160 degrees from four pieces of candidacy can squeeze(The answer A of figure 6 and the answer B of figure 6).

To choose the more suitable answer from these two answers, it considers that it is moving a robot by the direct instruction.

It is desirable to choose the answer as the movement of the robot arm can be continuously seen more smoothly.
Therefore, it thinks of updating a end effecter position every minute time and in the case, it thinks that it is basic and that the change quantity with robot arm joint angle should be small.
Therefore, in this system, it computes a difference from the posture of the robot before operation to each of the candidacy of two pieces of posture.
Then, it makes posture with the smaller operation quantity an answer(The answer A of figure 6).
This became able to do that the operator made to twist a robot fingertip move freely.

6. Detection of Force and Force Feedback

In the mechanical assembly, the fitting operation is frequently carried out. In the assembly operation, even if there are a few differences in position or orientation between two parts to be mated it is difficult to finish the operation.

To realize such operations with the robot arm, in addition to visual information introduction of haptic feedback one is needed which informs an operator of the time when they come in contact. If he could sense through the robot arm the information when two parts come in contact, as he will be able to control much delicately the robot arm using both visual information and haptic feedback one in the same way as in the real environment, the part assembly operation with robot arm will be successfully attained.

The system is needed which allows an operator to receive the haptic feedback information by using a force torque sensor and haptic feedback device.

The force torque sensor is installed at the end point of the arm and conveys the force acting on the arm end point to the computer. The force feedback device makes it easier for the operator to attain the fitting operation by informing him of the force acting on the arm.

6.1 Arm control using haptic feedback device

In all processes of instruction the force feedback device should be used, but it is difficult for us to control the robot arm freely with the device because the range of movement is restricted to the narrow range. In the work such as grasping or moving an object with a robot arm, which does not obviously need any haptic sensation, the direct instruction can be conducted with only a data glove and three dimensional position sensors.

The arm control with the haptic feedback device is necessary when the delicate control is demanded, except for the case when he confirms that an object grasped never collides with other objects. Right before the device is used to control the arm, the initial location of the haptic device is set to the location corresponding to the arm current position, afterwards the movement of the arm is controlled with the force feedback device (see Figure 7).

7. Experiment

7.1 The method of Experiment.

At first, both the virtual arm and the real robot arm are established in the home positions respectively. A goal is to assemble a part A and a part B shown in Figure 7. A remote direct teaching with a data glove is used in the first half part of the teaching process where the part A and part B are grasped, and they are come close each other. They are then put together while checking collision between them. A 5 DOF robot arm is controlled using a Haptic Master in the latter half of the teaching process.

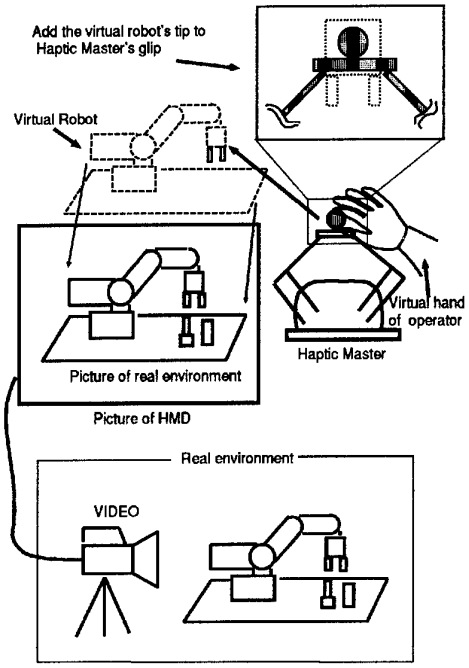


Figure 7.The initial position of a force feedback device

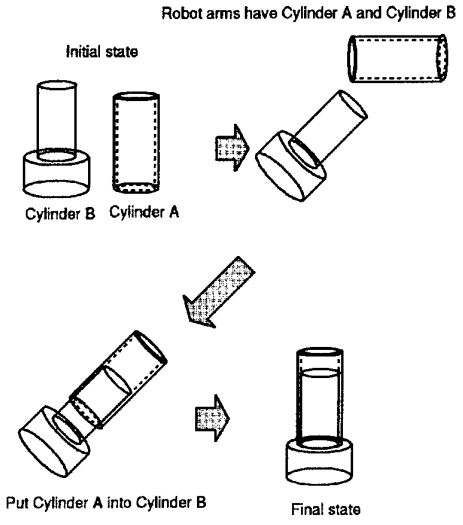


Figure 8 A goal assembly

7.2 Direct Teaching with Data Glove

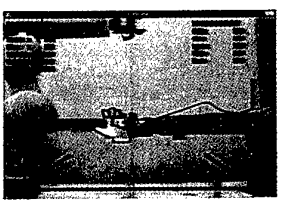
First, an end of a 5 DOF robot arm is grasped with a data glove and it is moved above the part A. While maintaining the arm end in the vertical direction if circumstances allow, after the middle part of the arm end is moved to the just above the part A, it is closed by keyboard input (Figure 9-4).

The 5 DOF robot arm end is grasped again, and the part A is lifted (Figure 9-5). Next, the 6 DOF robot arm end is grasped using a data glove. It is moved near the part B (Figure 9-6). Because the part A must be inserted into the part B, the 6 DOF robot arm is moved so that the head part of the part B can be grasped.

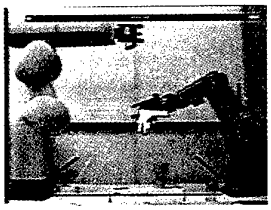
After the middle part of the end point is moved to just the side of the top of the part B, it is closed with keyboard operation (Figure 9-8). Next the part B is lifted with the 6 DOF arm (Figure 9-9).

Next in order to insert the part B into the part A, the head part of the part B must be turned aloft. So the 6 DOF arm end is spun 180 degrees using a data glove (Figure 9-10).

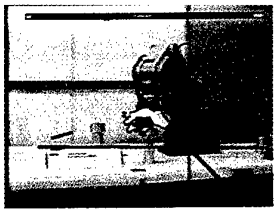
In the sequel, the 6 DOF point is moved within the work range of the 5 DOF robot arm, and is stopped there (Figure 9-12). Next the 5 DOF robot arm holding the part A is got hold of with a data glove and moved above the part B (Figure 9-13,Figure 9-14)



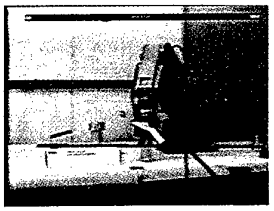
(1)



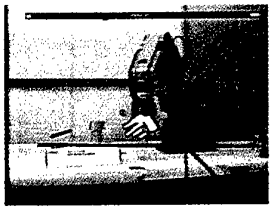
(2)



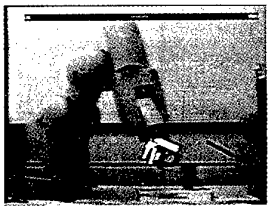
(3)



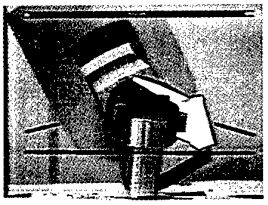
(4)



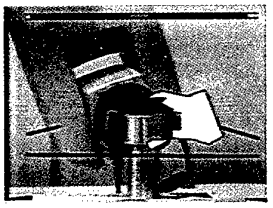
(5)



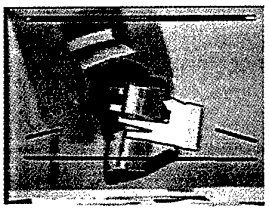
(6)



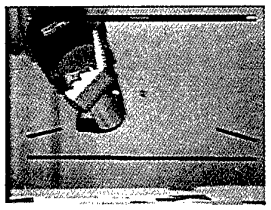
(7)



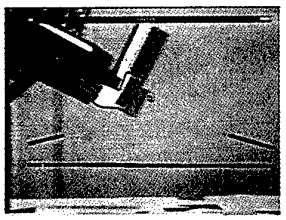
(8)



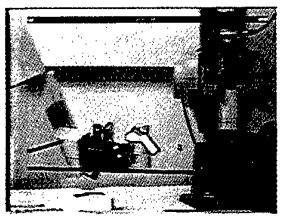
(9)



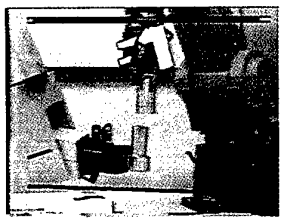
(10)



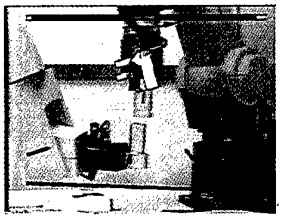
(11)



(12)

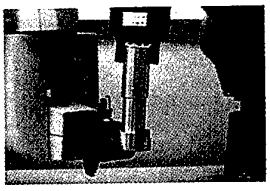


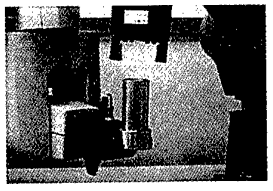
(13)



(14)

(15)





(16) (17)

Figure 9. Process of assembling task

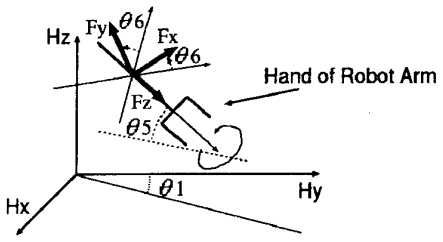
7.3 Control Using a Haptic Master

At first, when E of a keyboard is pushed, the coordinate of the arm end point at that time is set to the grip position of the Haptic Master. At the same time, a command that compels the sensor to output the force currently added to the force torque sensor. In this way the downward force caused

by the weight of the part grasped can be canceled. Afterwards, by giving 1/5 of movement of a Haptic Master to the robot arm end, it is control with the Haptic Master. Whenever the robot arm is operated, a value of the force torque sensor is read and given to the Haptic Master. This method allows an operator to sense the force acting on the arm end point through a Haptic Master when the grasped part comes in contact with other parts. The state of the Haptic Master immediately after the control with a Haptic Master starts is shown in Figure 9-15. An operator grasps the grip of a Haptic Master, and inserts the part B in the part A . He gives a command to open the 5 DOF arm end point from a keyboard, and separates the part B from the arm. (Figure 9-17).

When a robot arm is moved, the coordinate system of a force torque sensor installed in the arm end point also changes. Therefore, the state of the force torque sensor coordinate system is acquired from the posture of the arm, an output value adjusted to the coordinate system of the Haptic Master must be calculated. The technique is as follows.

First, a posture of the arm is acquired using where command. Next using OR command, force acting on the force torque sensor is detected. Using relation shown in Figure 10, a force component to be transmitted to the Haptic Master is computed by applying the following expression. (see Figure 11)



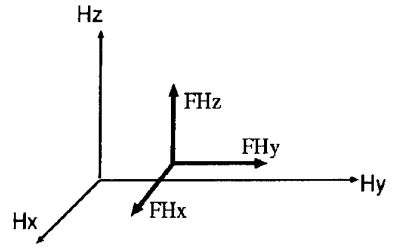
Fx,Fy,Fz are transformed into Hx,Hy,Hz

Figure 10 Coordinate of a robot hand

$$FHx = \{(Fy\cos\theta + Fx\sin\theta_6)\sin\theta_5 + Fz\cos\theta_5\}\sin\theta_1 - \{(Fx\cos\theta_6 - Fy\sin\theta_6)\cos\theta_1\}$$

$$FHy = \{(Fy\cos\theta + Fx\sin\theta_6)\sin\theta_5 + Fz\cos\theta_5\}\sin\theta_1 + \{(Fx\cos\theta_6 - Fy\sin\theta_6)\sin\theta_1\}$$

$$FHx = (Fy\cos\theta + Fx\sin\theta_6)\cos\theta_5 - Fz\sin\theta_5$$



Fx,Fy,Fz are output of Force Torque Sensor
Hx,Hy,Hz are coordinate system of Haptic Master
FHx,FHy,FHz are input for Haptic Master

Figure 11 Coordinate of HapticMaster

7.4 Results

The experiment is conducted in two ways; the first is controlling it only with a data glove, and the second is the one using the data glove and the force feedback device. Even if parts delicately touched each other when they come close to some extent without the force feedback device, no operator can notice the fact and in most cases the operation failed. On the other hand it was clearly easy to instruct the arm because he can detect the contact between parts in case the force feedback device is available, and he was able to have robot arms attain the movement such as rubbing them together.

8. Conclusion

We proposed a system that helps an operator telemanipulate two robot arms with a direct in struction method. And as a result cooperation with two robot arms was successfully implemented. At present, just one data glove and one Haptic Master are available, no co ncurrent manipulation of two arms is realized.

References

[1] Takao Horie, Kazuaki Tanaka, Norihiro Abe, Hirokazu Taki," Direct Teaching to a Virtual Robot Arm." International Conference on Virtual Reality and Tele-Existence(ICAT), pp.230-237,(1997)

[2] Kenji Funahashi, Takami Yasuda, Shigeki Yokoi, Jyun-ichiro Toriwaki:" Cooperattion model using both hand in virtual environment", Information Processing Society of Japan, Vol.39, No5, pp.1334-1341 (1998)

[3] Haruo Noma, Stutomu Miyasato, Ryouhei Nakastu, "Haptic feedback interface for cooperative manipulation of virtual objects", Information Processing Society of Japan, Vol.39, No5, pp.1343-1353 (1998)

[4] Hiroaki Yano, Hiroo Iwata," Software system for constructing virtual environment with haptic feedback", Virtual Reality Society of Japan, Vol.2, No1, pp.1-9 (1997)

ICAT 2000

Interactively Directing Virtual Crowds in a Virtual Environment

Tsai-Yen Li, Jian-Wen Lin, Yi-Lin Liu, and Chang-Ming Hsu
Computer Science Department, National Chengchi University
64, Sec.2, Chih-Nan Road, Taipei, Taiwan 11605, ROC
{li, s8512, s8506, s8505}@cs.nccu.edu.tw

Abstract

Simulation of emergent group behaviors for creatures such as birds and fishes has been widely used in computer animation. Although the same technique can be adopted to simulate crowds of virtual humans in a shared virtual world, it remains a great challenge to simulate the high-level intelligent behavior of a virtual human with planning capabilities. In this paper, we present a shared virtual environment crowded with real and virtual users. Virtual users, controlled by a world manager, are simulated in groups, each of which is led by an intelligent group leader. At run time a world manager can interactively assign a goal configuration to each group leader, and the system will automatically generate collision-free paths that bring the leaders to the goals. Since we allow multiple groups to move simultaneously in the virtual world, we adopt a decoupled path-planning approach, in which the paths being executed by the other leaders become the motion constraint of the current leader under consideration. The remaining members in a group follow the motion of the leader with emergent behaviors such as flocking. We believe that such an interactive interface will facilitate the simulation of controlled virtual crowds for applications such as 3D virtual shopping malls.

Key words: Virtual Crowd, Shared Virtual Environment, Decoupled Path Planning, Behavioral Animation, and Humanoid Simulation

1. Introduction

As 3D shared virtual environments are becoming prevalent in the cyberspace, the need for better authoring tools to direct groups of avatars also increase. For example, in a 3D virtual shopping mall, a well-controlled crowd of people will increase the realism of virtual shopping. The owners of virtual shops might want to hire crowds of virtual avatars to attract real users to their stores. However, most shared virtual environments today only accept real-user logins. Most of them do not have a flexible interface to adapt both virtual and real users. In addition, there are no good tools to quickly populate the world with virtual avatars that can be directed in an interactive manner.

In this paper we present a shared virtual environment (VE) system that allows coexistence of virtual and real users and enables interactive path planning for virtual crowds. The world is populated with groups of virtual users, controlled by a world manager. Each group is led by an intelligent group leader. At run time a world manager can interactively assign a goal configuration to a group leader through a graphical user interface at the VE server. The system will automatically generate collision-free paths that bring the leaders to their goals. Since we allow multiple groups to move simultaneously in a virtual world, the computational complexity of the involved path-planning problem is rather high. Therefore, we adopt a decoupled path-planning approach, in which the paths being executed by the other leaders become the motion constraint of the current leader under consideration. The remaining group members then take a more emergent strategy to follow their leaders.

We organize the remaining of the paper as follows. In the next section, we will review the researches pertaining to our work in artificial life and geometric planning. In the third section, we will give an overview of the architecture used in our VE system. In the fourth section, we will give a more detail description of the planning algorithm adopted in our system. Then we will present some implementation details and give some examples from our experiments. Finally, we will conclude the paper with some discussions on current limitation of our system and some possible future extensions.

2. Related Work

Simulation of emergent behaviors such as flocking has been widely used in creating realistic animations for groups of virtual creatures such as fishes or birds. [13][15] By applying simple emergent rules to each character, one can simulate realistic flocking behavior for animals. However, it is difficult to simulate a crowd of people simply with these principles because human, as an intelligent character, possesses higher degree of intelligence. In recent years, there have been many efforts in incorporating practical artificial intelligence techniques to create real-time animation. For example, a cognition model has been proposed in [7] to use a more complete control loop to simulate an intelligent

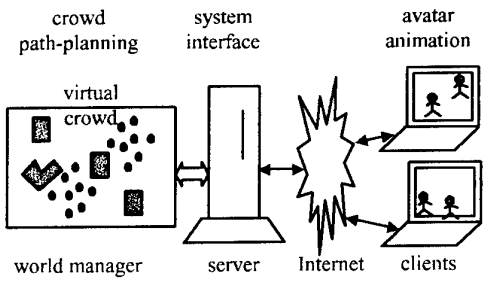


Figure 1. System architecture

character. Researches in virtual human also consider the problem of creating realistic humanoid group motions through various levels of controls. [4][5][11][12] However, most of these researches do not account for geometric reasoning capability such as path planning. On the other hand, motion-planning techniques have been successfully adopted for automatic movie generation[8] or customized tour guiding[10], although they usually focus on generating dexterous motions for a single human only. In robotics, efficient motion planning algorithms have been proposed to control more than one robot arm in an on-line manner[9]. However, we have not seen similar work been applied to simulate human crowds. Distributed interactive simulation (DIS) and shared virtual environments (SVE) have also been active research fields recently. Most research efforts are focused on system scalability, transmission efficiency, and scene management. In recent years, more and more SVE systems (such as ActiveWorlds[1] and Blaxxun's[3]) include programming interfaces for implementing virtual avatars (or called *bots*). However, they do not have systematic ways to simulate virtual crowds.

3. Overview of System Architecture

Our system is based on an open-source virtual environment (VE) system (VNet)[17]. This VE system adopts a client-server model that uses VRML[16] as its front-end 3D user interface. We augment the system with three major software modules to facilitate the control of group avatars: *system interface* module, *avatar animation* module, and *crowd path-planning* module, as shown in Figure 1.

First, the system interface module creates an interface at the server side between the VE server and the simulator for a world manager to control the motions of virtual crowds. Through this interface, a virtual user is treated in the same way as a real user at the VE server. From the graphical user interface of a client machine, a virtual avatar cannot be distinguished from a real avatar as seen by another real user. Second, the avatar animation module uses a modified messaging protocol used in VNet to send parameterized animations to the clients. With this module, these clients can convert motion-captured data on the fly into humanoid animations conforming to the VRML humanoid version 1.1 standard. Third, the crowd path-planning module is the key component that generates the motions for each group leader directed interactively by a world manager. It also includes a leader-follower steering module with flocking behaviors to generate motions for the remaining group members. The path planning and coordinating methods for multiple group leaders are described in details in the next section.

4. Planning for Crowd Motions

4.1. Problem description

The goal of our system is to provide an interactive interface for directing virtual avatars in a virtual environment. We are given a 2D polygonal description of the obstacles in the virtual world. Each of our avatars has three DOFs (x, y, θ) when they move on a plane. The parameter space for each avatar, called the *Configuration Space* (or *C-space* for short), is denoted by C_i. The overall C-space for the whole system, denoted by C, is the composite space of each individual C-space ($C_1 \times C_2 \times \ldots \times C_n$). At any time during the simulation, our system has to make sure that the generated motions for the virtual avatars be realistic and safe. In other words, the motions must be continuous in C and collision-free from other avatars and the obstacles in the environment. A virtual avatar should never make a move that will cause a collision. However, even if a virtual avatar does not make an illegal move, we can not guarantee that a collision will not happen when a real avatar intends to do so. However, except for this kind of situation, it is the job of the planning system to ensure the virtual crowds under its control do not collide.

In order to reduce the complexity of the planning problem, we assume that each avatar can be represented by an enclosing circle of radius r. Due to the geometric symmetry of a circle, we can reduce the degree of freedom for each avatar to two by temporarily ignoring the θ-dimension. The value for θ will be computed in a post-processing step after a path has been generated. For example, we can require that an avatar always face the tangential direction of a path. In order to facilitate collision detections in the planning process, we use a discrete approach by representing the polygonal obstacles with a bitmap and then grow the obstacle boundary by r to form the C-space of each avatar. This computation only needs to be done once when obstacle configurations are determined. The possible collisions between avatars are detected at run-time by checking the distances between the avatars.

4.2. The approach

Although the problem of path planning has been widely studied for the past three decades, one still cannot escape the curse of dimensionality. The planning problem becomes difficult for systems with high degrees of

freedom (DOFs) such as coordinating the motions of multiple mobile robots. The scenario of virtual crowds inherently also has such high complexity. For example, the dimension of the composite C-space (C) for the whole virtual avatars is $2m$, where m is the number of virtual avatars. Since the size of a C-space grows exponentially in the number of dimensionalities, a complete planning system deems to be infeasible. However, if we look at the problem of controlling virtual crowds interactively in a more practical manner, we can find several ways to simplify the planning problem and still make the solution interesting.

First, we assume that not every virtual avatar requires high-level planning. Instead, we assume that these simulated virtual avatars are in n groups. Each group, G_i consists of a leader, L_i, and a few followers, F_{ij} (where $i<=n$ is the index of a group and j is the index of a follower in its group). Only the leaders have path-planning capability and the followers will adopt local emergent rules to follow their respective leaders. With this approach, the number of avatars that require planning is greatly reduced, and the flocking behavior of a virtual crowd can also be achieved with traditional artificial life approaches.

4.3. Decoupled planning for group leaders

In robotics, the problem of path planning for multiple robots falls into two categories: *centralized* and *decoupled*. The centralized approaches consider the composite C-space of the whole system, which could be impractical to search exhaustively. On the other hand, the decoupled approaches usually only consider one robot at a time. In one such decoupled approach, each robot is planned independently and then their motions are then coordinated by velocity tuning techniques.[6] Another decoupled approach assumes that robot motions are generated sequentially and each robot is planned under the constraint of the robots whose motions are generated earlier.

In our crowd control system, we take the last decoupled approach by decomposing the overall planning problem for multiple virtual avatars into smaller subproblems. Each of these subproblems considers one virtual avatar at a time under the constraints of other avatars' motions. The same approach has been used in planning the motions of multiple robotic arms in an on-line manner.[9] Although this approach is not complete in nature, this planning scheme fits our application quite well since the needs for path planning happen sequentially. When the world manager directs a group leader by specifying a desired goal, the path planner is called on demand to generate a collision-free path based on the planned/scheduled motions of other group leaders.

At any time when the world manager would like to direct a group leader to a new goal configuration, the planner will try to generate a path that does not cause any

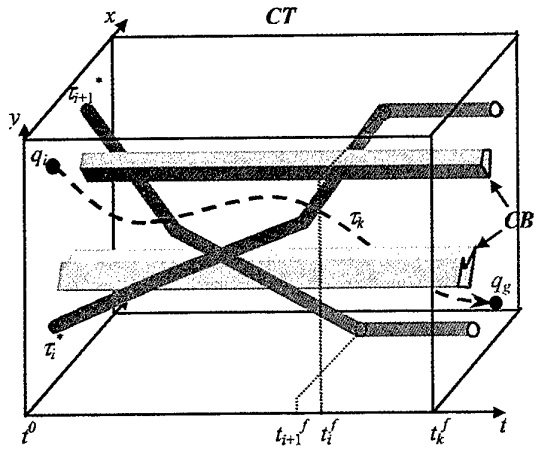


Figure 2. Searching for a feasible path in the CT-space

collisions with other group leaders as well as the obstacles in the environment. The paths of these group leaders (denoted by τ_i) have been determined as a function of time t. We further extend these paths to infinite time by assuming that an avatar will stay at the last configuration of its path when the path is finished. This extended path is denoted by τ_i^*. In order to account for these constraints, we search for a collision-free path in a so-called *Configuration-Time space* (*CT-space*), which is formed by augmenting the C-space with the time dimension. A legal path τ_k in the CT-space is one that does not overlap with the environmental obstacles (*CB*) and τ_i^* (for i = 1 to n and $i \neq k$) as shown in Figure 2. Each τ_i^* induces a time-dependent obstacle to the virtual avatar under consideration. The objective of the path planner is to find a collision-free path in the CT-space that can connect the current and the goal configurations. For realistic simulation, the velocity of a virtual avatar must be within some reasonable limit; therefore, the slope of any point along a legal path in this CT-space must also be positive (time is not reversible) and less than some user-specified value.

With the constraints mentioned above, the search in the CT-space is conducted in a best-first fashion based on the value of each configuration in an artificial potential field. This type of potential field is widely accepted as a good heuristic for motion-planning problems[2]. For efficiency consideration, we only construct a 2D potential field accounting for static environmental obstacles. The best-first algorithm returns a legal collision-free path when the search succeeds and gives up when all possible configurations have been visited. Note that a path is legal only if it can remain collision-free for the whole period when all other avatars are active. Therefore, we require that a goal configuration in the CT-space must have a time value that is equal to or greater than the latest finish time of all other virtual avatars. For instance, in Figure 2, t_k^f (the final time for τ_k)

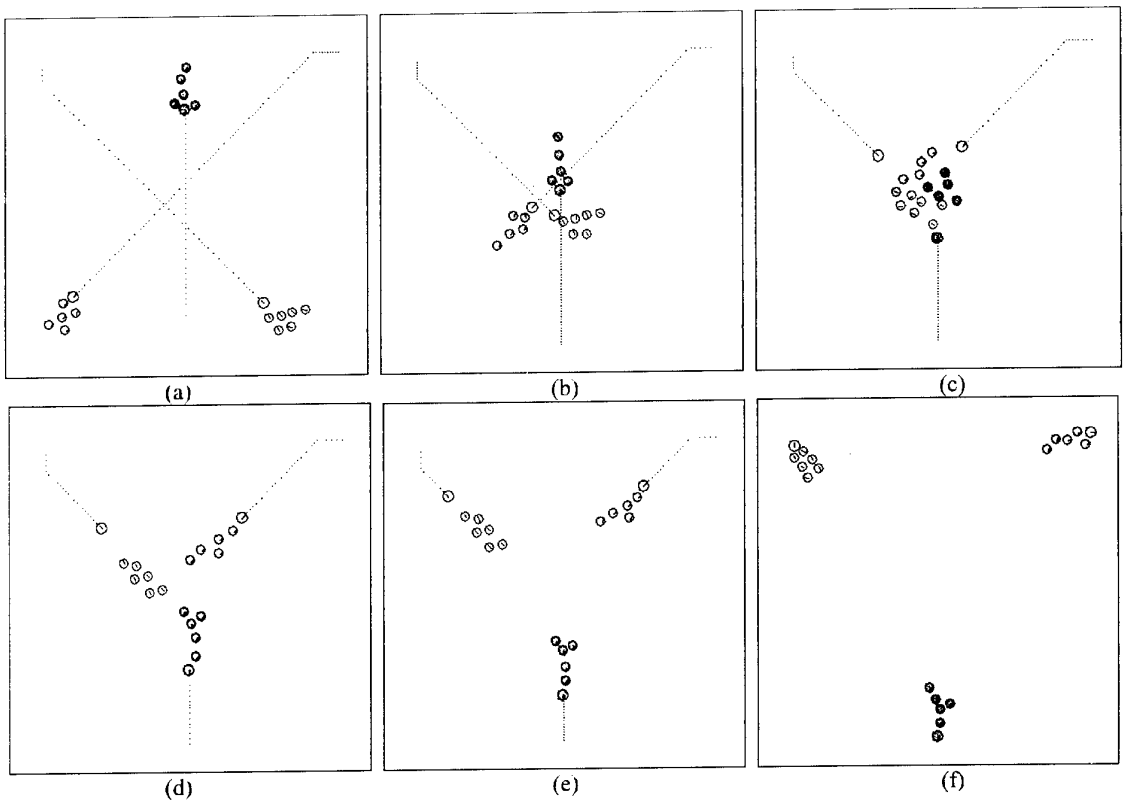


Figure 3. Snapshots showing three crossing virtual crowds

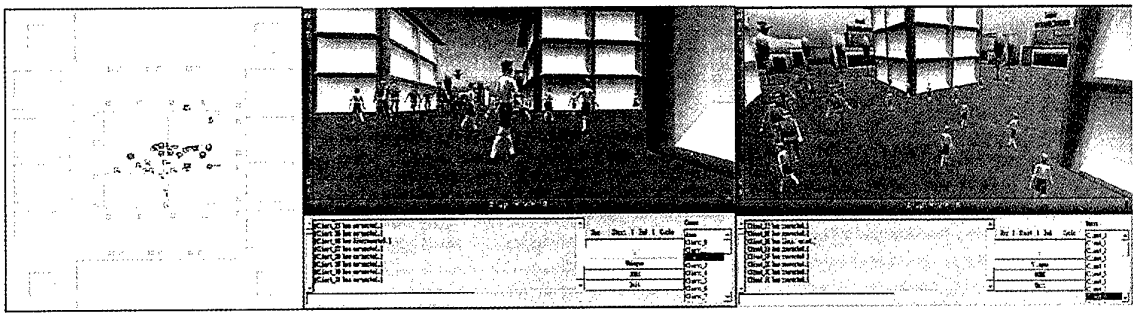
must be greater than t_i^f (which is greater than t_{i+1}^f).

4.4. Emergent behavior for followers

Human avatars' crowd motions are similar to other animals'. However, human avatars in a virtual crowd simulation may possess characteristics unique to human beings. For example, unlike other animals' flocking behavior where the leader-follower relation is formed automatically, a leader acting as a tour guide in a human group is specified explicitly. With the collision-free motions for the leaders generated in the previous subsection, we now describe how we generate the motions for the followers.

To simulate the human grouping behavior, we adopt a strategy similar to the one proposed in [14]. The strategy uses various attractive and repulsive forces to generate steering behaviors. In each control cycle of the simulation, an avatar perceives other avatars in its limited view cone and reacts by adjusting its velocity according to the composite force resulting from various steering and environmental criteria. For example, three steering forces (*separation*, *cohesion*, and *alignment*) were suggested to determine how an avatar reacts to other avatars in its local neighborhood. Separation force is computed according to the repulsive forces exerted by all of its nearby avatars within the view cone. Cohesion is computed by applying an attractive force from the average position of its neighbor avatars in the same group. Alignment is computed by averaging together the velocity of the nearby avatars of the same group. Note that only avatars in the same group exert the cohesion and alignment forces to each other while avatars in different groups can still affect each other with the separation forces. In addition to these three forces, we also apply a repulsive force to an avatar according to its distances from the nearby environmental obstacles. Furthermore, the leader of a group also applies a major attractive force to its followers. This attractive force, proportional to the distance from the leader, drives the followers to the leader even if the leader is not moving.

These five forces altogether are normalized and then re-weighted before they are composed. The weight of each force is dynamically adjustable according to the current world status and the past history. For example, if the force causes a follower avatar to collide with an obstacle, the weight of the repulsive force from the obstacle will be increased. When the collision disappears, the weight for this force will incrementally go down to its nominal value. However, a follower still may bump into obstacles because the repulsive and attractive forces cancel each other. In this case, a sliding force along the obstacle boundary is applied to pull the followers toward the leader.



(a) (b) (c)

Figure 4. Graphical user interface of the shared virtual environment system with virtual crowds

4.5. Examples

In Figure 3, we show an example of three virtual crowds interactively directed by a world manager to their respective goal. Environmental obstacles are not included in this example for clarity. The dashed lines in each figure show the remaining paths to be executed by the leaders. Although the traces appear to be overlapping with each other, the leaders do not collide with each other as the time advances. The virtual crowds start as three separated groups in Figure 3(a). They approach each other in Figure 3(b) when the leaders follow their paths to their goals. These three groups appear to be mixed in Figure 3(c) although the followers in each group are still following their leader. The three groups are separated again in Figure 3(d) although the followers are somewhat behind their leaders due to the delays resulting from the conflicts of performing group crossing. In Figure 3(e), the followers catch up their respective leaders (we allow the followers to move faster than the leaders). The leaders altogether with its followers finally reach their goals in Figure 3(f).

5. Implementation and Experiments

The aforementioned software modules have been fully implemented in Java based on the VNet shared VE system. A virtual world in VRML is created together with its 2D-layout map. This 2D map is an input to the path-planning module for simulating the virtual crowds. In Figure 4 we show sample screen dumps of this system with its 2D and 3D graphical user interface. The 2D interface is presented by a Java program at the server side for interactive crowd controls while the 3D interface is a VRML browser controlled by a Java applet that appears at the client machine.

In the path-planning module, the world is represented by a grid of 128x128. The same resolution is used in the CT-space for searching a feasible path. The time for planning the motion of each group leader is usually only fractions of a second. For example, in the example of Figure 3, the planning time for the path found for each leader in the three groups is 20ms and the computation time for the follower motions during the simulation are negligible. Therefore this kind of performance make the system well suited for our interactive directing purpose.

In our system, we have to ensure that the simulated virtual avatars do not cause any unsafe motions unless they are the intentions of a real avatar. Therefore, during the simulation, we let the leaders, whose motions need to be precisely synchronized, to have higher priorities in each step. The followers will then react according to the leaders' new configurations. However, since the motions of the followers are not planned, there are still no guarantees that they will not be blocking the leaders' ways. Similarly, if a real avatar intends to run into a virtual avatar, collisions are inevitable. Therefore, we perform collision checks for the leaders executing their paths at run time. Whenever a collision will occur in the next step according to the scheduled motion, the path will be cancelled, and the planner will be called again with the original goal and the latest world status.

Although the planner is capable of detecting unexpected collisions and replan accordingly, there exist situations where the path planner may fail to find solutions for the leaders to make moves. These solutions could actually exist but the planner fails to find one because of its incomplete nature of using a decoupled approach. However, in our experiments, this situation rarely happens unless we intend to test it on a pathological case. On the other hand, it is more often that a follower gets stuck at some location minimum of the composite steering force field. We think this situation is similar to the local minimum problem in potential-field-based motion-planning method. Although it is possible to construct local-minimum-free potential fields, it is too consuming to be used for on-line purpose. We are in the process of experimenting with other force fields that account for environmental obstacles in order to improve the situation.

6. Conclusions

In conclusion, we have proposed an interactive system for directing virtual crowds in real-time. The virtual avatars in a shared virtual environment can be controlled with high-level inputs via a graphical user interface. The

system is capable of generating collision-free motions with flocking behavior in an avatar group. The planning capability and efficiency have been successfully demonstrated in a public-domain shared virtual environment system. We are also incorporating the planner into the ActiveWorlds VE system in order to simulate autonomous virtual crowds in a 3D shopping mall applications.

Acknowledgement

This work was partially supported by grants from the National Science Council, ROC, under contracts NSC89-2218-E-004-008 and NSC89-2218-E-004-009.

References

[1] ActiveWorlds, URL: http://www.activeworlds.com.

[2] J. Barraquand and J. Latombe, "Robot Motion Planning: A Distributed Representation Approach," *International Journal of Robotics Research*, 10:628-649 (1991).

[3] Blaxxun Community Server, URL: http://www.blaxxun.com.

[4] T.K. Capin, I.S. Pandzic, N. Magnenat-Thalmann, D. Thalmann, "Integration of Avatars and Autonomous Virtual Humans in: Networked Virtual Environments", *Proceedings of. ISCIS 98, IOS Press*, pp. 326 – 333, Amsterdam, Netherlands, (1998).

[5] T. Capin, I. Pandzic, N. Magnenat Thalmann, D. Thalmann, "Avatars in Networked Virtual Environments", John Wiley & Sons, (1999).

[6] M. Erdmann and T. Lozano-Perez, "*On Multiple Moving Objects*," AI Memo No. 883, Artificial Intelligence Laboratory, MIT, (1986).

[7] J. Funge, X. Tu, and D. Terzopoulos, "Cognitive Modeling: Knowledge, Reasoning, and Planning for Intelligent Characters," *Proceedings of ACM SIGGRAPH*, pp29-38 (1999).

[8] Y. Koga, K. Kondo, J. Kuffner and J.-C. Latombe, "Planning motions with intentions,", *Proceedings of ACM SIGGRAPH'94*, pp 395-408 (1994).

[9] T.Y. Li and J.C. Latombe, "Online Manipulation Planning for Two Robot Arms in a Dynamic Environment," *International Journal of Robotics Research*, 16(2):144-167, (1997).

[10] T.Y. Li, J.M. Lien, S.Y. Chiu, and T.H. Yu, "Automatically Generating Virtual Guided Tours," in *Proceedings of the Computer Animation '99 Conference*, Geneva, Switzerland, pp99-106, (1999).

[11] S.R. Musse, F. Garat, D.Thalmann, "Guiding and Interacting with Virtual Crowds in Real-time," *Proceeding of Eurographics Workshop on Animation and Simulation '99 (CAS '99)*, pp.23-34, Milan, Italy, Springer, (1999).

[12] I.S. Pandzic, T.S. Capin, E. Lee, N. Magnenat-Thalamann, D. Thalmann "Autonomous Actors in Networked Collaborative Virtual Environments", *Proceedings of MultiMedia Modeling '98*, pp. 138-145, IEEE Computer Society Press, (1998).

[13] C. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model, *Proceedings of ACM SIGGRAPH'87*, pp.25-34, (1987).

[14] C. Reynolds, "Steering Behaviors For Autonomous Characters," *Proceedings of Game Developers Conference*, (1999).

[15] X. Tu and D. Terzopoulos, "Artificial Fishes: Physicas, Locomotion, Perception, Behavior," *Proceedings of ACM SIGGRAPH'94*, (1994).

[16] VRML97 International Standard, URL: http://www.web3d.org/technicalinfo/specifications/vrml97/index.htm

[17] VNET, URL: http://www.csclub.uwaterloo.ca/u/sfwhite/vnet.

Effects of viewing angle on performance of wayfinding and cognitive-map acquisition

Masao Ohmi
Kanazawa Institute of Technology
3-1 Yakkaho, Matto, Ishikawa 924-0838 JAPAN
ohmi@mattolab.kanazawa-it.ac.jp

Abstract

Effects of different viewing angle, namely an oblique-angle view and a straight-angle view, on performance of wayfinding and acquisition of cognitive map was investigated in virtual environment. It was found that performance of wayfinding is significantly better with oblique-angle viewing condition than with straight-angle viewing condition. On the contrary, quality of acquired cognitive map was significantly better with straight-angle viewing. It suggests that contents of visual field during exploration has different effect on two aspects of environmental learning.

Key words: Wayfinding, Cognitive map, Oblique-view display, In-vehicle route guidance system

1. Introduction

When we find a way in an environment, we use both an egocentric information and an exocentric information. The egocentric information for wayfinding is a real-time change of sight from our own point of view. The exocentric information for wayfinding is a mental representation of the environment. The exocentric representation is also entitled as a cognitive map, since it is supposed to be like a map of the environment.

It is believed that we learn the exocentric representation of an environment by integrating egocentric views that are perceived while exploring the environment (Weisman, 1981; Passini and Proulx, 1988). A Landmark-Route-Survey Map model (LRS model) is the most well-accepted model for describing how the exocentric representation of environment is acquired from the egocentric information (Siegel and White, 1975; Thorndyke and Hayes-Roth, 1982).

When we encounter an unfamiliar environment at the first time, we acquire descriptive information about a few landmarks (Landmark stage). Then, by using these landmarks as markers, we develop information about specific route (Route stage). This information is a set of paths and turns to reach a specific destination. Finally, we learn cognitive map, or survey map, of the environment and are able to take, for example, a short cut easily (Survey Map stage). Therefore, final understanding of a real-world environment is achieved by acquisition of cognitive map of the environment. It has been assumed that these representations are acquired successively as we have more experience in the environment.

In the real world, however, we not only develop the exocentric representation of the environment from the egocentric views through exploration, but also have access to plenty of artificial information such as a road map and a in-vehicle route guidance system. Therefore it is practically more important to understand effects of these artificial information on wayfinding and acquisition of cognitive map.

A semi-bird's eye display with oblique-angle viewing becomes so popular in the in-vehicle route guidance system. The oblique-view display has been introduced to facilitate wayfinding in unfamiliar environment. However, effects of the oblique-view display on acquisition of cognitive map have not been thoroughly investigated. Since the oblique-view display makes wayfinding easier, it could disrupt learning process of cognitive map. In this research, we investigated effects of viewing angle during exploration of the environment on performance of wayfinding and on acquisition of cognitive map in virtual environment.

2. Experiment

Method

A real-world environment was simulated by a maze with a hexagonal layout in a virtual environment. Landmark illustrated by different color or numeral was placed at each corner of intersections in the maze. There were two sizes of virtual maze. The small maze, of which example is shown in Figure 1, had five intersections. The large maze had seven intersections.

The egocentric view of observers was transformed according to their location in the maze. There were two viewing conditions: straight-angle viewing and oblique-angle viewing conditions. Egocentric view with straight viewing angle at an intersection is depicted in Figure 2.

Only landmarks at the immediate intersection were visible. Egocentric view with oblique viewing angle at an intersection is depicted in Figure 3. Landmarks at other intersections as well as at the immediate intersection were visible.

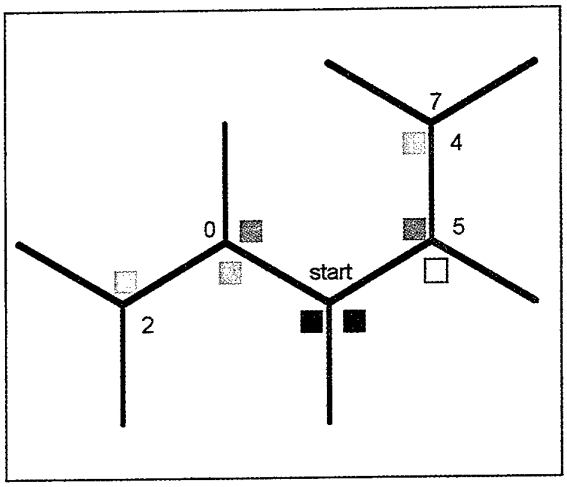


Figure 1 Top view of the virtual maze

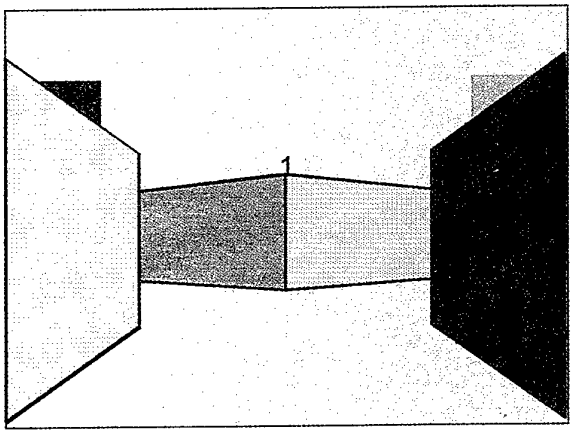


Figure 2 Egocentric view of the virtual maze with straight viewing angle

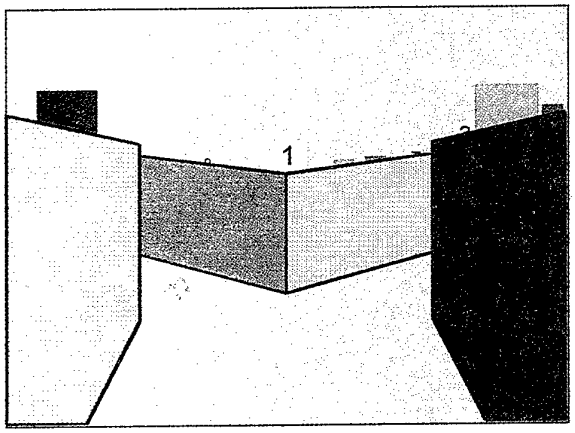


Figure 3 Observer's egocentric view of the virtual maze with oblique viewing angle

The virtual maze was created by a real-time graphical simulation application (WalkThrough Pro, Virtus) on a personal computer and was presented on a 21" computer monitor.

Procedure

Observers explored the maze consecutively by following instruction of the experimenter who verbally directed color or symbol of a landmark to which observers found a way to reach. Immediately after observers reached the landmark, color or symbol of next landmark was instructed. Time to spend for wayfinding from the start to the final landmark was measured.

After observers reached the final landmark, they were asked to draw cognitive map of the virtual maze that they acquired during exploration. The quality of drawn cognitive map was assessed by number of correct landmarks and streets in the drawn map.

Five kinds of maze with the same shape and with different disposition of landmarks were presented. There were twenty trials, namely two sizes of maze, two viewing conditions and five kinds of maze. Sequence of twenty trials was in random order. Fourteen undergraduate students participated as an observer.

3. Results

Time to spend for wayfinding from the start to the final landmark did not vary systematically for five mazes with the same size and viewing condition. Since there was no significant difference among observers neither, data were averaged among observers for each condition.

Averaged time to spend for wayfinding are shown with standard error in Figure 4. Each column shows value for two sizes of maze and for two kinds of viewing angle. Time for wayfinding was shorter with oblique-angle viewing than with straight- angle viewing for both sizes of maze.

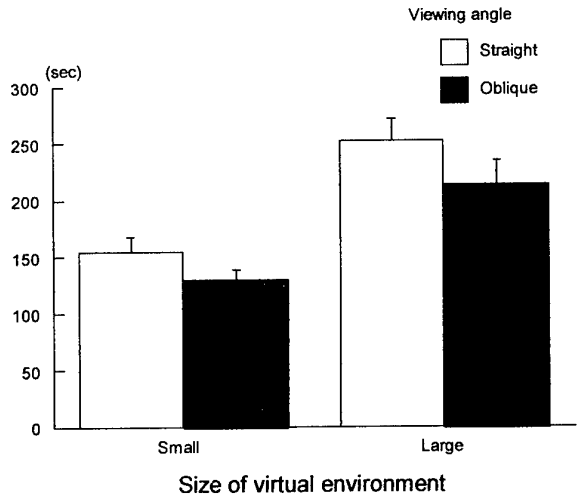


Figure 4 Averaged time to spend for wayfinding

There were statistically significant effects of viewing angle and of size of maze on time to spend for wayfinding. Since it is trivial to take more time for wayfinding in larger maze, the result means that performance of wayfinding is significantly better with oblique-angle viewing condition than with straight-angle viewing condition for both sizes of virtual maze.

Percent correct of cognitive map drawn by observers did not vary systematically for five mazes with the same condition. Since there was no significant difference among observers neither, data were averaged among observers for each condition.

Averaged percent correct of cognitive map are shown with standard error in Figure 5. Each column shows value for two sizes of maze and for two kinds of viewing angle. The quality of acquired cognitive map was better with straight-angle viewing than with oblique-angle viewing for both sizes of maze.

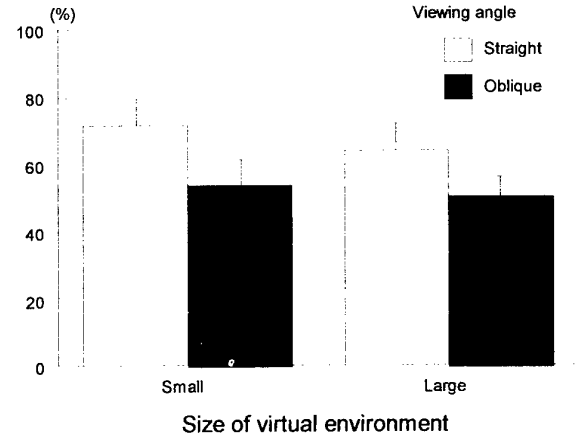


Figure 5 Averaged percent correct of cognitive map

There was statistically significant effect of viewing angle on percent correct of acquired cognitive map. The result means that, on the contrary to performance of wayfinding, observers acquired significantly better cognitive map of virtual maze with straight-angle viewing condition than with oblique-angle viewing condition for both sizes of virtual environment.

Since it took more time for wayfinding with straight-angle viewing condition, it might be argued that improvement of quality of cognitive map was due to longer exploring time. However, quality of each cognitive map drawn by observer after seventy trials with longer exploration time was not significantly different from that drawn after seventy trials with shorter exploration for both small and large mazes. Therefore, it is more likely that better cognitive map is acquired by exploring environment with straight-angle viewing rather than by spending more time in environment.

4. Discussion

It is concluded from the results that contents of visual field to be presented during exploration of new environment has different effect on two indispensable aspects of environmental understanding. It is rather surprising that acquisition of cognitive map is not improved by presenting more visual contents by oblique-angle viewing, even though performance of wayfinding is facilitated. Gillner and Mallot (1998) reported similar effects of amount of visual contents on acquisition of cognitive map. Their account that the amount of knowledge acquired was determined not by its availability but by the different needs in the task would explain our unexpected results.

On the other hand, the process for developing exocentric representation of environment from temporally changing egocentric information requires information about observer's orientation. Orientation could be obtained through sensation of self-motion (Asakura, Ohmi & Suzuki, 1999) or by information about observer's relation with landmarks and heading (Ohmi, 1999). Less contents in straight-angle viewing condition would force observer to remember spatial relationship among intersections of maze more carefully. Therefore, sense of orientation could be facilitated more in straight-angle viewing condition.

It has been reported that people can be grouped by a preference of environmental representation (Ohmi, 1998). Almost half of people prefer a route representation and memorize environment as a set of paths and turns from a start to a destination. Other half of people prefer a survey map representation and memorize environment as a map. Although it was reported the performance of wayfinding was not significantly different for both groups, results of this

research suggest that acquisition of cognitive map could be different between these two groups.

In order to investigate individual differences among observers, they were asked after experiment which viewing angle they preferred for wayfinding task and for cognitive map task. Not surprisingly, all observers reported that they preferred oblique-angle view for wayfinding task. On the other hand, for cognitive map task nine observers reported that they preferred straight-angle view and five observers reported they preferred oblique-angle view.

Left panel of Figure 6 depicts averaged percent correct of cognitive map for observers who preferred straight-angle viewing. It shows that these observers acquired better cognitive map by exploring environment with straight-angle viewing. The difference of percent correct of cognitive map between two viewing conditions was statistically significant.

Right panel of Figure 6 depicts averaged percent correct of cognitive map for observers who preferred oblique-angle viewing. On the contrary to their preference, quality of acquired cognitive map was similar for both viewing conditions. It means that there is no advantage of oblique-angle viewing on learning of cognitive map even if observer claim that they prefer it.

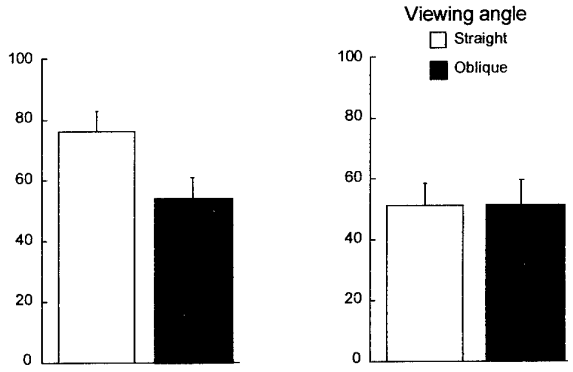


Figure 6 Averaged percent correct of cognitive map for two groups of observer

Our results suggest that brain mechanism for finding a way in environment is distinct from that for acquiring cognitive map of environment. For more practical point of view, presenting semi-bird's eye view in in-vehicle route guidance system is not necessary a good idea, because it would disturb acquisition of cognitive map of environment, which is essential for civilized life.

5. Conclusion

1. Performance of wayfinding is facilitated by presenting egocentric view with oblique viewing angle.

2. Acquisition of cognitive map is disrupted by presenting oblique-angle view during exploration.

3. It would suggest that brain mechanism for finding a way in environment is distinct from that for acquiring cognitive map of environment.

4. Semi-bird's eye display in the in-vehicle route guidance system is not necessary a good idea since it would disrupt acquisition of cognitive map of environment, which is essential for civilized life.

References

Asakura N, Ohmi M and Suzuki R (1999) The effect of self-movement information on orientation dependency in human spatial memory. Proceedings of Conference on Human Interface Processing, Sept 99, 1-8 (in Japanese)

Gillner S and Mallot HA (1998) Navigation and acquisition of spatial knowledge in a virtual maze. Journal of Cognitive Neuroscience, **10**, 445-463

Ohmi M (1998) How egocentric and exocentric information are used to find a way in virtual environment? Proceedings of the Eighth International Conference on Artificial Reality and Tele-Existence, 196-201

Ohmi M (1999) Roles of additional information for wayfinding in virtual environment. Proceedings of the Eighth International Conference on Artificial Reality and Tele-Existence, 189-193

Passini R and Proulx G (1988) Wayfinding without vision. Environment and Behavior, **20**, 227-252

Siegel AW and White SH (1975) The development of spatial representations of large-scale environment. In HW Reese (Ed.) Advances in Child Development and Behavior. (pp. 9-55) Academic Press, New York

Thorndyke PW and Hayes-Roth B (1982) Differences in spatial knowledge acquired from maps and navigation. Cognitive Psychology, **14**, 560-589

Weisman J (1981) Evaluating architectural legibility: Way-finding in the built environment. Environment and Behavior, **13**, 189-204

Implementation and Evaluation of GIS using IPT

Kuang Li Chen, Tuck Seng Kong, Jun Kukimoto,
Noriyuki Kitajima, Byungdug Jun* and Takashi Takeda
Nagasaki Institute of Applied Science, Japan,
536, Aba-machi, Nagasaki-shi, 851-0193, JAPAN
*PECK Ltd., Japan
7-1, Shimonishiyama-machi, Nagasaki-shi, 850-0004, JAPAN
takeda@csce.nias.ac.jp

Abstract

We will show the design of the new system utilizing GIS (Geographic Information Systems). 3D-geographic information of Nagasaki-shi was constructed from tracing the aerial photograph and adding the height information of measured buildings. In large multi-plane stereo presentation equipment called IPT (Immersive Projection Technology), this 3D-map was presented in actual scale. As the result, the observer could obtain three-dimensional information of various lands as direct experience, in spite of there being the self in the IPT inside. Furthermore, we set a treadmill, an ambulation device, in the inside of the IPT. When it synchronized to the walking of the observer and scene change, it seemed to obtain the more accurate spatial information. Finally, we conducted an experiment for evaluation of this system. If the virtual town in the system is similar to the actual one, virtual experience come useful for real life.

Key words: IPT, GIS, 3D-map, VR, VRML, Evaluation

1. Introduction

Recently, the movement that intends to utilize cooperated GIS database is activated in the Japanese government. In the local government of each prefecture, peculiar administration support model using the various GIS database is proposed. In the Seihi town in Nagasaki Prefecture, the welfare support system by GIS technique is made. It is a purpose to attempt database preparation and reduction in the cost that depends on the maintenance for the each every post of the administration. It is not fixed only in improving the efficiency improvement of the administration business according to the GIS database. Constructed database spreads to various utilization fields by adding the GIS technology. This study is that it adds 3D image reproduction technology and virtual reality technology to the GIS database and shows the possibility of utilizing as townscape simulation information.

2. Present state of the townscape simulation

Until now, it is the mainstream that the simulation makes landscape model of the object area in respect of the townscape. For the landscape simulation of the gcod accuracy, the building feature is stuck in the model. The partial correction is difficult for this model, and it is easy to deteriorate, and permanent preservation is not possible. In such work of the labor, this model needs very much large cost and work period. It is also simple to correct the model, if this model is realized on the computer, and permanent preservation is also possible. And, the simulation is possible always, when they are necessary. For the Tateyama district in Nagasaki, the computer simulation model (the GIS database) was made. And, this model was carried out in virtual reality expression equipment (IPT).

3. The research area

Research region of this study chose representative hillside area (the Tateyama district) in the Nagasaki City. The Tateyama district is a optimum region of the object of the modeling. Hillside area of the Nagasaki City was formed in the high growth period after 1960's. There was no correspondence performance on hillside area of Nagasaki in the movement of the motorization in 1980's. Therefore, hillside area became a region where decrease of the young generation and hollowing of the population and aging advanced. The simulation for the regional activation is a necessary region on hillside area from the viewpoint of community planning. In this study, for the reason of the superscription, Tateyama district (zone from the Tateyama 1 chome to 5) in the Nagasaki City was digitized as a map of the 1/2500 accuracy. 3D image of the Tateyama district was made using this digital map. And, the feature of the building was stuck in the 3D image. Finally, the townscape simulation was carried out on IPT for the Tateyama district.

4. The GIS data improvement

4.1 The GIS basis data

There are so many GIS data in Japan. The digital map in Japan issues all 21 types (National Land Agency: 8 type Geographical Survey Inst.: 13 type). In this inside, numerical value map of 2500 (spatial databases) has the

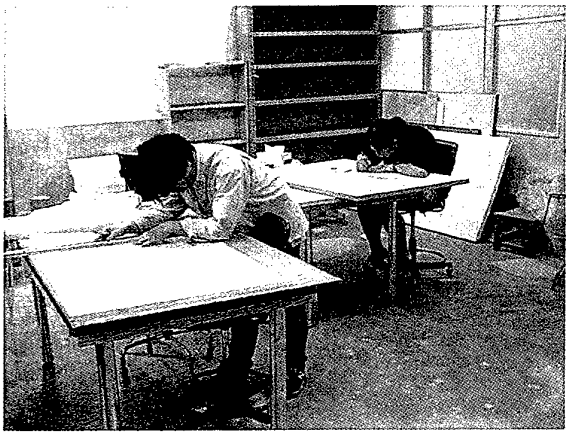


Fig. 1 Tracing work of digital map

Fig. 2 Ended sheet of tracing work

from the city planning map (the Tateyama district map (No.62, No.72): 1/2500 accuracy) of 2 sheets.

4.2 From the city planning map of 2 sheets to the numerical value map

4.2.1 Graphical data

From the city-planning map of 2 sheets, digital elevation model (DEM) of topographic data (point information measured with the contour line) was made. Spatial data made building information, administration field zone information, road and stair information, railway and train route information, coastline information as layer information. Figure.1 is tracing work photograph. And, Figure.2 is the ended sheet of tracing work.

4.2.2 DEM Data

DEM Data of this study is the triangulation irregular network (TIN) model. TIN makes the triangle group arise from landform point placing in the random state, and it is a kind of making DEM. The reason for choosing TIN in this study as a DEM model is because the model of which the generation of DEM is possible at the good accuracy from contour line and measure point (No.62 are 440 points, 320 points, sum total of 760 points No.72). However, the tracing work was carried out in this study at the 10m unit. Vector data of the contour line is converted as line information, and altitude data of the contour line is input into each line information as attribution information. Measurement point placed in the random state becomes the very effective data in order to make TIN. However, the detailed DEM preparation is difficult in the mountainous area, because the measurement point is little. And, there is a weak point in which the expression near the summit is scarce in the case of TIN according to contour line. In this study, TIN (Figure.3) got from contour line of mountainous area and TIN (Figure.4) got from measurement point of the random state are superimposed, and TIN (Figure.5) of the detailed Tateyama district make. Figure.6 extracted administration field zone from the Tateyama 1-chome to sufficient utilization accuracy as a GIS data. However, there is no this data in present, August 2000, Nagasaki City. Then, the digital numerical value map was made

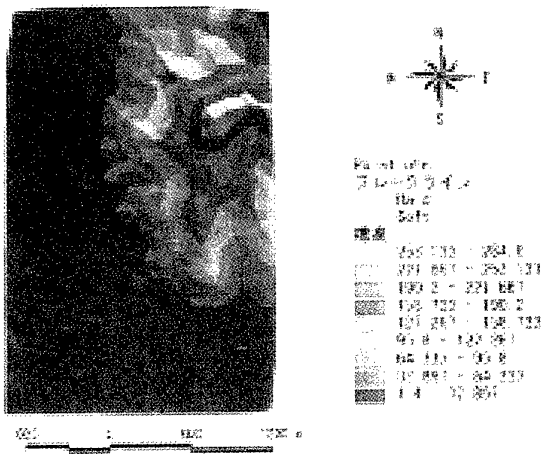


Fig. 3 TIN made by measure points

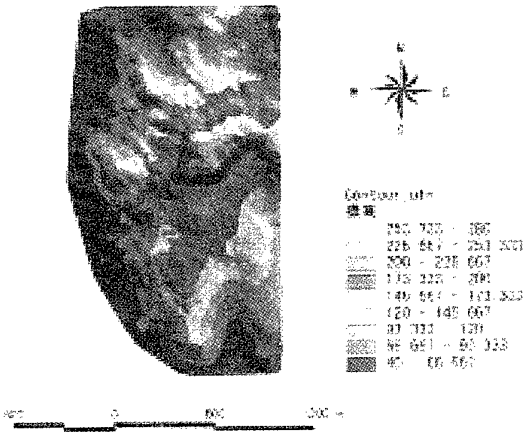


Fig. 4 TIN made by 10m contour

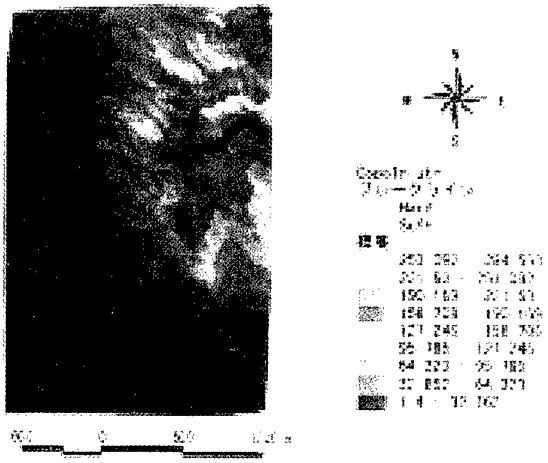


Fig. 5 TIN made by points and contour

and coastline information as a polygon data. And, the following are input as a line data: Road and stair and

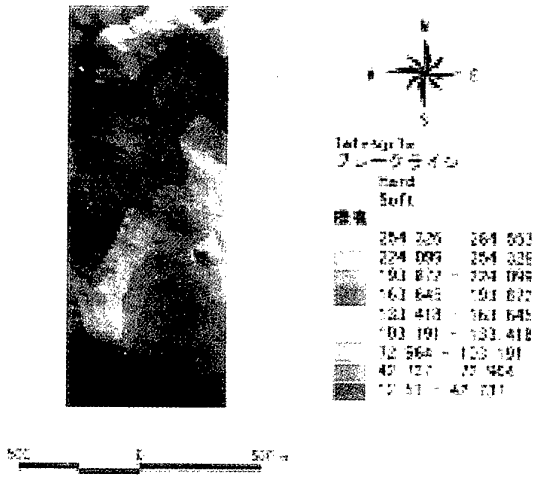


Fig. 6 Finished TIN in this study

railway and train route of spatial data. Figure.7 is spatial data that show building and road. The attribution data of the building input classification (wooden construction and non-wooden construction and concrete, etc.), nameplate and building application (it is classified into 22 types such as housing, store, public facility) with the rank. And, vacant land and plowed field, parks and planned road and plan parks, etc. are added information. *4.2.4 3D image (VRML)*

The 3D-preparation image was made to be VRML format that could grasp the whole town. Figure.8 is 3D image that converted as a VRML format. By adding the feature, the image on IPT used it.

5. Simulation of Urban View in IPT

Observers can experience virtual world by Immersive

5-chome.

4.2.3 Spatial data

Spatial data input administration field zone and building

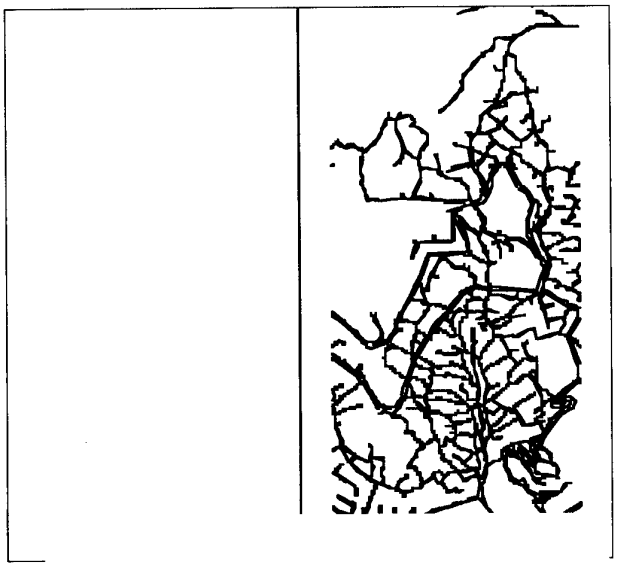


Fig. 8 VRML image of Tateyama region

Projection Technology (IPT), which is constituted of multi wide-screens and stereo system utilizing liquid-crystal-shutter or polarized plastic framed glasses. In this study to simulate urban views, we used IPT having front, both sides and floor screens. Graphics work station (Onyx2) stored VRML files and projected them using Performer library. Figure.9 shows the simulation of whole town. As original maps have cross-sections of buildings with height information only, all buildings seem like simple boxes. To make more reality, we will have to add roofs of Japanese houses and so on. Figure.10 shows different scale views. Observers can see the town from aerial view and enter the same scale town as real world. Building's windows and entrances were obtained by texture mapping using digital photograph taken at the places. The advantages of simulation GIS in IPT are that observers can see views from various angles and change scale size as if they are in the town. Furthermore, if we set a treadmill in the inside of IPT

and the rotation speed of the belt synchronizes images, observers can experience walking sensation in the town (see Figure.11). In addition, our treadmill can change the slope of the belt, and then some sensation of going up an ascent can be obtained too.

6. Evaluation of GIS in IPT

Next, we conducted an experiment for evaluation of GIS in IPT. If the virtual town in the system is similar to the actual one, virtual experience will come useful for real life. Here, we observed the walking performance to go to a goal point in real world with learning the route in IPT.

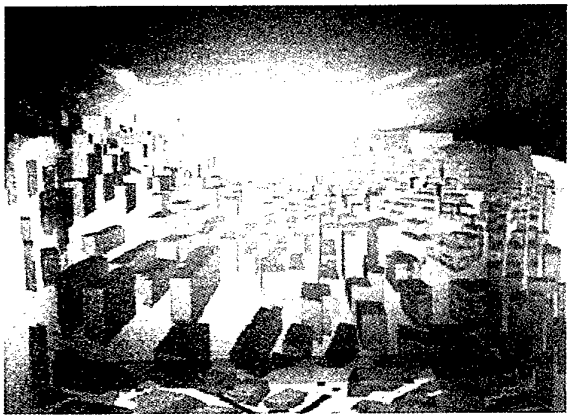


Fig. 9 Whole town in IPT

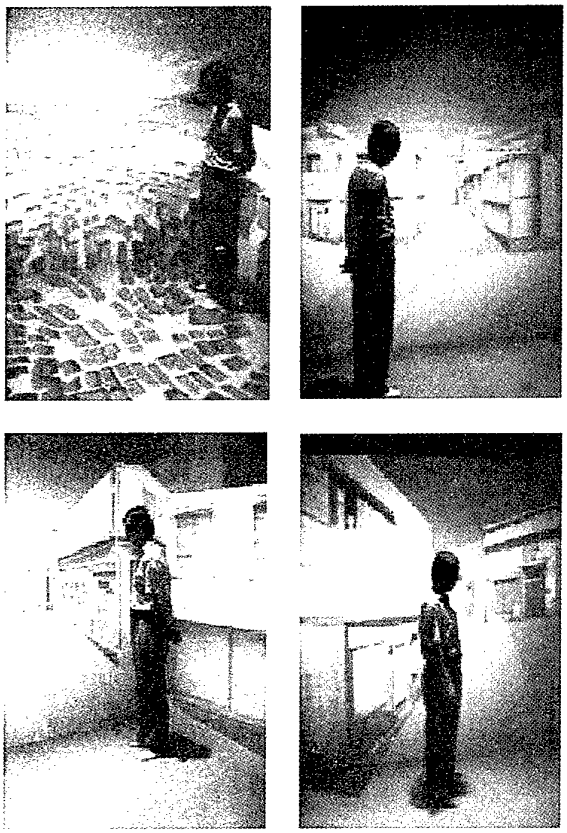


Fig. 10 Various scale views from various angle

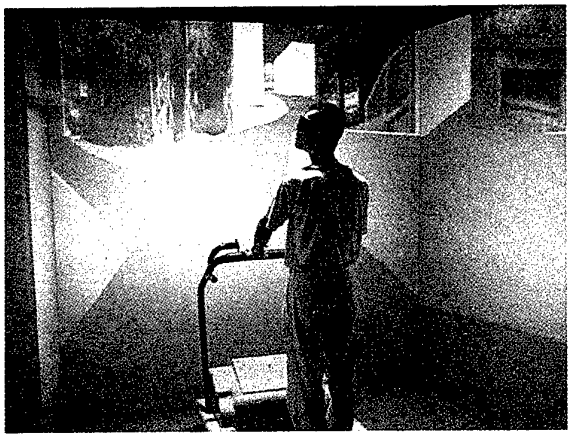


Fig. 11 Walking with treadmill in IPT

6.1 Method

Figure.12 shows the route, in Tateyama 2-Choume, which subjects walked from the start point to the goal point. Ten subjects who haven't been to the route were divided into following two groups of five subjects (two females and three males).

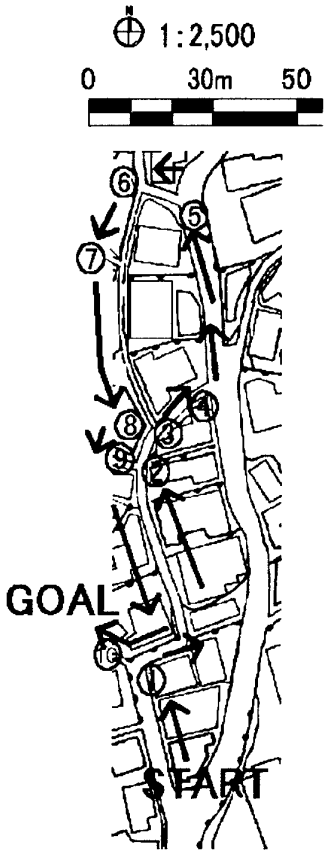
IPT group (IG): Subjects learned the route in IPT before walking in real world. They stood in the center of IPT, and the real-scale image was flowed at 3 km/hour, as if walking the route. The building's textures were obtained at the place. Learning the route was repeatedly until they could memorize the route. The interval of learning repetitions was 7 min. In order to check extent of subject's memory, a red ball, which had 1 m radius, was presented at the parting of the route for about 4 seconds just before bending. Subjects had to report which road they should take during the ball being with the except of the first leaning. At the first learning, they were instructed to ignore the ball. When all bending were correct, learning term terminated. After that, subjects must report the sequence of which bending each parting from the first corner (e.g. right, right, left, ... or left, right, left... etc.). There was possibility of judging the route by merely sequence of right and left, despite we asked them to memorize the route utilizing virtual scenes before learning. This result was used to check subject's strategy. About 1 hour later, they arrived at the start point in Tateyama 2-Choume by car, and were asked to walk the route to go to the goal point. If subjects took a wrong course, the experimenter taught them real one. We recorded subject's behavior on video.

IPT with treadmill group (ITG): Almost all conditions were same as IG's ones, except for using the treadmill, which was synchronized with the image, in IPT. One might suppose that walking is useful for memorizing distance from the corner to the next one. If walking in learning the route is important factor, ITG will perform better than IG.

6.2 Results

6.2.1 Learning in IPT

Since the learning implies training for subjects to memorize the route, the number of learning repetition



doesn't include last repetition at which they could decide all bending correctly. Figure 13 shows the average of

Fig.12 The route in this experiment

learning repetitions across all subjects for each group; error bars represent standard deviation. There is significant difference between IG (1.6 repetitions) and ITG (2.6 repetitions) ($t(8) = 2.89$, $p < .05$). The right column of Table.1 gives the corners that each subject decided incorrectly in the learning. Incorrect decisions were concentrated at corner 9 and 10.

6.2.2 Walking in real world

Figure.14 shows a walking appearance of a subject. Figure.15 shows the average of incorrectly decided corners across all subjects for each group. There is no difference between IG and ITG ($t(8) = 0$, $p>.1$) and the averages are 1.2 times in both group. If subjects decided the route at random, ie ineffective learning in IPT, the number would be five (chance level). The left column of Table.1 gives the corners that subjects decided incorrectly in the route. Incorrect decisions were concentrated at corner 5 and 10.

We observed performance of walking in unknown place with learning in IPT to investigate how useful GIS in

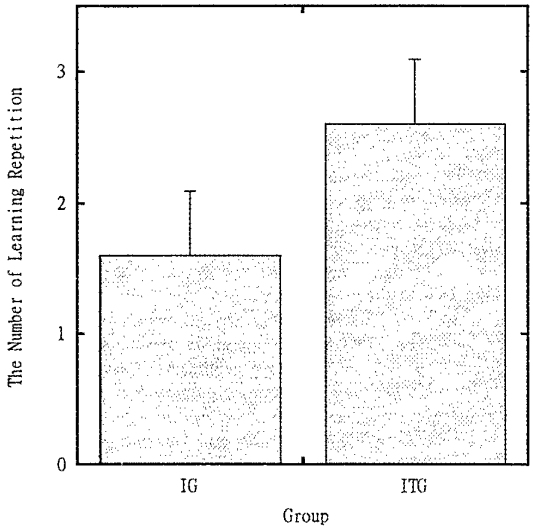


Fig. 13 The learning repetition for each group

Fig. 14 Walking appearance of a subject

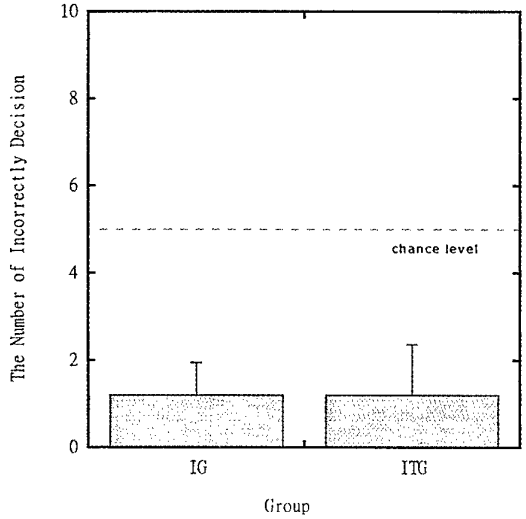


Fig. 15 The number of incorrectly decision for each group

6.3 Discussion

IPT is in real life. When subjects walked in real world

Table 1. Incorrect corner for each subject

Subjects	In Learning Term	In Real World
	IG	
NA	9(2)	7
AK	*None*	*None*
KY	3(2)	10
YO	3(2) , 9(2)	5 , 10
TS	*None*	5 , 10
	ITG	
SY	9(2)	*None*
TO	5(2) , 6(2) , 10(2)	5 , 6 , 10
EU	6(2) , 10(2) , 9(3)	10
SK	1(2) , 5(2) , 10(2) , 10(3)	*None*
HU	5(2) , 3(3)	5 , 10

( )represents learning repetition

after learning the route in IPT, even though the interval between learning in IPT and walking in real world was about 1 hour, incorrect decisions at partings were only 1.2 times much less than the chance level. Thus, this virtual town made by GIS is similar to real one and useful for our life.

6.3.1 Learning with treadmill

Although there was no difference of incorrect decisions in real world between IG and ITG, ITG's subjects needed learning repetitions more than IG's ones to memorize the route perfectly. We expected that walking in IPT would be helpful to learn the route, whereas it made to memorize the route difficult. It is probably due to the complicated task which subjects viewed virtual scene and accompanied with the belt of the treadmill rather than simply viewing the scene. At least, using the treadmill interfered with leaning the route in present experiment. However, if the route was memorized in IPT, it is almost same for both groups that subjects could their the memory to real world.

6.3.2 Incorrect corners

Corner 9 and 10 were frequently mistaken in learning. Since the buildings around corner 9 are especially built-up as a small maze, it might take many times to memorize directions that subjects should take. Once the memory is stored, however, it isn't difficult to choose the route and no one mistook at the corner 9 in real world. At corner 10, the performance was bad in not only IPT but also real world. Mallot & Gillner[1] found that local landmarks are stronger cues than global configurations for route navigation. Around corner 10, there aren't o \(out st ardi rg lui ldtherefore, it might be so difficult to memorize the route and apply their memory to real world. Also the street near corner 10 in IPT is broader than real world' one. Because we simply attached a wall surrounding a house to buildings as texture, the width between buildings didn't change from initial 2D map. It might be cause for subjects to mistake the route at corner 10 in real world. We would like to think details to pursuit more useful virtual system.

6.3.3 Strategy to memorize the route

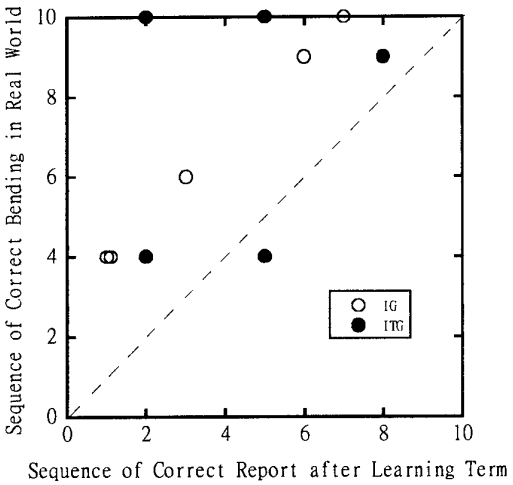


Fig. 16 Comparison sequence of correct report with sequence correct bending

How did subjects memorize the route in IPT? If they simply memorized a sequence of which bending each parting from the first corner (ie a word sequence of "right" and "left"), virtual experience in IPT isn't insignificant very much. Here we compare the sequences of correct report after learning term with the sequence of correct bending from the start point in real world for each subject. The sequence of correct report means upper limit by means of the word sequence. If subjects utilized a word sequence only in real world, in Figure.16, each symbol should plotted under dashed line. However, almost all symbols plotted over the line mean that subjects utilized other cue to take directions at least. Thus, it is probably that real world enhanced memorized virtual scene and the scene navigated the route.

7. Conclusion

It is valuable for simulation of urban views, emergency prevention and navigation that GIS is implemented in IPT using computer technology. Our experiment supported that GIS in IPT is similar to real world and the virtual experience can come useful for our life. However, in present study, using printed maps to realize this system took a lot of time and labor. If 3D information from satellites will be obtained easily in the future, unknown places will appear in IPT very soon.

Reference

1. H. A. Mallot and S Gillner: "Route Navigating without Place Recognition: What is recognized in recognition-triggered responses?," *perception*, pp.43-55 (2000).

Dynamic Analysis for Realistic Motion Simulation in Virtual World

Daisuke Tsubouchi*, Tetsuro Ogi** and Hirohisa Noguchi***

*Department of Mechanical Engineering, Keio University
3-14-1 Hiyoshi, Kouhoku-ku, Yokohama, 223-8522, Japan
tsubo@noguchi.sd.keio.ac.jp
**MVL Research Center, Telecommunications Advancement Organization of Japan
/ IML, Univirsity of Tokyo
***Department of System Design Engineering, Keio University

Abstract

In this study, in order to realize real-time motion simulation in virtual environment, a new efficient time integration scheme for finite element method was proposed. The proposed method is called 'iterative Newmark' method. In this method, it is not necessary to calculate the inverse of coefficient matrix like the explicit time integration scheme and has a stability criterion of the conventional Newmark method. This method was applied to several examples of the motion simulation, and the real-time and realistic motion was realized.

Keywords : Finite Element Method, Dynamics, Real-time Simulation, Virtual Reality

1. Introduction

In order to construct realistic virtual worlds, it is important to simulate the realistic movement of the objects. For instance, the virtual objects should be moved according to the law of motion and be deformed by applied force.

By using the finite element method, the realistic movement and deformation can be simulated. As for the static analysis using the finite element method in virtual world, several studies have been presented, especially in the field of haptic rendering[1]. However it is impossible to simulate the dynamic motions (deformation, translation, rotation etc.) of virtual objects by only using the static analysis. In order to realize these motions, dynamic analysis using the time integration scheme is indispensable. Although several time integration schemes have been introduced for dynamic analysis, these existing schemes are not suitable for virtual reality. Because the virtual reality applications require efficiency and stability for the interactive and real-time simulation, while the scientific analysis gives priority to the accuracy in the calculation.

In this study, a new efficient and stable time integration scheme was proposed to overcome these problems.

2. Methods

2.1. Overview of time integration schemes

In general, two kinds of time integration schemes are utilized for finite element method. One is an explicit time integration scheme such as central difference method, and the other is an implicit time integration scheme such as Newmark-β method. Although the explicit time integration scheme spends low computation cost as it does not need to calculate the inverse matrix, its stability is conditionally guaranteed with a small time increment. On the other hand, the implicit time integration has opposite features.

In this study, a new time integration method based on the Newmark-β method with the advantage of explicit methods was proposed.

2.2. Newmark-β method

The discrete equation of motion is formulated by the following equation[2].

$$\mathbf{M}_t\ddot{\mathbf{u}}_t + \mathbf{C}_t\dot{\mathbf{u}}_t + \mathbf{K}_t\mathbf{u}_t = \mathbf{F}_t \tag{1}$$

where $\mathbf{M}_t$ is the mass matrix, $\mathbf{C}_t$ is the viscous damping matrix, $\mathbf{K}_t$ is the stiffness matrix, $\mathbf{F}_t$ is the applied force vector and $\ddot{\mathbf{u}}_t$, $\dot{\mathbf{u}}_t$ and $\mathbf{u}_t$ are the acceleration, velocity and displacement vectors, respectively. When the elapsed time is $t+\Delta t$, Eq. (1) is represented by Eq. (2).

$$\mathbf{M}_{t+\Delta t}\ddot{\mathbf{u}}_{t+\Delta t} + \mathbf{C}_{t+\Delta t}\dot{\mathbf{u}}_{t+\Delta t} + \mathbf{K}_{t+\Delta t}\mathbf{u}_{t+\Delta t} = \mathbf{F}_{t+\Delta t} \tag{2}$$

When the Newmark-β method is used, $\mathbf{u}_t$ and $\dot{\mathbf{u}}_t$ are approximated as follows:

$$\dot{\mathbf{u}}_{t+\Delta t} = \dot{\mathbf{u}}_t + \frac{\Delta t}{2}\left(\ddot{\mathbf{u}}_t + \ddot{\mathbf{u}}_{t+\Delta t}\right) \tag{3}$$

$$\mathbf{u}_{t+\Delta t} = \mathbf{u}_t + \Delta t\dot{\mathbf{u}}_t + \frac{\Delta t^2}{2}\left\{(1-2\beta)\ddot{\mathbf{u}}_t + 2\beta\ddot{\mathbf{u}}_{t+\Delta t}\right\} \tag{4}$$

Eq. (3) and Eq. (4) are finite difference formulas. The parameter β determines the characteristics of stability and accuracy of this algorithm.

By substituting Eq. (3) and Eq. (4) for $\mathbf{u}_t$ and $\dot{\mathbf{u}}_t$ in Eq. (2), the following equation is obtained

$$\begin{aligned}&\left(\mathbf{M}_{t+\Delta t} + \frac{\Delta t}{2}\mathbf{C}_{t+\Delta t} + \beta\Delta t^2\mathbf{K}_{t+\Delta t}\right)\ddot{\mathbf{u}}_{t+\Delta t}\\ &= -\mathbf{C}_{t+\Delta t}\left(\dot{\mathbf{u}}_t + \frac{\Delta t}{2}\ddot{\mathbf{u}}_t\right)\\ &\quad -\mathbf{K}_{t+\Delta t}\left\{\mathbf{u}_t + \Delta t\dot{\mathbf{u}}_t + \frac{\Delta t}{2}(1-2\beta)\ddot{\mathbf{u}}_t\right\} + \mathbf{F}_{t+\Delta t}\end{aligned} \tag{5}$$

Assuming that $\mathbf{u}_t$, $\dot{\mathbf{u}}_t$ and $\ddot{\mathbf{u}}_t$ are known from the previous step of the calculations, $\ddot{\mathbf{u}}_{t+\Delta t}$ is determined by solving Eq. (5). And the $\dot{\mathbf{u}}_{t+\Delta t}$ and $\mathbf{u}_{t+\Delta t}$ are determined from Eqs. (3)-(4).

However, this method can hardly be applied to virtual reality applications, because it costs much computational time to calculate the inverse of coefficient matrix in Eq. (5).

On the other hand, the Newmark-β method has an advantage of unconditionally stability under the condition of $\beta \geq \frac{1}{4}$. In this study, we propose iterative Newmark method that has a stability criterion equivalent to the conventional Newmark method and does not need to calculate the inverse of coefficient matrix like the explicit time integration scheme.

2.3 Iterative Newmark method

2.3.1 Overview of iterative Newmark method

Moving the term of a stiffness matrix in the left hand side of Eq. (5) to the right hand side, Eq. (5) is rewritten as Eq. (6).

$$\begin{aligned}&\left(\mathbf{M}_{t+\Delta t} + \frac{\Delta t}{2}\mathbf{C}_{t+\Delta t}\right)\ddot{\mathbf{u}}^{(2)}_{t+\Delta t}\\ &= -\mathbf{C}_{t+\Delta t}\left(\dot{\mathbf{u}}_t + \frac{\Delta t}{2}\ddot{\mathbf{u}}_t\right) - \beta\Delta t^2\mathbf{K}_{t+\Delta t}\ddot{\mathbf{u}}^{(1)}_{t+\Delta t}\\ &\quad -\mathbf{K}_{t+\Delta t}\left\{\mathbf{u}_t + \Delta t\dot{\mathbf{u}}_t + \frac{\Delta t}{2}(1-2\beta)\ddot{\mathbf{u}}_t\right\} + \mathbf{F}_{t+\Delta t}\end{aligned} \tag{6}$$

In this equation, the acceleration term in the right hand side is represented by $\ddot{\mathbf{u}}^{(1)}_{t+\Delta t}$, and the acceleration term in the left hand side is represented by $\ddot{\mathbf{u}}^{(2)}_{t+\Delta t}$. $\ddot{\mathbf{u}}^{(1)}_{t+\Delta t}$ means the first predictor, and $\ddot{\mathbf{u}}^{(2)}_{t+\Delta t}$ the second. Then the n th predictor (n th iteration) can be represented using the (n-1) th after predictor as follows:

$$\begin{aligned}&\left(\mathbf{M}_{t+\Delta t} + \frac{\Delta t}{2}\mathbf{C}_{t+\Delta t}\right)\ddot{\mathbf{u}}^{(n)}_{t+\Delta t}\\ &= -\mathbf{C}_{t+\Delta t}\left(\dot{\mathbf{u}}_t + \frac{\Delta t}{2}\ddot{\mathbf{u}}_t\right) - \beta\Delta t^2\mathbf{K}_{t+\Delta t}\ddot{\mathbf{u}}^{(n-1)}_{t+\Delta t}\\ &\quad -\mathbf{K}_{t+\Delta t}\left\{\mathbf{u}_t + \Delta t\dot{\mathbf{u}}_t + \frac{\Delta t}{2}(1-2\beta)\ddot{\mathbf{u}}_t\right\} + \mathbf{F}_{t+\Delta t}\end{aligned} \tag{7}$$

In this equation, since $\mathbf{M}_{t+\Delta t}$ and $\mathbf{C}_{t+\Delta t}$ can be diagonalized, $\ddot{\mathbf{u}}^{(n)}_{t+\Delta t}$ can be obtained without calculating the inverse matrix. By iterating this calculation until convergence, $\ddot{\mathbf{u}}_{t+\Delta t}$ is finally obtained.

2.3.2 Prediction of the acceleration term

In this method, it is important to predict appropriate initial acceleration term $\ddot{\mathbf{u}}^{(1)}_{t+\Delta t}$, If an inappropriate initial predictor were given, $\ddot{\mathbf{u}}_{t+\Delta t}$ could not be converged.

However, if we can choose a valid prediction, the solution may be converged in smaller number of iterations. In order to realize a real-time simulation, calculation performance more than 40 Hz is required.

When the time step Δt is enough small, we can assume approximately that the acceleration changes linearly. Therefore, in this method, the first prediction was given as follow:

$$\ddot{\mathbf{u}}^{(1)}_{t+\Delta t} = 2\ddot{\mathbf{u}}_{t} - \ddot{\mathbf{u}}_{t-\Delta t} \tag{8}$$

This assumption would be appropriate on the condition that the acceleration changes slightly.

2.3.2 Convergence of iteration

In this section, we discuss the convergence condition of $\ddot{\mathbf{u}}^{(n)}_{t+\Delta t}$. By subtracting Eq. (7) from the equation for (n+1), Eq (9) is given.

$$\ddot{\mathbf{u}}^{(n+1)}_{t+\Delta t} - \ddot{\mathbf{u}}^{(n)}_{t+\Delta t} = -\beta\Delta t^2\left(\mathbf{M}_{t+\Delta t} + \frac{\Delta t}{2}\mathbf{C}_{t+\Delta t}\right)^{-1}\mathbf{K}_{t+\Delta t}\left(\ddot{\mathbf{u}}^{(n)}_{t+\Delta t} - \ddot{\mathbf{u}}^{(n-1)}_{t+\Delta t}\right) \tag{9}$$

The convergence condition is given as follows:

$$\lim_{n\to\infty}\left(\ddot{\mathbf{u}}^{(n)}_{t+\Delta t} - \ddot{\mathbf{u}}^{(n-1)}_{t+\Delta t}\right) = \mathbf{0} \tag{10}$$

$$\lim_{n\to\infty}\left(\ddot{\mathbf{u}}^{(n)}_{t+\Delta t}\right) = \ddot{\mathbf{u}}_{t+\Delta t} \tag{11}$$

Therefore, as for the convergence condition in this method, the following equation is finally obtained.

$$max|\lambda_i| \le 1 \tag{12}$$

where λ_i is the i th eigenvalue of the coefficient matrix in the right hand side of Eq. (9).

3. Experiment

3.1 Hardware

In order to evaluate the effectiveness of this proposed iterative Newmark method, we implemented this algorithm in several kinds of motion simulations of the object in the virtual environment. We used a workstation (SGI Octain R12000 300MHz×2, IRIX 6.5).

3.2 Judgement of convergence

Based on Eq. (10), we regarded that $\ddot{\mathbf{u}}^{(n+1)}_{t+\Delta t}$ is converged to $\ddot{\mathbf{u}}_{t+\Delta t}$ on the following condition.

$$\ddot{\mathbf{u}}^{(n+1)}_{t+\Delta t} - \ddot{\mathbf{u}}^{(n)}_{t+\Delta t} \le 10^{-5} \tag{13}$$

If $\ddot{\mathbf{u}}_{t+\Delta t}$ is not converged after more than 100 iterations, the time step Δt is reduced in half in order to avoid divergence.

3.3 Analysis model

As for the analysis model, simple spring model was used. Fig. 1 shows the example of the analysis model. This model consists of springs, dampers and masses, and the springs are intersected partially in order to represented a share stiffness. Though the model is not completely accurate for the purpose of the strict scientific analysis, it may be used to simulate the deformation of the object in the virtual world.

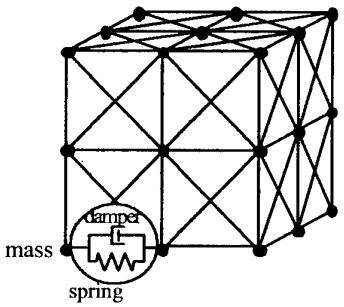


Fig. 1 Analysis model

3.4 Boundary condition

In this experiment, the proposed method described in the previous section was applied to three types of movements, such as deformation, translation and rotation. Fig. 2 shows the condition for the deformation test. Fig. 3 shows the condition for the movement test that includes the translation and the deformation. Fig. 4 shows the condition for the movement that includes deformation, translation and rotation. We adjusted the virtual world and the real world by sleeping computation, because the calculation time is much faster than the time in the real world in this example.

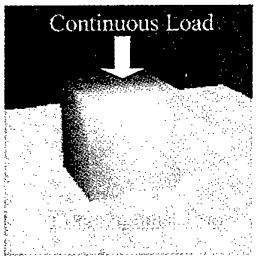


Fig. 2: Deformation test

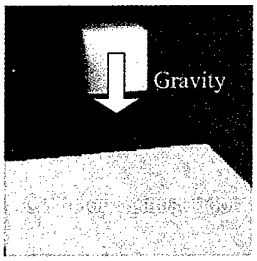


Fig. 3: Translation and deformation test

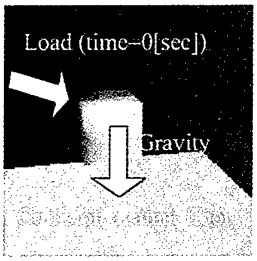


Fig. 4: Rotation, translation and deformation test

3.5 Motion simulation

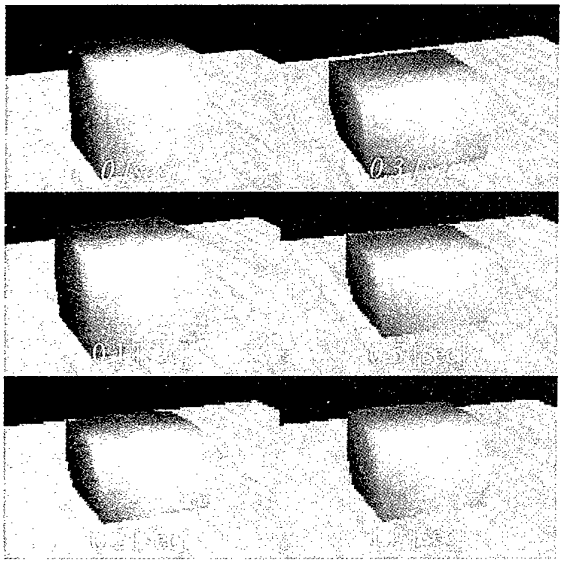


Fig. 5 Movement of deformation

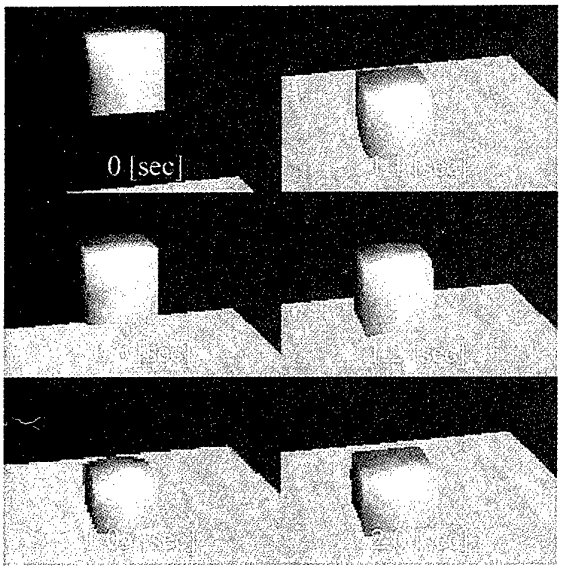


Fig. 6 Movement of deformation and translation

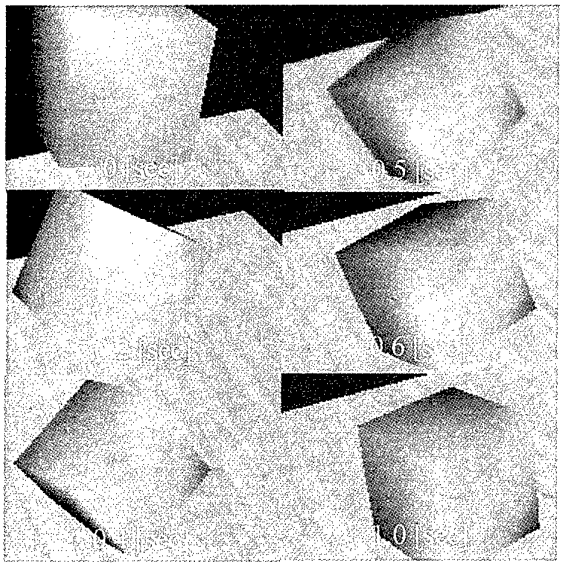


Fig. 7 Movement of deformation, translation, and rotation

In the deformation test, the load was applied to the object from the upper side continuously in time. The movement of the object simulated by the proposed method was shown in Fig. 5. In Fig. 5, after the vibrations, the equilibrium state was obtained between applied forces and the reaction forces.

In this experiment, since the computation time for one step was shorter than Δt (=0.001[sec]), a real-time simulation was realized. This spring model was simply assumed to be the linear, however, the simulated move-

ment of the object seemed to be natural.

In the translation and deformation case, the object was dropped according to the gravity. In Fig. 6, the simulated movement of the object is shown. In this experiment, the simulated movement seemed to be natural, though the strict physical model, such as the friction between the object and the floor, was not implemented.

In the deformation, transition and rotation case, the object was dropped with the applied force from the side direction. In Fig.7, the simulated movements of the virtual object was shown. In this experiment, the rotation on the floor is somewhat unnatural. For example, the objects rotated to the wrong direction in this model, because we didn't take the friction against the floor and the balance of the angular moment into account. In order to achieve more realistic motion including rotation, the strict physical model should be required. However, in this experiment, the real-time calculation was achieved by using the proposed method.

3.6 Comparison of results by Newmark-β method and iterative Newmark method

We compared the average computation costs for solving $\ddot{\mathbf{u}}_{t+\Delta t}$ the both in the Newmark-β method and the iterative Newmark method. Table 1 shows the result. The numerical experiments were conducted under the same condition of deformation simulation in Fig. 2. The analysis model shown in Fig. 1 was used, and the number of nodes was varied from 8 to 64. In the both cases, the computation costs were increased according to the increase of the number of nodes. And, in any case, the computation cost in the iterative Newmark method was smaller than the Newmark-β method.

Table 1: Computation cost for the solution of $\ddot{\mathbf{u}}_{t+\Delta t}$

Node	Newmark-β method [ms]	iterative Newmark method [ms]
8	1.732	0.051
27	2.093	0.219
64	10.065	0.597

3.7 Iteration number and computation time

We simulated several movements by changing the time step Δt and the iteration number to the convergence was counted. In this test, the analysis model shown in Fig.1 was used and the number of nodes was 27. Fig. 8, Fig. 9 and Fig. 10 show the iteration numbers of the calculation for each test. In these figures, the iteration numbers for $\Delta t = 0.01$[s] and $\Delta t = 0.001$[s] were compared among these movement tests.

From these results, the number of the iteration was obviously decreased when the fraction size of the time step was small. In addition, it was noted that the iteration number became large, when the acceleration was changed suddenly at the collision point against the floor as shown in Fig. 9.

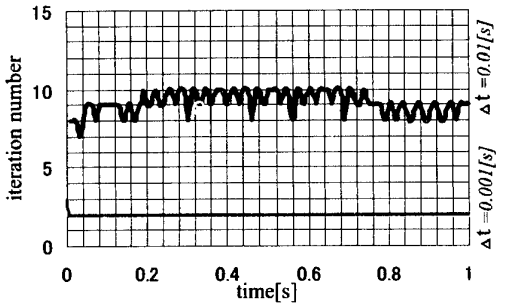


Fig. 8: Iteration numbers of the case in Fig. 2

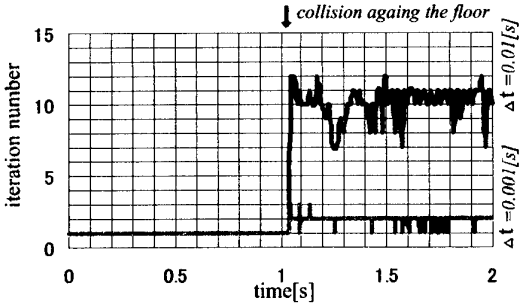


Fig. 9: Iteration numbers of the case in Fig. 3

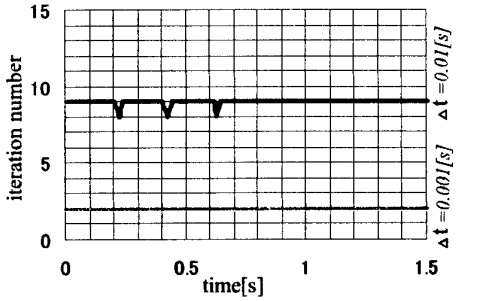


Fig. 10: Iteration numbers of the case in Fig. 4

Furthermore, we examined the computation cost and the iteration number in detail. The analysis model of Fig. 1 was used, and the number of nodes was changed from 8 to 64.

In Fig. 11, we can see that, in the case of Δt =0.01[s], the computation cost was increased according to the increase of the node number. Then we compared the computation cost for each iteration (Fig. 12) and the average iteration number (Fig. 13).

In Fig. 12, the computation cost for each iteration was almost the same between in the case of Δt =0.01[s] and Δt =0.001[s]. The computation cost increased linearly, when the analysis model were bigger.

However, we found from Fig. 13 that the iteration number became larger when Δt =0.01[s], and the total computation time depended on the iteration number.

In order to realize a real-time simulation, the total computation time (iteration number times each computation time) must be shorter than the time step. Therefore, we must carefully examine the relation between Δt and the iteration number to realize the real-time simulation. In addition, we must also examine the more appropriate prediction method of the acceleration term.

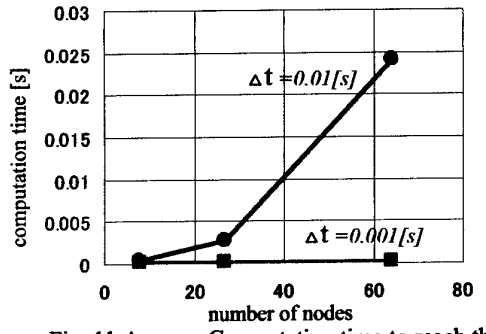


Fig. 11 Average Computation time to reach the solution of the next time step $\ddot{\mathbf{u}}_{t+\Delta t}$

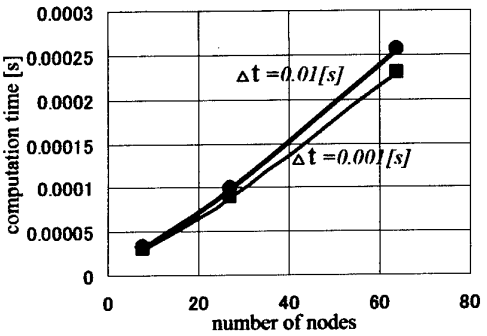


Fig. 12 Computation time to reach the solution of the next iteration $\ddot{\mathbf{u}}^{(n)}_{t+\Delta t}$

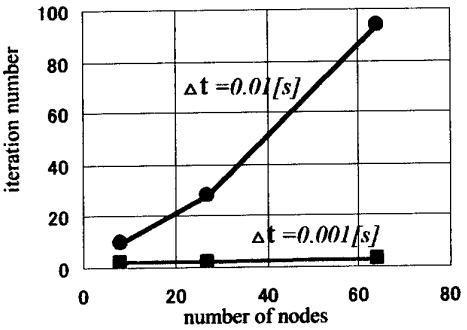


Fig. 13 Iteration number to reach the solution of the next time step $\ddot{\mathbf{u}}_{t+\Delta t}$

4. Conclusions

In this study, in order to realize real-time deformation analysis in virtual reality, a new efficient time integration scheme for finite element method was proposed.

This method named 'iterative Newmark method' is suitable for virtual reality applications, because this method has the result by computational efficiency and stability, compared with the Newmark-β method.

This method was applied to several motion simulations such as deformation, movement and rotation of the object, and the real-time calculation was achieved by using the proposed method.

References

1.Koichi, HIROTA., Toyohisa, KANEKO.: Representation of Soft Objects in Virtual Environment, ICAT 98, pp.59-62(1998).

2. Belytschko,T. and Thomas,T.J.R.: COMPUTATIONAL METHODS FOR TRANSIENT ANALYSIS,NORTH-HOLLAND,Chap.2 (1983)

ICAT 2000

An Immersive Modeling Workbench using a Combination of Two- and Three-Dimensional Interface

Haruo Takemura, Hayato Yoshimori, Masatoshi Matsumiya, and Naokazu Yokoya
Graduate School of Information Science,
Nara Institute of Science and Technology
takemura@is.aist-nara.ac.jp

Abstract

In this paper, described is a prototype of a novel hybrid 3-D object modeling system, NIME - NAIST Immersive Modeling Environment, which inherits the advantages of Basic concept of our system is to combine advantages of 2-D and 3-D modeling environments in one environment. By employing a slant rear-projection display, NIME integrates 2-D and 3-D modeling environments into a unified modeling space. On the surface of the display, NIME provides a user 2D GUI modeling interface. NIME also provides the 3D modeling environment with a field sequential stereoscopic imaging of objects and 6-DOF pen-type input device. A user can create models seamlessly switching between these two modeling environments.

Key words: Modeling, Virtual Reality, 3-D Interface

1. Introduction

In this paper, described is a prototype of a novel hybrid 3-D object modeling system, NIME - NAIST Immersive Modeling Environment, which inherits the advantages of both traditional 2-D GUI based modeling and 3-D immersive modeling environments.

3-D computer graphics (3D GG) [1], today, are widely used in various fields of visual expression, such as motion pictures, television, graphic design, presentations, and home video games. Since computers have advanced fast enough to render various complex shapes in a small amount of time, 3-D modeling methods that can efficiently model various complex shapes are needed.

In general, 3-D CG software for 3-D modeling use traditional WIMP (Windows, Icon, Menu and Pointers) interface, which uses a CRT monitor and a mouse, a 2-D input device [2,3]. In these modeling environment which utilize 2-D display surface, input degree of freedom which users can simultaneously control is limited to just one or two. This enables users to design objects accurately and precisely. However, as 3-D objects are designed using 2-D input devices, a mental mapping between 2-D input space and 3-D modeling spaces is used in users' cognition. Thus, it is possible to assume that the mental workload of controlling 3-D object using 2-D input device is relatively high compared to that of 3-D direct manipulation.

In such an environment, a 3-D operation must be decomposed into a combination of 2-D operations, which is not intuitive [4]. Moreover, a lack of depth perception makes it difficult for user's to understand objects' shape and their spatial relationship [5].

In order to overcome these problems, Virtual Reality (VR) technologies, which typically use a head mounted display or 3-D mice, are used in several experimental 3-D modeling systems. These systems are called immersive modelers, as users of such systems immersed in a 3-D environment where a user can directly manipulate 3-D objects [4-12].

3-D object modeling using immersive modelers have the following advantages.

1) Objects can be displayed stereoscopically with depth perception and motion parallax. Therefore, the shape of complex objects can be easily understood.

2) By using input devices with three DOF or more, modeling objects can be directly manipulated or altered in 3-D space with intuitive manner. There is no need to perform mental mapping between 2-D input space and 3-D working space.

However, it is also known that humans are not good at simultaneously controlling multiple degrees of freedom and are not good at precise or accurate operation in 3-D space. This results in difficulties of performing accurate or precise design operation in immersive modelers.

Several methods to improve designing performance or accuracy in immersive modelers are reported [13-16]. For example, force or tactile feedback, which limits inputs DOF, is used in several systems. Other uses grid or other constraints or collision detection or avoidance, which limits the degree of freedom in operation when objects interfere with other objects [6].

However, posing constraints does not always provide as good performance as it may get when used in 2D environment. Also, it is known that typical force or tactile feedback device limits the user's workspace or needs large mechanical structure around users

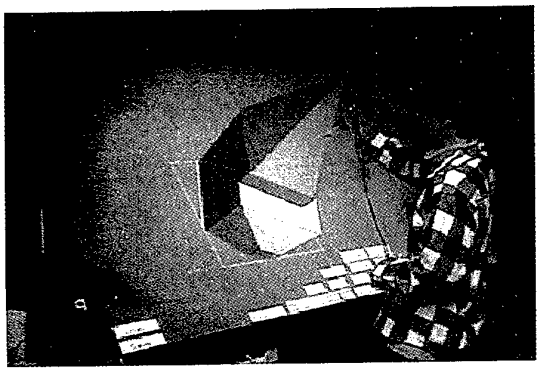


Figure 1: A scene of modeling in NIME

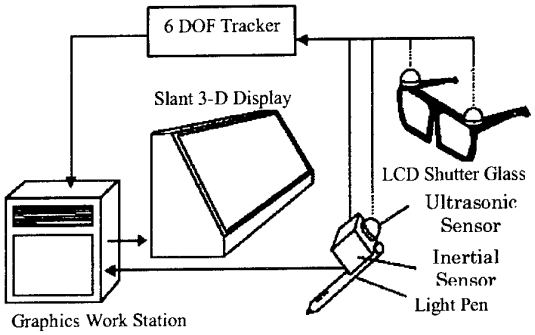


Figure 2: System Configuration of NIME

workspace.

Therefore, 2-D and 3-D environment has its own merits and demerits. It is more reasonable to combine both 2-D and 3-D environment so that a user can gain benefit of both environment.

In this paper, we propose to combine these two modes of operation seamlessly in single modeling application. In particular, we employ slant 3-D display of which surface can be used as drawing table and still the user can view screen stereoscopically with motion tracking stereo. Basic concept of our system is to combine advantages of these 2-D and 3-D modeling environments in one environment.

In the following sections, the system's overview, user interface design for modeling operation, examples of modeling operation, and discussion about the feasibility of proposed methods are described respectively

2. NIME System's Overview

Figure 1 is a picture of a user using the system. A user wears LCD shuttered stereo glasses and holds a 3-D light-pen-type input device. A modeling object is displayed in viewpoint tracking stereoscopic display. A user can have not only binocular parallax but also motion parallax when viewing the objects.

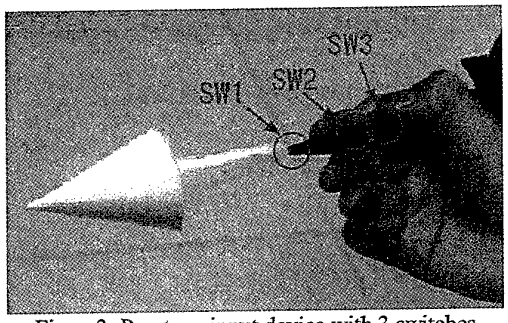


Figure3: Pen-type input device with 3 switches

2.1 System's Configuration

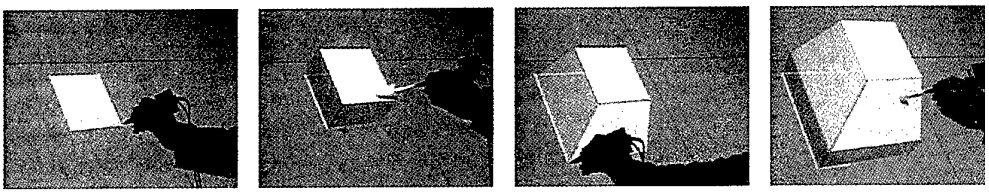
Figure 2 illustrates the system's configuration. By employing a slant rear-projection display, NIME integrates 2-D and 3-D modeling environments into a unified modeling space. On the surface of the display, NIME provides a user 2D GUI modeling interface. NIME also provides the 3D modeling environment with a field sequential stereoscopic imaging of objects and 6-DOF pen-type input device. Therefore, a user can create models seamlessly switching between these two modeling environments.

Unlike the conventional 3D CG software systems, which display the projected images of manipulation targets, NIME provides a user the 2D modeling environment by showing the intersection of targets and the display surface on the screen. By showing both stereoscopically displayed object and its intersection all the time, a user can create objects seamlessly either in 2D or 3D modeling environments without any operation to switch one modeling environment to another.

2.2 Input Device

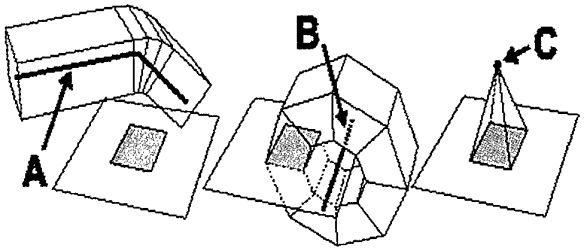
A 6-DOF pen-type input device (Fig. 3) is developed and used in this system. The device is a combination of a light pen, an inertial sensor, and an ultrasonic sensor. This pen-type input device can be used in both 2D and 3D modeling environments with the 3 switches arranged at the tip and the side part. By calculating the distance between the display surface and the tip of the pen-type input device, NIME detects which modeling environment the user intends to use. When the distance is within 5 mm, the user's operation is considered as for the 2D modeling environment and a dot cursor is shown according to a series of input from the light pen. On the contrary, when the distance is beyond 5 mm, it is considered as for the 3D modeling environment and an arrowhead cursor is shown according to a series of input from the inertial sensor and the ultrasonic sensor.

Figure 3 also shows the arrowhead cursor, which is used in 3-D modeling environment. By showing arrowhead cursor, a user can see if he in 3-D modeling mode or not. An arrowhead cursor also helps user to converge his or her eyes to see stereoscopically displayed objects,



(a) (b) (c) (d)

Figure 4: An example of "direct extrude"



(a) path-extrude (b) revolving (c) pyramidal shape

Figure 5. Prepared operations and their referential objects

because it has consistent accommodation and convergence with other 3-D objects displayed on the screen.

Forsberg et al. [17] build similar system, but their system does not allow users to perform modeling in 3-D environment. In their system, modeling operation is carried out in 2-D environment and 3-D environment is mainly used to display created shapes on a field sequential stereoscopic display.

3. Modeling Objects with Two- and Three Interface

In NIME, a user can perform a number of modeling operations in both 2D and 3D environments according to the nature of each operation.

3.1 Creating Objects

To create objects, first, a user makes 2D plane shape in 2D modeling environment. By using lithe-pen type device, a user can draw 2-D shapes on display surface. Several types of predefined shapes such as square or circle can also be used.

After creating a 2-D shape, a user can create 3-D objects based on this 2-D shape. One way to create a 3-D object is to extrude the 2D shape directly into 3D space. We call this way of creating objects "direct extrude". Once a user extrudes the 2D shape, the intersection of the object becomes to be possible to edit.

Fig.4 shows an example of "direct extrude" operation;

(a) A rectangle is created in 2D modeling environment by clicking one corner of the rectangle and expanding the rectangle by dragging the diagonal corner of the rectangle.

(b) The rectangle becomes a parallelepiped by "direct extrude" operation. The operation is performed by dragging the rectangle towards a perpendicular direction against the display surface.

(c) The intersection can be edited by clicking and dragging a vertex of the shape and

(d) "Direct extrude" can be repeated again.

Three more types of object creation from 2-D shapes are prepared in NIME system. These are "path-extruded", "revolving" and "pyramidal shapes" operations as shown in Figure 5 respectively. Same as "direct extrude", first, a user creates 2D shape on a display plane. After that, a user specifies which types of objects a user wants to create by clicking a specific button placed on the surface of the display with a pen device. Then, a referential object that corresponds to the specified shape appears on the screen. The referential object is a extrude path for "path-extruded"(A in Fig. 5), the axis of rotation for "revolving"(B in Fig. 5), or the summit point for "pyramidal shapes"(C in Fig. 5). Users can always edit the referential objects during creation of each object, and at the same time, the original 2D plane shape can also be edited. Users can confirm the effects of the change to the referential object and the original 2D plane shape immediately in 3D modeling environment stereoscopically.

3.2 Object Modification

To modify created objects, NIME offers four types of operations to users, these are, "extrude the intersection", "edit points", "virtual magnet" and "Boolean operation".

By using "extrude the intersection", users can modify the object as shown in Figure 6. In this example,

(1) The sphere is arranged so that the bottom part will intersect with the surface of the display (Fig. 6 (a) and (b)).

(2) After pushing button of "extrude the intersection" in 2-D menu, a user can extrude the intersection by translating the object in 3D space (Fig. 6 (c)).

(3) Same as the "direct extrude", the intersection can be edited by clicking each vertex of intersection or

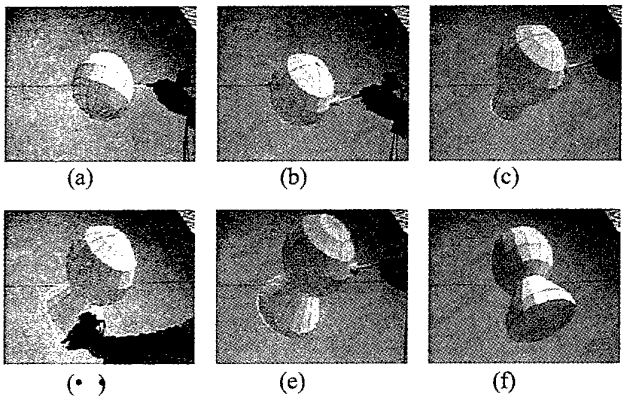


Figure 6: An example of "extruding intersection"

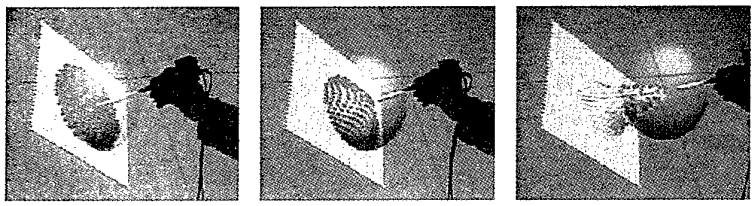


Figure 7: Shape deformation using "virtual magnet"

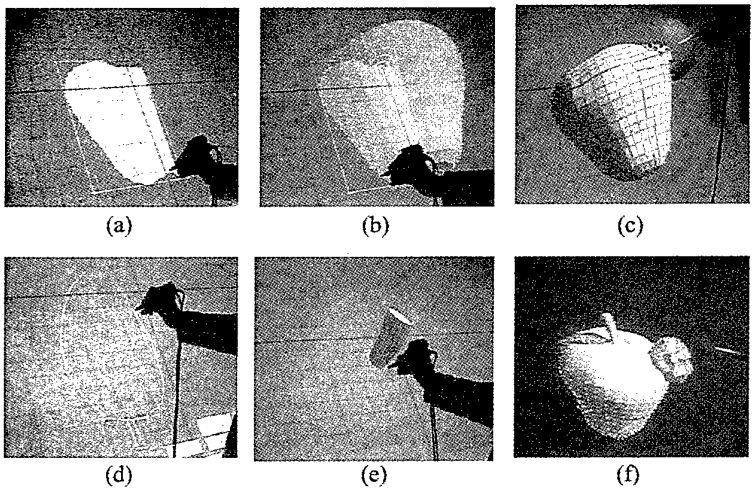


Figure 8: An example of modeling an apple

clicking and dragging handle of intersection (Fig. 6 (d)).

(4) Then, a user can repeat extrusion by pulling up the object (Fig. 6 (e))

(5) Finally, to stop extrusion and finalize the modification, a user pushs the menu button again.

Users can move the selected points or vertices of object both in 2-D and 3-D modeling environments. When users move the pen-type input device in 2-D modeling environment, the selected points translate only on the 2-D plane. When in 3-D modeling environment, the selected points translate freely in 3-D direction.

The "virtual magnet" is used when a user wants to give the object smooth gradation. An example of using "virtual magnet" is shown in Figure 7. When a user selects "virtual magnet", the arrowhead of the cursor changes to a spherical head, which shows the area of influence of "virtual magnet". A user can modify the object by moving the pen-type input device. Vertices of the objects are attracted to virtual magnet and objects' shape is modified as shown in Fig. 7. In this example, a flat surface is modified by pulling central part of plane making shape like a mountain.

The "Boolean operation" is implemented to perform Boolean operation among objects in 3-D modeling environment. Logical "and ", "or", and "exclusive or" operations are prepared. These operation enables a user to combine multiple objects created by NIME system. The result of operations is easy to understand, because the operations are performed in 3-D environment and it becomes easier for a user to understand the relationship between objects

3.3 An example of Modeling

In order to discuss feasibility of the proposed modeling method, the modeling process of an apple is shown in Figure 8.

(1) First, the 2D plane shape, which is the cross section of an apple, is created in 2D modeling environment. (Fig. 8 (a))

(2) By revolving the 2D plane shape, the body of an apple is created. (Fig. 8 (b))

(3) The "virtual magnet" operation is performed to give the body some natural distortion or bumps. (Fig. 8 (c))

(4) A leaf is created as 2D plane shape in 2-D modeling environment and bended in 3-D modeling environment using "virtual magnet". (Fig. 8 (d))

(5) Beginning with a circle in 2-D plane, repeatedly applying "direct extrude" in 3-D environment and editing the intersection create a stem of an apple.

(6) Each object is arranged at appropriate position in order to apply "Boolean operation", so that objects are unified into a single apple model.

Figure 9 shows the final rendered image of the apple

created in NIME. At this moment, an attribute to polygons of an object surface for rendering, such as colors etc., cannot be given in NIME system. Therefore, external modeler is used for final touch up of the objects such as coloring and texture mapping.

4. Discussion

Through the modeling of an apple, the following characteristics of the proposed method are confirmed.

(1) 2-D editing surface where a user can enjoy merit of 2-D modeling is embedded in 3-D environment.

(2) A user can enjoy merits of both 2-D modeling environment and 3-D modeling environment while a user performs modeling without explicitly switching operational mode.

In this prototype system, a slant display is used. The surface of the display successfully provided a user a physical drawing surface, which constraints the users controllable degrees of freedom and helps their easy free hand drawing on the surface. This is mainly due to the following two reasons. Firstly, a pen type input device has an appropriate friction against the display surface and provided users an appropriate tactile and force feedback. Secondly, a slanted display created an appropriate drawing surface just like a drafting table. As a result, 2-D drawing in this system is found relatively natural and easy.

The switching between 2-D modeling and 3-D modeling in this system was implicitly performed based on the modeling command user performs and 3-D position of the pen type input device. A user can smoothly work on modeling without explicitly switching modeling modes. However, it is found that a user sometimes confused, when he is not familiar with the modeling operation implemented in this system. In other words, a user has to know which modeling operation is performed in 3-D mode and which in 2-D mode. This caused some modeling difficulties, when a novice user tests the system.

Figure 9: A photo-realistic rendered apple

5. Summary

We have built a prototype of an immersive modeling system which combines 2-D and 3-D GUI. The system consists of a large slant rear projector and 3-D light –pen type input device. A user can seamlessly combine 2-D and 3-D operation to model objects. The feasibility of the method is discussed based on Informal user study. The study suggests that the system is easy to use for those who have an introductory knowledge of computer graphics. However there are some difficulties in using NIME, when a user first timers of CG modeling.

For future study, we are planning to conduct more detailed study on usability of the system. At the same time, the system will be expanded to accommodate more useful operations such as giving an objects' color and so on.

Acknowledgment

This work is partly supported by the JSPS Grant-in-Aid for Scientific Research no. 11558038.

References

1. LightWave 3D Version 5.5 Reference Manual, NewTek, Inc, 1997.
2. 3D Studio MAX RELEASE 3 Reference Manual, Autodesk, Inc, 1999.
3. Foley, J., van Dam, A., Feiner, S. and Hughes, J.: Computer Graphics, Addison-Wesley Publishing Company, Inc, 1990.
4. Liang, J. and Green, M.: JDCAD: A Highly Interactive 3D Modeling System," Proc. 3rd International Conf. on CAD and Computer Graphics, pp.217-222, 1993.
5. Butterworth, J., Davidson, A., Hench, S. and Olano, T. M.: 3DM: A Three Dimensional Modeler Using a Head-Mounted Display," Proc. ACM Symposium on Interactive 3D Graphics, pp.135-139, 1992.
6. Kiyokawa, K., Takemura, H., Katayama, Y., Iwasa, H., Yokoya, N. VLEGO: A Simple Two-Handed 3D Modeler in a Virtual Environment", Electronics and Communications in Japan, Part3, Vol.81, No.11, pp.18-28, 1998.
7. Bowman, D. A. and Hodges, L. F.: User Interface Constraints for Immersive Virtual Environment Applications," Graphics, Visualization and Usability Center Technical Report, GIT-GVU-95-26, 1995.
8. Stoakley, R.,Conway, M. and Pausch, R.: "Virtual Reality on a WIM: Interactive Worlds in

Miniature," Proc. ACM CHI'95 Conf. on Human Factors in Computing Systems, pp.265-272, 1995.

9. Mine, M.: Working in a Virtual World: Interaction Techniques Used in the Chapel Hill Immersive Modeling Program," UNC Chapel Hill Computer Science Technical Report TR96-029, 1996.

10. Deering, M. F.: The Holosketch VRSketching System," Communicationsof the ACM, Vol.39, No.5, pp.54{61, 1996.

11. Hill II,L. C., Chan, C. and Cruz-Neira, C.: Computer Aided Design in an Immersive Environment: The Virtual Architectural Design Tool (VADeT), "http://www.icemt.iastate.edu/lchill/cpe575paper/cpe575.html, 1997.

12. Zeleznik, R. C., Herndon, K. P. and Hughes, J. F.: "SKETCH: An Interface for Sketching 3D Scenes," Proc. SIGGRAPH'96, pp.163-170, 1996.

13. Schmandt, C. M.: "Spatial Input/Display Correspondence in a Stereoscopic Computer Graphics Workstation," Proceedings of ACM SIGGRAPH '83, pp.253-262, 1983.

14. Hinckley, K., Goble, J. C., Pausch, R. and Kassell, N. F.: "New Applications for the Touchscreen in 2D and 3D Medical Imaging Workstations," Proc. SPIE Medical Imaging '95, Image Capture, Formatting, and Display, 1995.

15. Mine, M.: "A Review and Analysis of Through-the Window Computer-Aided Modeling Systems," UNC Chapel Hill Computer Science Technical Report TR94-070, 1994.

16. Hinckley, K., Rausch, R., Goble, J. C. and Kassell, N. F.: "A Survey of Design Issues in Spatial input," Proc. ACM Symposium on User Interface Software and Technology, pp.213-222, 1994.

17. Forsberg, A. S., LaViola Jr, J. J. and Zeleznik, R. C.: "ErgoDesk: A Frame work for Two- and Three-Dimensional Interaction at the ActiveDesk," Proc. 2nd International Immersive Projection Technology Workshop, 1998

A Rendering Module of MPEG-4 System Based on VRML97 for Virtual and Natural Scene Integration

Kuo-Luen Perng, Yu-Chung Lee, Wei-Ru Chen, Yu-Li Huang, An-Lung Teng, and Ming Ouhyoung

Communication and Multimedia Lab
Dept. of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan, R.O.C.

Abstract

An architecture design and implementation of an MPEG-4 rendering module is proposed. MPEG-4 is an object-based international compression standard established by ISO, and this standard has many significant features that make it very suitable for Internet applications with variable or very low bit rate. In addition, the object-oriented characteristic allows greater user interaction than before. To obtain the powerful and attractive features of MPEG-4, the rendering module has to interpret the scene structure in MPEG-4 scene description language, which is based on VRML97 – the most famous industrial standard to describe a 3D scene on the Internet. Various nodes are implemented – geometry nodes, non-geometry nodes, route nodes, texture nodes, etc. As an example, the system is able to show three moving cubes with each face containing a video running at 70 fps.

Because MPEG-4 is a highly extensible standard, new features are possibly added as objects of the scene. The system should have good flexibility to include new features. We'll take a panoramic image viewer as an example to show the ability of our system to integrate new features.

Figure 1: Using our rendering module to browse an MPEG-4 scene, which is composed of natural and synthetic objects.

Key words: MPEG-4, DirectShow, VRML97, BIFS

1. Introduction

There are lots of multimedia standards in storage and communication usage established by the organizations such as ISO or ITU. But when being applied to environment differs from its original purpose, most of them will lead to unpredictable outcomes. In the recent years, technology in communication and multimedia field is making great progress. Various new applications are appearing in different fields rapidly, which are with different bandwidths, different computation powers, different transmission error rates, etc. Obviously, old multimedia standards are becoming unable to satisfy applications in different environments. MPEG-4[1] [3][4] is a standard with new ideas in many aspects. First, to compare with previous frame based standards, MPEG-4 takes "object" as the basic unit of the scene. Each object could be edited or adjusted individually, and could be treated with dif-

ferent codecs. It brings great flexibility and freedom to the content authors, service providers, and end users. Another significant feature of MPEG-4 is the ability of Synthetic Natural Hybrid Coding (SNHC). This not only enriches the content of MPEG-4 scenes, but also leads to more reasonable manipulation of limited bandwidth. To accomplish the above features, MPEG-4 must draw up a scene description language to describe the structure of the scene. The language takes VRML97[2] as the basis and adds some new nodes for other purposes. The rendering module composites and renders the scene according to the structural information and the media samples dealt by the visual codec. Furthermore, the rendering module has to implement several important mechanisms so that the MPEG-4 system can bring its ability into full play, such as navigation in the scene, changing the viewpoint, individually adjusting playing quality of video objects, and the animation mechanism.

1.1 System Overview

In essence, our system is an implementation of a VRML browser under the MPEG-4 architecture. The difference between other VRML browsers and ours is the VRML scene data acquired through the BIFS Decoder (Binary Format for Scene Stream Decoder). The video/audio data required by scenes are processed through a video/audio decoder in our system.

Our rendering module consists of the following tasks. Two of them are about composition and displaying the scene onto a screen, and others are about cooperation with other modules in the system:

1. To control the 2D/3D rendering engine.
2. To interpret the scene tree structure, compose the scene, and set up the geometry framework.
3. To support the node definition and the structural mechanism of scene description language.
4. To link up with the media codec, get the visual media sample, and manage buffers.
5. To interact with users, provide navigation ability, and feedback users' requests to the system.

Figure 2: The figure above is the implementation of our rendering module which is the part to the right of the line of COI (Composition Interface).

Before the final MPEG-4 system integration currently, we have our own independent testing environment. In this testing system, MPEG-4 scenes are described in the VRML grammar, and then are interpreted by the parser. The decoder for still images/video can read the necessary texture data in advance for testing.

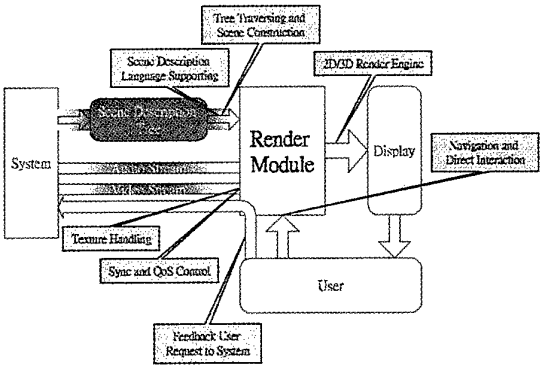


Figure 3: Illustration of the I/O flow of the whole rendering module.

In the implementation of MPEG-4, we use the Microsoft multimedia architecture, Directshow. From software points of view, the kernel of DirectShow is a modularized pluggable system, based on the usage of the so-called filters. The most significant advantage of DirectShow comes from its ability to make the multimedia application design more clear and easy. By carefully dividing the work into connected filters in the DirectShow architecture, each filter can be implemented by different program developers. Another advantage of DirectShow is the filter re-use, which speedups the developing of new multimedia applications. So our program of rendering can be independent from other parts in the system, and is wrapped to be a filter according to the DirectShow architecture.

2. Implementation

The rendering module is developed on the Microsoft Windows 98/2000 platform. OpenGL and DirectX are used to implement rendering. In order for the convenience of cross-platform compatibility, we wrapped our program in a new interface for the use of OpenGL and DirectX. In actual implementation, when there are more video textures in the scene, we can have greater performance by adopting DirectX for rendering, because we can take advantage of hardware acceleration on most video cards that supports the DirectX hardware abstraction layer. Currently there are no special functions designed for 2D image processing in OpenGL, so acceleration for 2D image processing is processed purely by software. If there are not many video textures, the per-

formance in adopting DirectX or OpenGL is nearly the same.

2.1 Scene Tree

Scenes are depicted by various nodes in VRML. Our system records all of the nodes in the scene by constructing a tree, and the tree structure is called "Scene Tree". In the integrated system, the scene tree is constructed by the BIFS Decoder. In the testing platform prior to the final system integration, a VRML parser sample provided by SGI is used to interpret the scene tree after reading the VRML file.

Each time a scene needs to be re-rendered, the whole scene tree will be re-traversed. When a node is met, the corresponding handle function is executed. After traversing all the nodes in the scene tree, the rendering of the scene is completed.

In the rendering module, traversing a scene tree will be iterated, not just done once at first. It is because that in the MPEG-4 system, the tree structure will be updated in run time, and it's possible for the route mechanism to change the data of the nodes in the tree all the time.

2.2 Geometry Node

In the initial stage, the nodes such as sphere, cone, circle, and cylinder will need that all required triangular data be pre-calculated. Our program provides users to adjust the level of details of these figures.

In the future, we will add a function to have the program itself adjust the level of details. So when objects are far away, the program will automatically draw these nodes with fewer triangles.

The mechanism of text-showing nodes is different in Direct3D and OpenGL. Because until so far, the program still cannot get the vector data of letter forms. In Direct3D, every character is implemented as a rectangular polygon mapped with a texture including alpha blending data. But OpenGL provides functions to convert letter forms to OpenGL lists, so the real vector characters can be drawn.

2.3 Non-geometry Node

Some nodes are not designed for drawing, like interpolator nodes, sensor nodes, and transform nodes used to change positions of geometry nodes.

Interpolator nodes are utilized to produce the animation effects of geometry nodes. Each time when an interpolator node is traversed, the interpolated key value is established by the route mechanism. The handle function of the interpolator node is to compute the result of interpolation through the key value.

The handle function of the time sensor is to compute the value of the time counter during each execution. The handle function of the transform node is to record the transformation matrix by a stack mechanism while drawing.

Based on the new transformation data, new matrix can be produced. OpenGL itself provides a stack mechanism, but for Direct3D a stack mechanism was implemented.

2.4 Texture

There are three types of textures: still images, video and CompositeTexture.

With execution efficiency in mind, textures of still images are constructed by the mip-map method, but not for video. In implementation only single texture is needed for video, because it will refresh constantly, unlike still images. The constructed textures for a still image can be used repeatedly while in video the case is different. It will be a waste of time to construct video textures by the mip-map method. The way to handle video textures is: if the required image-space size isn't large, a smaller space size of a video texture can be constructed for it. Since the video part of MPEG-4 contains the function of scalability, the decoder will be informed that lower quality data is required in this case.

CompositeTexture is a texture created from an image of VRML scene, then pasted upon the geometry node. In OpenGL and Direct3D, we use a similar process. We allocate a part of memory, and assign the render target of OpenGL/Direct3D to the memory, and then convert the image of the memory mentioned above to a texture, before pasting it to the geometry node.

2.5 Route Node

The data of route nodes is unique. It's not stored in the scene tree, and we use a separate table to record those nodes which will influence the value of other nodes.

```
struct
{
        NODE *SourceNode;
        FIELD *FieldNode;
        NODE *DestNode;
        FIELD *FieldNode;
};
```

Generally speaking, source nodes can be categorized in two types of nodes: sensor nodes and interpolator nodes.

During traversing a scene tree, the values of source nodes will be computed first. After finishing traversing, all the data of the corresponding source fields will be copied to the destination fields. Thus, next iteration during rendering scenes, new data will be used to derive the animation effects.

2.6 Selecting an Object

The touch sensor provides the function of selecting objects by using a mouse. After pasting a video texture on the geometry node, with the user interface, we can select an object at any time and perform play, stop or pause to the video on the object.

The way to select objects is very different in OpenGL and Direct 3D. OpenGL provides a mechanism of selecting objects. There is no such function in Direct3D. The way we do it is each time after drawing a geometry node, the ZBuffer value of the mouse cursor will be checked. And if the value changes, it means the mouse has pointed to the node. After drawing all the nodes, the last one selected is the real selected node by the mouse.

2.7 User Interaction

Since MPEG-4 Scenes are composed of many 3D objects, the system should provide functions that users can roam or rotate/translate objects in the scene. In the near future, functions for users to insert/delete objects will be added.

3. Applications

Besides MPEG's conventional function of playing video, MPEG-4 rendering can be used for new applications mentioned below.

3.1 Panoramic Image

The first application is to show a spherical panoramic view with arbitrary viewing angles. In implementation, in addition to the most essential function of changing viewing angles, we draw a big sphere, and paste ready-made panoramic image on the big sphere. Thus, our system can be a tool to view a panoramic scene.

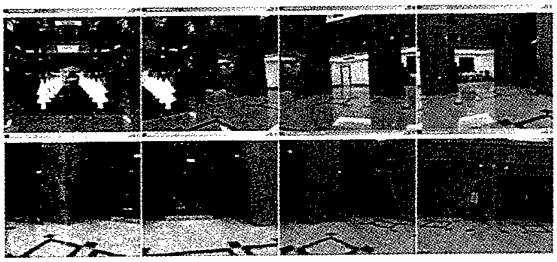


Figure 4: Use our rendering module to combine a panoramic image and a VRML scene together.

3.2 Virtual Meeting

MPEG-4 defines the face node to render human heads. We can render a talking head in real-time from the description of Facial Animation Parameters/Facial Destination Parameters node conveyed from the remote end.

Figure 5: Use our rendering module to display a scene which contains a human head model with facial animation and other synthetic objects.

3.3 Caption Mechanism

In MPEG-1, let caption be part of the image is the only way of showing the caption during the movie is playing. But in MPEG-4, "Text" is an independent geometry node. We can record the subtitles of the movie by real characters, instead of images. Users can change the font size, or the language.

3.4 Non-video Movie

Through the timer mechanism of the time sensor and the act of the interpolator node, a route mechanism can generate the effect of geometry objects moving around in a MPEG-4 scene. We can utilize this function to produce pure 3D animation movies. Users can choose by which viewpoint they want to see a movie, and the required bandwidth may be much lower than the original bandwidth required of video. The resolution of the video is fixed, but the effect of instant 3D rendering can be easily appreciated as users' computer performance raised.

3.5 Multimedia Hyperlink

MPEG-4 scenes provide anchor nodes. Users can attain the function of hyperlinking by anchor nodes. For example, at the beginning, there is a huge television wall and each TV set is broadcasting a different program. After a user selects any of the TV set, this selected TV screen will be maximized to a full-screen view for the user to watch.

4. Conclusion

We have designed and implemented the architecture of an MPEG-4 rendering module. To be integrated with other modules, it will play a key role in a MPEG-4 player or a scene editor. Because of modulization on system design, new features can easily be added into the system.

Our research is running toward the third year, which is a part of an industrial academic collaboration project sponsored by National Science Council (NSC) of Taiwan. Although many research institutions around the world

are devoted into MPEG-4 implementation, our laboratory is among the few ones that proposed a total solution from the deliver layer to the composite layer. The demo system is put on the WWW, and could be downloaded at http://www.cmlab.csie.ntu.edu.tw/cml/g/Projects99.html

An alternative version of our system that is wrapped as an ActiveX control is under developing. The MPEG-4 rendering module can then be combined with Internet, and will bring more fancy applications in various fields.

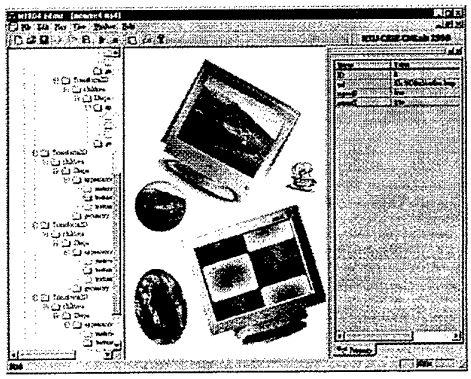


Figure 6: Snapshot of the rendering module, which is integrated with other modules to show 2D nodes. The monitors may contain video objects. A user can individually change the attributes of objects in the scene, e.g. rotate or resize the monitor where a video is attached to it.

5. Future work

Our system hasn't supported all of the MPEG-4 nodes yet, and more functions are under development. If the integration with decoders can be improved, we can accelerate the performance of video playing. For example, let the decoder write data directly into the texture memory. At present, there is only front-half face of the head drawn by the face node, so it seems to be normal only when the head approximately faces to users.

Figure 7: Snapshot of the rendering module, which is integrated with other modules to show 3D nodes. In this picture, there are three moving cubes with each face containing a video, and the background texture is also a video object. The average performance is about 70 fps.

Acknowledgement

This project is partially funded by National Science Council (Taiwan) at CyberLink Co. under the grant NSC88-2622-E-002-002.

References

[1] ISO/IEC FDIS 14496, Information Technology – Generic Coding of Audio-Visual Objects – Part 1: System, Part 2: Visual, International Organization for Standardization, 1998.

[2] ISO/IEC 14772-1:1998, Information technology – Computer graphics and image processing – The Virtual Reality Modeling Language – Part 1: Functional specification and UTF-8 encoding.

[3] Yi-Shin Tung, Ja-Ling Wu, Chia-Chiang Ho "Architecture Design of an MPEG-4 System", Proceedings of IEEE International Conference on Consumer Electronics, June 2000(ICCE 2000), pp. 122-123.

[4] Deng-Rung Liu, Meng-Jyi Shieh, Yu-Chung Lee, and Wen-Chin Chen "On the Design and Implementation of an MPEG-4 Scene Editor", Proceedings of IEEE International Conference on Consumer Electronics, June 2000 (ICCE 2000), pp. 120-121.

[5] I-Chen Lin, Chien-Feng Huang, Jia-Chi Wu, Ming Ouhyoung "A Low Bit-rate Web-enabled Synthetic Head with Speech-driven Facial Animation", in Workshop on Computer Animation and Simulation 2000 (EuroGraphics CAS' 2000), pp. 29-40.

[6] ISO/IEC IS 11172, Information Technology – Coding of Moving Pictures and Associated Audio for Digital Storage Media up to about 1.5 Mbit/s – Part 2: Coding of Moving Picture Information, Part 3: Audio, International Organization for Standardization, 1991.

[7] ISO/IEC IS 13818, Information Technology – Generic Coding of Moving Pictures and Associated Audio Information – Part 2: Video, Part 3: Audio, International Organization for Standardization, 1994.

[8] CCITT SG XV, Recommendation H.261 – Video Codec for Audiovisual Services at px64 kbit/s, COM XV-R37-E, International Telecommunication Union, August 1990.

[9] ITU-T IS, Recommendation H.263 – Video Coding for low bit-rate communication, International Telecommunication Union, November 1995.

[10] ISO/IEC FDIS 14496, Information Technology – Generic Coding of Audio-Visual Objects – Part 1: System, Part 2: Visual, Part 3: Audio, Part 6: DMIF, International Organization for Standardization, 1998.

[11] Video For Window, Microsoft Corporation.

[12] DirectX Media: Multimedia Services for Microsoft Internet Explorer and Microsoft Windows, Microsoft Corporation, October 1998.

[13] PC Video Synchronization and Playback, Microsoft Corporation, November 1998.

[14] DIS 10918 (JPEG), Information Technology- Digital Compression and Coding of Continuous Tone Images – Part 1: Systems, International Organization for Standardization, 1992.

[15] RFC 2205: Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification. R. Braden, Ed., L. Zhang, S. Berson, S. Herzog, S. Jamin. September 1997.

[16] RFC 1899: RTP: A Transport Protocol for Real-Time Applications. Audio-Video Transport Working Group, H. Schulzrinne, S. Casner, R. Frederick & V. Jacobson. January 1996.

[17] International Standard ISO/IEC 14772-1:1997 – VRML 97: The Virtual Reality Modeling Language.

[18] DirectX 7.0 Programmer's Reference, Microsoft Corporation.

ICAT 2000

Composition of 3D Graphic Objects and Panorama

Chu-Song Chen [1], Wen-Ten Hsieh [1,2]

[1] *Institute of Information Science, Academia Sinica, Taipei, Taiwan*

[2] *Department of Computer Science and Information Engineering, National Taiwan University*

Email: song@iis.sinica.edu.tw

Abstract

In this paper, we propose an efficient and easy-to-implement method for the interactive placement of virtual objects in a panorama. In particular, we developed a systematic approach for estimation of the camera parameters using a single panorama with reasonable human-computer interactions.

Key words: 3D/2D Composition, Camera Pose Estimation, Panorama, Augmented Reality, Virtual Reality.

1. Introduction

There are two common approaches to build a VR world: the *image-based approach* (e.g., Quick-Time VR, Surround Video, Real VR, and IPIX) and the *model-based approach* (e.g., AutoCAD, 3D Studio). The panorama is the most popular image-based approach which creates an omni-directional view by seaming photographs. Image-based approach can generate photo-realistic scenes. However, it is difficult to allow the user to view the scene from arbitrary viewing directions. Model-based approaches construct the 3D models of the real world objects and then generate views by rendering the 3D models. It allows the users to interactively view the virtual world from arbitrary viewing directions. However, most model-based approaches use manually-created virtual objects, and thus the generated virtual world is usually not realistic enough for sophisticated objects.

A hybrid VR system is a good solution to exploit both the advantages of these two types of approaches. In this paper, we proposed a simple and systematic method to combine the panorama generated by an image-based approach and the virtual objects generated with a model-based approach. To solve this image-composition problem, two major issues have to be considered: (1) *geometry consistency*, and (2) *photometry consistency*. In this paper, we focus on the problem of geometry consistency. However, our system can also generate realistic shadows of the virtual objects by setting light positions manually. In particular, we allow a user to interactively place the 3D graphic objects in arbitrary positions of the 3D world photographed in a single panorama in a geometrically-reasonable way.

Many methods have been proposed for the composition of virtual objects and images or videos [1][4][8]. However, no approaches are suitable for composition of virtual 3D objects and panoramas because most of the above approaches have to use the disparity information generated by the point correspondences among images. Nevertheless, a panorama is a wide-angle *static* image, while there is no disparity information allowed to be used in a static image. Although some methods can extract 3D structures from panoramas [7][9], at least two panoramas are required.

2. Criterions of Specific Shape

In this paper, we developed a method which can insert virtual 3D objects in a single panorama. To insert 3D graphic objects into a panorama while maintaining their geometry consistency, it is necessary to know the rigid transformation between the object coordinate system and the coordinate system defined by the panorama. This problem is referred to as the *camera pose estimation* in the computer vision community. Basically, estimation of the camera parameters from a single image is ill-posed if there is no additional constraints on the reference objects.

What we try to solve in this paper is to estimate camera parameters using a single panorama. To provide suitable geometrical constraints, our basic idea is to allow the users to draw an appearance of the exemplar shape on the panorama via his/her own perception to the scene. Based on the exemplar shape drawn by human, the camera parameters can be computed by using the related geometrical constraints. In principle, we hope that the exemplar shape satisfies the following criterions:

I. It can provide *sufficient* constraints for computing the camera parameters.

II. It is as simple as possible, so as to release user's b Us burden for drawing it. That is, constraints provided by it are also *not redundant*.

III. It is intuitive and *easy to be perceived* by human.

To find an exemplar shape satisfying the above criterions, the shapes with metric information are not considered because that they are not easy to be perceived

by human. Standard camera calibration [5] or pose estimation methods [3] use the 3D control points with metric information that the distances between each pair of the control points have to be given in advance, and thus they are not suitable for our work. In this paper, we use the shape *without metric information.* In particular, what we need is to use the geometric information less constraining to obtain to human perception, such as parallelism, orthogonality of lines, and so on. Inspired by a previous work [2], we select the specified shape to be *three lines joining at a single point and are orthogonal to each other.* It can also be treated quite naturally as *the origin and the three axes of a 3D Euclidean coordinate system.* In fact, such a coordinate system may appear in many natural scenes (for example, the one shown in Figure 4(b)). It is also easy and intuitive for the users to hallucinate such orthogonal axes (for example, the one shown in Figure 5(b).

3. Camera Parameter Estimation with A Single Panorama

In this section, we show that the exemplar shape selected above provides sufficient constraints for computing the camera parameters. There are usually two types of data structures for storing a panorama: the cylindrical type and the spherical type. Without lost of generality, we use the cylindrical type for the illustration in the sequel. Nevertheless, our method can be easily generalized to the spherical type.

3.1. Intrinsic Parameters

The intrinsic parameters (e.g., focal point and focal length) of any de-warped views of a panorama can be computed directly from the de-warping process for either cylindrical or spherical types of panoramas. In fact, the focal point (i.e., the point which is the orthogonal projection of the lens center in the image plane) of a de-warped image is set to be in its center in almost all cases. The focal length (in pixels) of a de-warped view can be approximately computed by $P/(2\pi)$ where P is the number of pixels of the width of the panorama.

The intrinsic parameters can also be computed more accurately. In fact, an important property of a panorama is that the intrinsic camera calibrations are recovered as part of the panorama construction [9]. That is, the intrinsic parameters can be directly computed from the panorama. More precisely, consider the panorama recorded in the surface of a cylinder as shown in Figure 1(a). A panorama viewer allows the user to see the contents of the panorama from arbitrary viewing directions specified by the user. The panorama viewer de-warps the panorama recorded in a cylinder to an image in a plane, as shown in Figures 1(a) and 1(b). The de-warped image (DI) is photographed in a rectangular plane tangential to the cylinder. The perspective imaging equation of a DI can be written as follows:

$$\lambda p = K[R_{3\times3} \mid t_{3\times1}]P \tag{1}$$

where P is the homogeneous coordinate of a 3D point, p is the homogeneous coordinates of its 2D image point, R and t are rotation and translation with respect to the world coordinate system, and K is an upper-triangular matrix consisting of the intrinsic parameters, where

$$K = \begin{bmatrix} f_u & s & u \\ 0 & f_v & v \\ 0 & 0 & 1 \end{bmatrix}.$$

In most cases, the coordinate system selected by the panorama viewer to represent the pixel grids in DI is orthogonal, and thus $s=0$. In addition, the Panorama viewer usually de-warps the panorama to a square patch, and hence the aspect ratio $\frac{f_u}{f_v}=1$. Also, the tangential point is always set to be the center of the de-warped image in a panorama viewer, as shown in Figure 1(c). Therefore, the image center of DI, (u, v), is $(0, 0)$. If there are N pixels in a horizontal scan-line of DI, as shown in Figure 1(c), then the pixel resolution in DI is $du = 2f\tan(\frac{\theta}{2})/N$. Consequently,

$$f_u = du/f = 2\tan(\frac{\theta}{2})/N. \tag{2}$$

3.2. Extrinsic Parameters

Once the intrinsic parameters of a de-warped image are obtained, what we need is to compute the extrinsic parameters of it, i.e., the rotation and translation between the camera coordinate system and the Euclidean coordinate system drawn by the user. This problem is referred to as the *camera pose estimation* in the computer vision community. Given a trihedral with the angles between each pair of lines being 90°. By using the results shown in [6], we can compute the camera pose by solving a second-degree polynomial equation system. In this paper, we derive this result in another way. The detailed procedure of computation is shown in the following.

Given three lines joining at a point P_0 and orthogonal to each other, as shown in Figure 2. We select three control points, P_1, P_2, P_3, in the three lines, respectively. Assume that the homogeneous coordinates of their image points are p_0, p_1, p_2, p_3 respectively. Based on these three lines, we define an orthogonal object coordinate system that the origin is P_0, and the **X**, **Y**, and **Z** axes are defined to be along the directions from P_0 to P_1, P_0 to P_2, and P_0 to P_3, respectively. Let $\|P_0P_1\|=a$, $\|P_0P_2\|=b$, $\|P_0P_3\|=c$, then the coordinates of P_0, P_1, P_2, P_3 are $[0\ 0\ 0]^T$, $[a\ 0\ 0]^T$, $[0\ b\ 0]^T$, $[0\ 0\ c]^T$,

respectively. From (1), we can list the following four equations:

$$\lambda_0 p_0 = KRP_0 + Kt = Kt \quad (4)$$

$$\lambda_1 p_1 = KRP_1 + Kt \quad (5)$$

$$\lambda_2 p_2 = KRP_2 + Kt \quad (6)$$

$$\lambda_3 p_3 = KRP_3 + Kt \quad (7)$$

Since there always exists a scale factor which can not be computed, we set λ_0 =1 (i.e., the distance from the lens center to P_0 is the *unit length*) and it will not affect the camera pose estimation results. Hence, (4) becomes

$$Kt = p_0$$

From the above equation, we can solve the translation vector,

$$t = K^{-1} p_0 \quad (8)$$

where K is given in (3).

Substituting (8) to (5), (6), (7) and multiplying K^{-1} to the left side of (5), (6), (7), we can obtain the following equations:

$$K^{-1}(\lambda_1 p_1 - p_0) = RP_1 \quad (9)$$

$$K^{-1}(\lambda_2 p_2 - p_0) = RP_2 \quad (10)$$

$$K^{-1}(\lambda_3 p_3 - p_0) = RP_3 \quad (11)$$

Since the three vectors $\vec{P_0P_1}$, $\vec{P_0P_2}$, $\vec{P_0P_3}$ are orthogonal to each other. By computing the inner products of each of the two equations of (9), (10), and (11), we can obtain the following equations:

$$(\lambda_1 p_1 - p_0)^T K^{-T} K^{-1} (\lambda_2 p_2 - p_0) = 0 \quad (12)$$

$$(\lambda_2 p_2 - p_0)^T K^{-T} K^{-1} (\lambda_3 p_3 - p_0) = 0 \quad (13)$$

$$(\lambda_1 p_1 - p_0)^T K^{-T} K^{-1} (\lambda_3 p_3 - p_0) = 0 \quad (14)$$

where

$$K^{-T}K^{-1} = \begin{bmatrix} f_u^{-2} & 0 & 0 \\ 0 & f_u^{-2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

There are three unknowns, $\lambda_1, \lambda_2, \lambda_3$ in (12) - (14). Since the left side of (12)-(14) are bilinear forms, expanding (12)-(14) yields the following three bilinear equations:

$$a_{11}\lambda_1\lambda_2 + a_{12}\lambda_1 + a_{13}\lambda_2 + a_{14} = 0 \quad (16)$$

$$a_{21}\lambda_2\lambda_3 + a_{22}\lambda_2 + a_{23}\lambda_3 + a_{24} = 0 \quad (17)$$

$$a_{31}\lambda_1\lambda_3 + a_{32}\lambda_1 + a_{33}\lambda_3 + a_{34} = 0 \quad (18)$$

where a_{ij} are the coefficients computed from K and P_0, P_1, P_2, P_3 by expanding (12)-(14). The equations (17) and (18) yield that

$$\lambda_2 = \frac{-a_{23}\lambda_3 - a_{24}}{a_{21}\lambda_3 + a_{22}} \quad (19)$$

and

$$\lambda_1 = \frac{-a_{33}\lambda_3 - a_{34}}{a_{31}\lambda_3 + a_{32}} \quad (20)$$

By substituting (19) and (20) to (16), we can obtain a quadratic equation in terms of λ_3:

$$\begin{aligned} &a_{11}(a_{33}\lambda_3 + a_{34})(a_{23}\lambda_3 + a_{24}) \\ &- a_{12}(a_{33}\lambda_3 + a_{34})(a_{21}\lambda_3 + a_{22}) \\ &- a_{13}(a_{23}\lambda_3 + a_{24})(a_{31}\lambda_3 + a_{32}) \\ &+ a_{14}(a_{21}\lambda_3 + a_{22})(a_{31}\lambda_3 + a_{32}) = 0 \end{aligned} \quad (21)$$

Hence, λ_3 can be obtained by solving (21). After solving $\lambda_1, \lambda_2, \lambda_3$, the rotation matrix R can be obtained using (9) - (11) because the three columns of R are the unit vectors of $K^{-1}(\lambda_1 p_1 - p_0)$, $K^{-1}(\lambda_2 p_2 - p_0)$, $K^{-1}(\lambda_3 p_3 - p_0)$, respectively. In addition, *a, b, c* are the lengths of these three vectors, respectively.

4. Experimental Results

We have implemented a user interface which allows the users to draw an appearance of the exemplar shape and composite the virtual graphic objects in a geometrically-consistent way. Figure 3(a) shows an example of the three axes of a Euclidean coordinate system drawn by the users. In particular, a cuboid will appear in our interface if the user drawing makes the solution of (21) exist, as shown in Figure 3(b).

Some experimental results are shown in Figures 4 and 5 to clarify the effectiveness of our method. Notice that in both experiments we only have to estimate the camera parameters from a single de-warped view, the same parameters can then be used for other views while maintaining highly-convincing geometric consistencies of the generated composition views.

5. Summary

In summary, we developed a simple and intuitive approach in this paper for inserting geometrically consistent virtual 3D objects in a single panorama. To provide sufficient and not redundant geometrical constraints, what a user required to do is simply to draw the three axes of a 3D Euclidean coordinate system in the de-warped image according to his (or her)

perception to the scene. Then, our method allows the user to interactively place 3D graphic objects in arbitrary positions of the 3D world photographed in the panorama.

Acknowledgement

This work was supported in part by the National Science Council, Republic of China under Grant NSC-89-2218-E-001-004.

References

[1] C. S. Chen, et al. "Integrating Virtua +I ntegrat into Real Images for Augmented Reality," *Proceedings of ACM Symposium on Virtual Reality Software and Technology, VRST'98*, pp. 1-8, 1998.

[2] C. S. Chen, C. K. Yu, and Y. P. Hung, "New Calibration-free Approach for Augmented Reality Based on Parameterized Cuboid Structure," *Proceedings of International Conference on Computer Vision, ICCV'99*, Corfu, Greece, September 1999.

[3] D. F. Dementhon and L. S. Davis, "Exact and Approximate Solutions of the Perspective-Three-Point Problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, pp. 1100-1105, 1992.

[4] O. Faugeras, "From Geometr (From Geomt r y t Calculus: Theory and Applications of Three-Dimensional Vision," *Proceedings of IEEE and ATR Workshop on Computer Vision for Virtual Reality Based Human Communications, CVVRHC'98*, Bombay, India, pp. 52-71, January 1998.

[5] R. K. Lenz and R. Y. Tsai, "Techniques for Calibration of the Scale Factor and Image Center for High Accurate 3-D Machine Vision Metrology," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 713-719, 1988.

[6] Y. Wu, S. Iyengar, R. Jain, "A New Generalized Computational Framework for Finding Object Orientation Using Perspective Trihedral Angle Constraint," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 10, pp. 961-975, 1994.

[7] S. B. Kang and R. Szeliski, "3D Scene Data Recovery Using Omnidirectional Multibaseline Stereo," *International Journal of Computer Vision*, 25(2).

[8] K. N. Kutulakos, J. R. Vallino, "Calibration-Free Augmented Reality," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 4, pp. 3-20, 1998.

[9] H. Y. Shum, et al., "Interactive 3D Mode +I nt from Multiple Images Using Scene Regularities," *Proceedings of SMILE'98, Lecture Notes i Computer Science, Vol. 1506*, pp. 236-252, 1998.

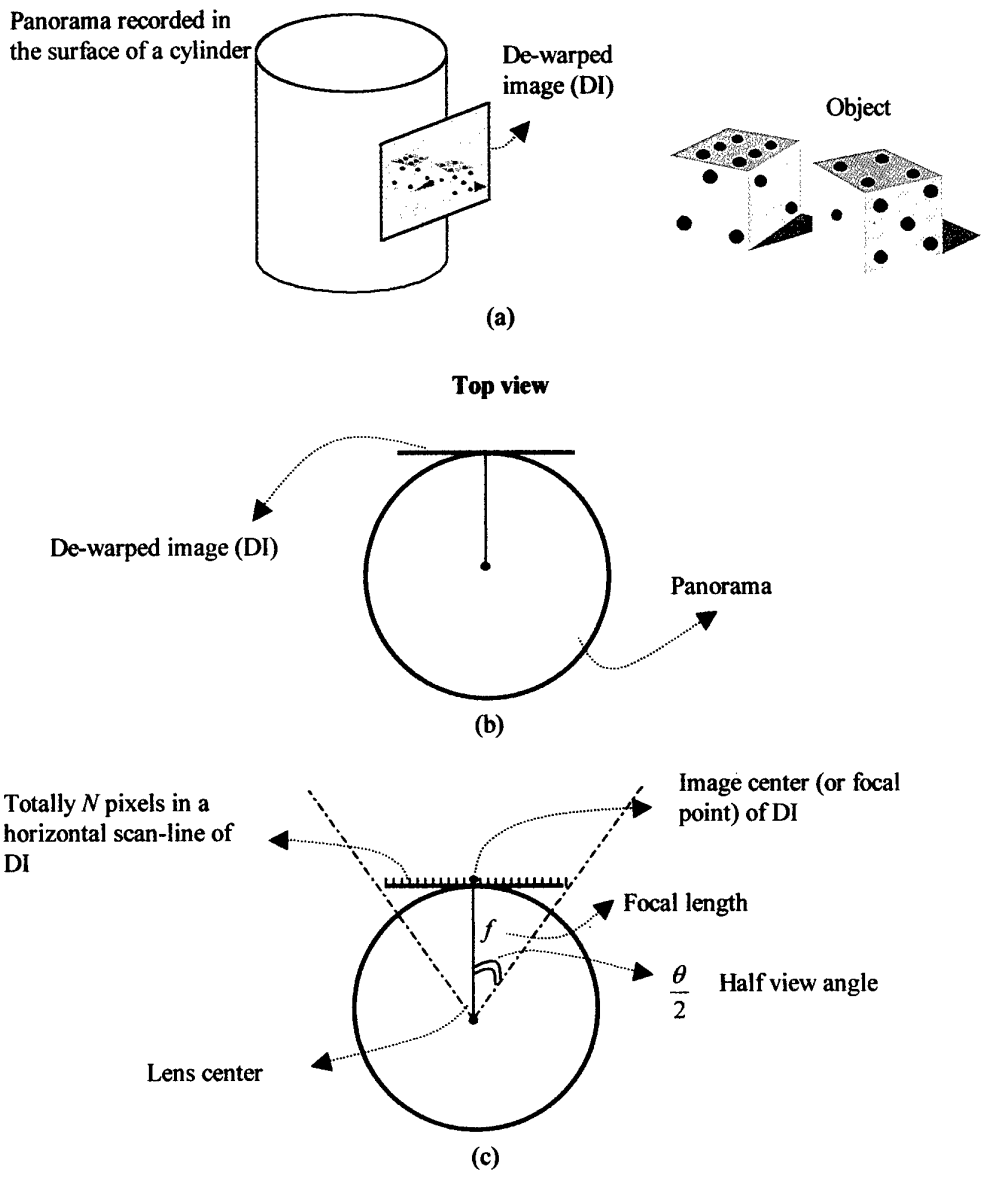
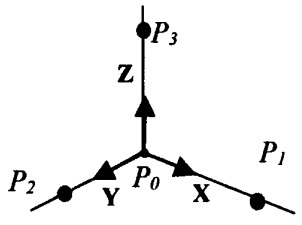
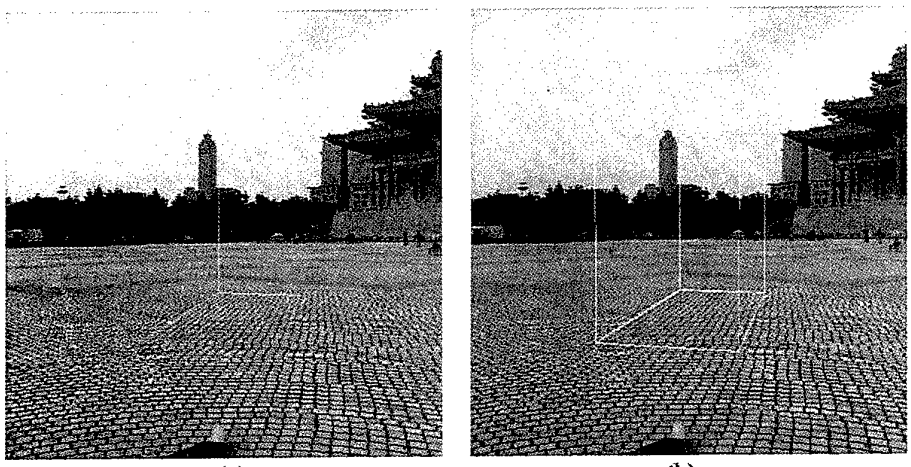


Figure 1. (a) A panorama recorded in the surface of a cylinder. A panorama viewer de-warps the panorama to a planar image (DI). (b) The top view of (a). (c) The intrinsic parameters of DI.



$P_0 = [0\ 0\ 0]^T$
$P_1 = [a\ 0\ 0]^T$
$P_2 = [0\ b\ 0]^T$
$P_3 = [0\ 0\ c]^T$

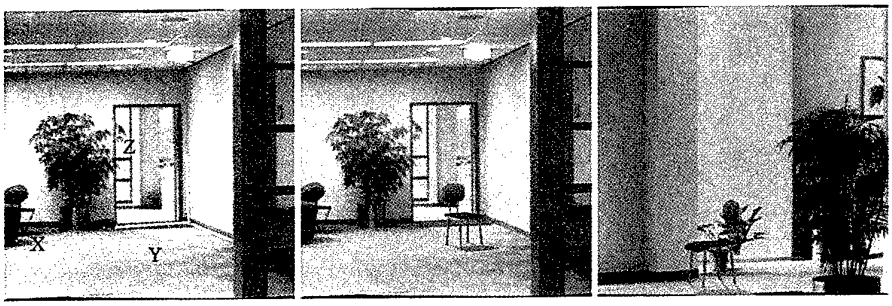
Figure 2. The object coordinate system defined in three orthogonal lines.



(a) (b)

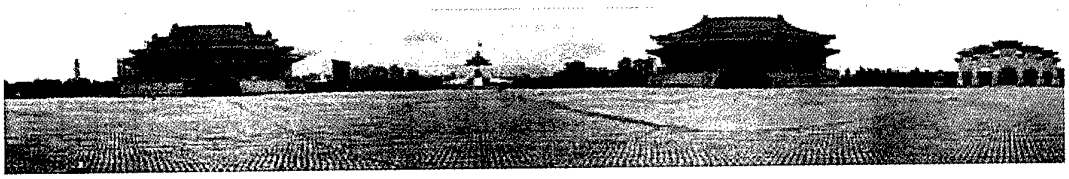
Figure 3. (a) An appearance of the three axes of a 3D Euclidean system drawn by a user. (b) A cuboid will appear in our interface if the user drawings allow the solutions to exist.

(a)

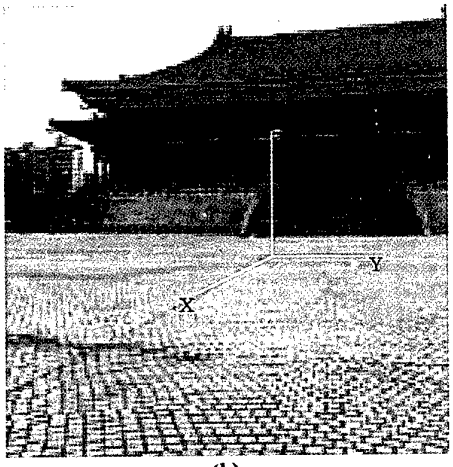


(b) (c) (d)

Figure 4. (a) A panorama. (b) Camera parameter estimation with the "front view" using the three axes of an Euclidean coordinate system drawn by human. Notice that such a coordinate system exists in the scene *explicitly*, and a user can easily identify it easily via his (or her) perception. A virtual chair is then inserted in (c) the "front view" and (d) the "side view" (using the same set of estimated camera parameters).

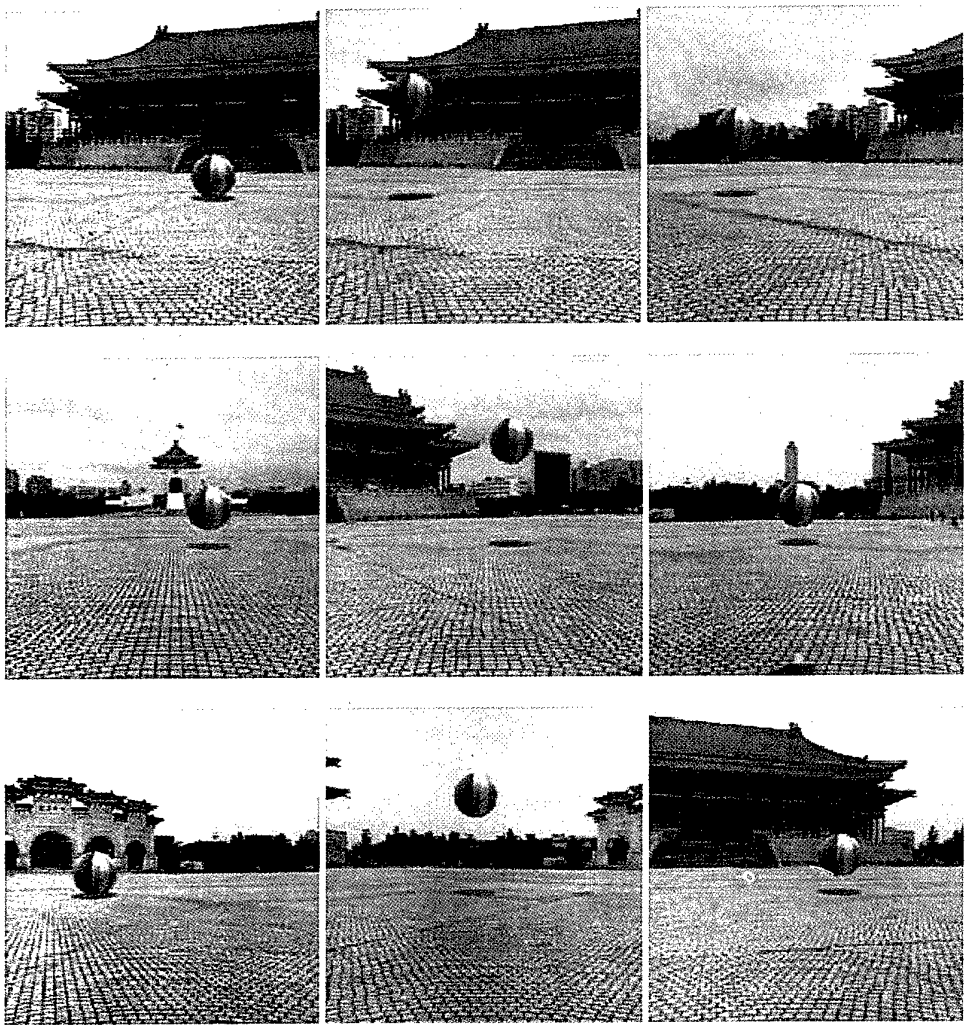


(a)



(b)

Figure 5. (a) A panorama of the Chung-Cheng Memorial Hall. (b) Three axes of the Euclidean coordinate system drawn by human for camera parameter estimation. Notice that although such a coordinate system does not exist explicitly in the scene, a user can still draw it *implicitly* via his (or her) perception.



(c)

Figure 5 (continue). (c) The insertion of an animation of a bouncing ball in the panorama. Also, the same set of camera parameters estimated from the Euclidean coordinate system drawn in Figure (b) is used for all views.

High Resolution Displays and Roadmap *

Darrel G. Hopper

Air Force Research Laboratory
Wright-Patterson AFB, OH 45433-7022 USA
darrel.hopper@wpafb.af.mil

Abstract

Synthetic vision systems for artificial reality and tele-presence remain far short of the resolution of the human visual system. Current electronic display systems support 20/20 visual acuity or less, yet human vision is dramatically better than the 20/20 measure implies. Compelling applications and products will require ever more resolution, grayscale, etc. Current technology must grow from 1 megapixel devices common in the year 2000 to 10-100 megapixels devices by 2010-2020 to support, eventually, systems with aggregate resolution well over 1 gigapixel. A vision of displays for the next decade and century will be provided along with a roadmap for high resolution display devices.

Key Words: Electronic Displays, Synthetic Vision

1. Introduction

Artificial reality and tele-existence systems are limited by display technology. Advances in displays and digital television are now poised to enable a 10-to 100-fold growth in capability (e.g. resolution) by 2010. Such improved displays will pay for themselves via increased productivity in work, home, and entertainment applications. Simulators and trainers might leverage this digital display trend to produce synthetic vision systems at 20/20 resolution (170 megapixel needed versus 16 megapixel at present). Such 20/20 simulators could save fuel and increase safety by reducing training needed in real vehicles, increase the effectiveness of pre-mission rehearsal, and enable realistic human factors research. Uninhabited vehicle interfaces will require 10-100X more resolution just to keep up with advanced sensors (video at 25 megapixels per frame) and databases (100 megapixel portions of 8-30 gigapixel scientific and terrain domains). Knowledge walls and complete audio-visual environments (CAVEs) for control rooms and education will prepare the way for in-vehicle hectomegapixel display systems with 200 megapixels like the Ford 24/7 concept car. Entertainment applications include home IMAX. This paper reviews electronic display trends enabling synthetic vision concepts. A vision of displays over the next 10, 20, and 100 years is presented.

2. Synthetic Vision Concepts

Displays have crossed the megapixel threshold. The human visual system (HVS), however, is capable of processing one gigapixel color images of full motion video. A substantial closing this 1000X gap will dramatically increase productivity.

2.1 Need

The common "20/20" met 20/20" rat met ric for visua arc seconds subtended at the pupil) is defined for a room maintained at a very dim ambient illumination (e.g. 100 lx). In Nature the range of illumination is many orders of magnitude higher (0.01-108,000 lx). In the real world the luminance contrast is usually sufficient to resolve objects far less than 50 arc seconds. For example, stars in the night sky subtend perhaps as small as 5 arc seconds or less, yet people see stars. Similarly, glint from a highly reflective surface is readily visible, but often subtends < 20-25 arc second. Also, 20/20 is defined for black/white only and ignores color, 3D, and motion as image resolving features of human vision.[1]

Humans move in a 3D world with images arriving from all directions. These images are continually being integrated as one moves about and looks in any direction at will. Thus, an ideal display would cover the full 4π sr of a natural world scene; this solid angle is equivalent to over 1.3 billion two-dimensional picture elements (pixels). Adding a third dimension leads to volume element (voxel) resolutions up to 22 trillion voxels. Resolution comparisons for

* Citation: D.G. Hopper, "High Resolution Displays *Hgh Re solutio *Proceedings of the 10th International Conference on Artificial Reality and Tele-existence (ICAT'2000),"* 25-27 October 2000 in Taipei, Taiwan.

4π sr are provided in Table I (pixels) and Table II (voxels).

It is true that human visual acuity in the foregoing discussion refers to an instantaneous attention angle of about 2 arc degrees. However, it is also true that this acuity (and far better, down to 0.5 arc second for verier acuity) actually exists in real world scenes over 4π sr. Also, the high rate of eye scan and head movement, combined with the sensitivity of peripheral vision to motion, requires full image be present continuously at full visual acuity over 4π sr, ideally, just as in Nature.

Table I. Number of resolvable pixels in 4π steradians.

Acuity	Comment	Pixels
50 arc seconds	20/20 vision	213,860,000
25 arc seconds	Glint and	855,450,000
20 arc seconds	Stars *	1,336,700,000

* Real world luminance & chromaticity contrast effect.

Table II. Number of resolvable voxels in 4π sr.

Depth Layers	2D Acuity	Voxels (billions)
10	50 arc seconds	2
	20 arc seconds	13
100	50 arc seconds	21
	20 arc seconds	134
1000	50 arc seconds	214
	20 arc seconds	1,337
	5 arc seconds	21,386

* Holodeck of Starship Enterprise $\approx$ 1 trillion voxels.

High definition digital television will be but a first phase of efforts to close, somewhat, the gap between fielded displays and the HVS capability in Tables I and II. Business, entertainment, education, advertising, training, and other applications will drive the creation of rooms in which every surface (walls, furnishing) have embedded displays. Pixel rooms will take the form of walls covered entirely with flat panel displays (FPD); covering all walls creates a CAVE or FPD igloo for immersive systems. Also, vehicles—including cars, trains, and aircraft—often must be operated under conditions in which the outside world is not clearly visible due to conditions of night or bad weather. A view of nothing might be dramatically improved by providing larger area and synthetic vision display systems under these conditions.

2.2 Super-Panoramic Cockpit (SPC)

A program of studies conducted by the Air Force Research Laboratory has demonstrated the productivity improvements available when one begins to deal with the display technology challenge identified above. The approach in this program, entitled "Panoramic Cockpit Control and Display System (PCCADS)," is to provide a pilot with large area displays and a helmet-mounted off-axis target-acquisition weapon-targeting system. There were two projects, one focused near term, one far.[2]

The PCCADS 2000 cockpit was designed to be realizable with 1995 technology with production by 2000 and featured a 25 cm (10 in.) square tactical situation display and two 15 cm (6 in.) square secondary multifunction displays on either side. All displays were full color capable with a total area of 1110 cm^2 (172 in^2). The test mission was for an F-15E. A 28% increase in exchange ratio was achieved versus the standard F-15E cockpit. An 18% increase was observed for the addition of helmet cueing to the F-15E baseline cockpit. Coupling this large display with a helmet-mounted cueing system for off axis target acquisition resulted in a 45% increase. The F-22A Raptor will realize the PCCADS 2000 concept in a production cockpit (video wall comprising six flat panel AMLCDs with an aggregate resolution of 1.35 megapixels at 5-bit greyscale in 1290 cm^2 (201 $in.^2$) plus an HMD add-on. Beyond the PCCADS 2000 cockpit was PCCADS.[2] PCCADS was designed to be realizable with beyond 2000 technology and featured a 2000 cm^2 (300 in^2) head down display system which appears seamless, but which, in fact, must be implemented in a physically redundant fashion to meet fail-soft and reliability requirements.

This PCCADS research demonstrated the payoff in increased situational awareness from integrating all information and displaying it to the pilot on one very large display format. The PCCADS cockpit, plus curved "wing" Ywi rg" di d spl ays f or-machine interface and a closable inner curtain, is illustrated in Figure 1. This concept, the super-panoramic cockpit (SPC), has features which might be explored over the next 5-20 years to enable closed cockpit operations. A closable curtain gives way to a flexible canopy display in the far term. Stowable flat panel displays or projection screens are deployable either side of the head-up display.

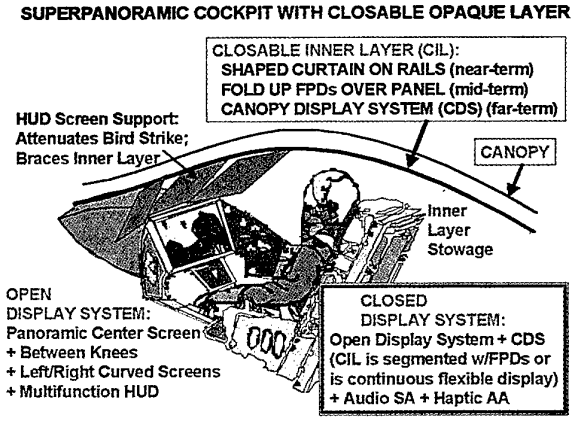


Figure 1. Super-panoramic cockpit (SPC) with closable curtain or flexible display, plus fold-up FPD screens.

2.3 Synthetic Vision Perspectives

There are two complementary approaches to the design of synthetic vision sysems: outside-looking-in (OLI) and inside-looking-out (ILO). In the OLI approach, the viewer perceives himself to be located outside looking in on the world presented on the display. In the ILO approach, the viewer perceives himself to be located inside the displayed world looking out at it. Large field of view, 120 x 60° or more, is required for one to "think" one is actually immersed in the world presented on the display(s). The ILO approach is achieved today by the real world itself as viewed via real immersion or real windows in vehicles. For aircraft the windows are often in the form of a transparent canopy: the pilot is centered in a real world with all display elements coming from real world phenomena. Today's fielded desktop and auto/cockpit head down displays (HDDs) represents the OLI approach. Significant development in display technology is required to implement either the large area OLI or, eventually, the ILO approach.

The head-mounted version of either the OLI or the ILO approach has yet to catch on despite the great hype. In all human applications people exhibit a strong, visceral aversion to head-mounted solutions. Even companies developing head mounted displays (HMD) to replace computer monitors and cell phone displays do not yet use their own product in their own office. This leads us to a rule that applies to HMDs:

Rule for Head-Mounted Equipment (HME):
People will wear HME only if they will die if they don't

Soldiers wear helmets to decrease the chance of death. Individuals who must wear corrective lenses do so in order to live (e.g. drive cars safely) and many opt for contacts or laser eye surgery to remove the need to wear glasses. People will wear HMDs as a necessity or a novelty, if at all, and will ***not*** wear them *in lieu* of displays elsewhere (walls, television, computers, monitors, cell phones, personal digital assistants, etc.).

Different parts of the display system will employ small, medium and large area direct-view visual displays. Niche applications like military must leverage the commercial market to the maximal extent possible. The creation of a display technology, even after key inventions have been made, takes 3 to 20 years for manufacturing process development followed by integration and ruggedization for military applications.

Research directed at the creation of display technology required to support both the OLI and ILO approaches is reviewed in subsequent sections.

2.4 Cockpit Vision

Fieldable cockpit display technology in 2000 is represented by the B-777 commercial transport, F-22A fighter, and RAH-66 helicopter. Each pilot has 650-1300 cm^2 (100-200 in^2) comprising 2 to 6 color multifunction displays (MFD).

The cockpit vision in Figure 1 comprises a 4000 cm^2 (600 in^2) super-panoramic direct view head-down display (HDD) system coupled with a simple helmet display for off-boresight cueing of smart munitions. The HUD is still present as a ballistic munitions targeting reticule unambiguously and accurately aligned with the airframe. Deployable displays may be integrated either side of the HUD. The cockpit canopy may be turned opaque via a simple shade or a complex display shell. A world view is created in the closed cockpit mode from on-board/off-board digital data bases, the on-board sensor suite, and the off-board sensor suite.

The 2020 vision is an encapsulated cockpit as illustrated in Figure 2. The pilot may have no windows. The cabin may be a self-contained spheroid embedded within the aircraft or, possibly, elsewhere. This display system might be much like that of a present-day trainer/simulator—only far, far better. The system will be color and high resolution. The pilot has the option of retaining or selectively removing real world visual effects of weather and night. The 2020 vision includes actual views from not only ownship, but also from a variety of other platforms via cameras, data bases, and data links. The capsule is a node in a digital network.

Figure 2. Encapsulated cockpit realized as combination of direct-view, projection and head-mounted displays.

2.5 Display Vision

Our goal is to create a display technology base to enable the design of panoramic and immersive cockpits. The opportunity to do so arises from significant investments by both the commercial and government sectors "to make the impossVto make t he i for an ever-expanding global industry of visual digital applications. In this endeavor we are the beneficiaries of the information age and the insatiable market for better and more visual communication and entertainment devices. Our strategy is to pursue multiple technological approaches: revolutionary new display technologies, groupings (arrays,seamless tiling) of flat panel displays, and projectors. Haptic, auditory, and olfactory displays will also merit consideration. A vision for the evolution of displays is discussed in more detail by Hopper.[3]

2.6 Performance Specification

Visual displays must be readable in a variety of situations. Performance specifications range from detailed (dozens of parameters) to summary (3-4 aggregate metrics). Some key specifics follow.

2.6.1 *Large area with high resolution*
Display module sizes must measure at least 25 cm (10 in) up to more than 150 cm (60 in) diagonal. Pixel densities for display screens places 60 cm (24 in.) from the viewer must be at least 32 cm^{-1} (80 in^{-1}) up to 100 cm^{-1} (240 in^{-1}). Several modules or sizes can be grouped together as necessary to achieve the total aggregate display area required up to e.g. 25 m (100 ft) for IMAX or NASDAQ.

2.6.2 *Sunlight readable*
Persons with normal vision must be able to read the display in both direct and occulting sunlight. In each case the sun is not attenuated. Direct sunlight means the sun shines on the display; occulting, into the viewer's eye. The goal inherent in this requirement is usually expressed in terms of the luminance (light intensity) emitted and contrast maintained by the display for a specified illumination condition. Full daylight is taken to be an illuminance of either (a) 108000 lx (10000 fc) directly incident on the display with luminance of 1710 cd m^{-2} (500 fL) incident at the specular angle with respect to the test viewing angle, or (b) 21500 lx (2000 fc) illuminance, with 6850 cd (2000 fL) luminance at specular. The contrast ratio must be at least 4.66:1 (5 grayshades) under the highest luminance condition and 10:1 under 40 % of

over 3,400 cd m^{-2} (1000 fL) is required.

2.6.3 *Variable brightness, grayscale, night vision*
Viewers must be able to adjust the brightness to be viewable in a continuum of over six orders of magnitude of ambient illuminance from 108000 lx down to 0.11 lx (10000 fc to 0.01 fc). The electronics to accomplish this dimming ratio (0.01 to 1000 fL) is half the cost of a sunlight readable display. Eight colors (3 bits) often suffice for symbology; color graphics and video systems ideally require 48 bits. Military applications must be compatible with night vision systems.

2.6.4 *Environmental*
There are two enviromental aspects. First, new displays must take the impact on the environment into account: from the mining of raw materials, through manufacture, during use, and ending with disposal. Second, the conditions during use—temperature extremes, shock, vibration, humidity, electronic interference, dust, kicking, etc.—must be considered in display design.

2.6.5 *Aggregated Metrics*
Aggregate metrics are required to describe displays to communities of widely differing backgrounds. Such metrics include: life cycle cost (LCC) for several years of operation (e.g. 10 yr.); power efficiency in terms of efficacy in lm/W; and visual information thrust in Mb/s. Thus, LCC is needed to show a return on investment (ROI) of over 3:1 to justify investment; experience for cockpit FPDs is an ROI of 13:1, which justifies insertion of FPDs in place of electromechanical and cathode ray tube displays. Power efficiency is vital in all weight-sensitive applications. Visual information thrust (VIT) in bits/s was introduced by Hopper.[4] The definition of VIT with examples is provided in Figure 3.

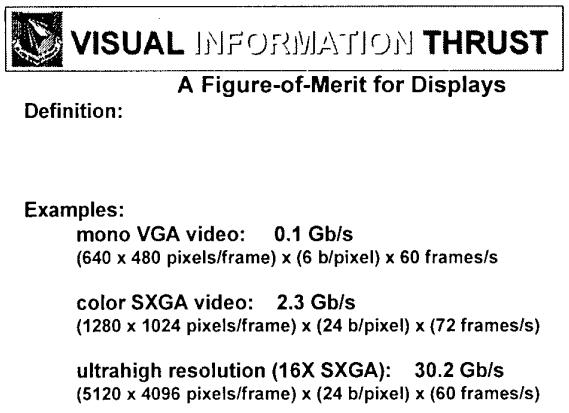


Figure 3. Visual Information Thrust: an aggregate metric of what a display is capable of providing.

3. Technological Approaches

Technological approaches to large area, panoramic, and immersive displays include array, tiling, and projection. Direct view arrays are known commercially by such names as video wall. Tiling retains several individual displays, but removes the spaces between them. Tiling can be accomplished by several methods: juxtiposition; circuit pasting; optical stitching (appears seamless). Flexible displays produced by roll-to-roll web processing may affordably provide seamless display screens of very large size in the far term. The status of display technology development was reviewed recently at the U.S. DoD Defense Advanced Research Projects Agency (DARPA) Information Exchange Conference on High Definition Systems.[5] Integration of display technology into aerospace and defense applications is documented in a series of widely available conference proceedings, comprising almost 3000 pages, published in seven volumes by the International Optical Engineering Society edited by Hopper. [6]

3.1 Direct View Displays

3.1.1 *Cathode tubes and electromechanical*
Avionic CRTs and electromechanical (EM) instruments have problems with reliability, availability, sunlight readability, and scalability. Also, CRTs and EM cannot be scaled to 2000 cm^2 and larger areas with space, weight, and power in most applications. Research in areas like flat-CRTs may provide new options, however.

3.1.2 *Flat Panel Active Matrix Liquid Crystal Display*
The active matrix liquid crystal display (AMLCD) is the only flat panel display technology currently capable of high brightness (sunlight readable) and full color. It is the preferred display technology for all applications. Research to invent the AMLCD began about 1969. The first commercial product successes—hand held TV and small cockpit displays—occurred about 1988 when the pixel density reached 80/in. Sizes range from 8 to 700 mm (0.25 to 30 in.). Tiled versions go up to 1m (40 in.).

3.1.3 *Flat panel: Thin-Film Electroluminescent*
Research started in 1994 has created an additional FPD technology for avionics and military applications that is sometimes a better choice than an AMLCD. The yellow thin-film electroluminescent (TFEL) passive matrix addressed FPD has been developed for monochrome video in sizes up to 20.3 x 11.4 cm (8 x 4.5 in.).

3.1.4 *Flat Panel: Field Emission Display (FED)*
A field emission display (FED) is another possibility. Performance demonstrated to date will ***not*** support applications. Flashover problems associated with high voltages (5-12 kV across a small gap of <1mm) have prevented success. The FED is still a technology in

search of a birth date even after 8 years of strong effort.

3.1.5 *Flat Panel: Organic Light Emitting Diode*
Over the past five years yet another flat panel display technology has begun to appear in products: active matrix organic light emitting diode (AMOLED) display. Initial low information content applications are now available for car radios. Cell phones. digital assistants, cameras, and avionics versions are in development.

3.1.6 *Flexible Displays*
A revolution in display technology has begun. Displays fabricated on glass may eventually be replaced by displays fabricated from plastic. Substrates might be expanded to flexible thin sheets of steel. The dream of roll-up displays and less weight/power is being pursued. Research has demonstrated an ability to fabricate the thin-film transister (TFT) electronic circuitry for AMLCDs or AMOLEDs at process temperatures as low as 75 °C. A second approach to flexible displays is the "optical lattice" in which light is generated at the edge of the screen (infrared or visible) and piped via optical waveguide structures to pixels.

3.1.7 *Printable Displays*
Large, flexible, flat displays will require a second revolution: roll-to-roll (so-called "web" equipment, as in newspaper production) with cutting of desired displays sizes from the "cloth" produced in the display production line. In 1998 Polaroid successfully demonstrated roll-to-roll production of passive matrix liquid crystal display cells. Transition of other display technologies to web processing is underway.

3.2 Tiling

The vision for the fully immersable concept, such as Figure 2, will require a significant expansion of the current state of the art in display technology. A complementary alternative is to move the current discrete displays so close together that one perceives one large display rather than several discrete displays. In this way it becomes possible to present a seamless panoramic display across the tiled array yet retain physical redundancy to maintain reliability.

3.2.1 *Justapositioning.*
The individual displays could just be placed next to one another with the viewer tolerating the clearly visible gaps or seams. The 1990 state of the art was represented by the 6144 x 2048 pixel, 152 x 51 cm (60 x 20 in.) prototype built from three 2K x 2K color CRTs by at MIT. Air Force satellite constellation management uses seven of these CRTs, for a total resolution of 29 Mpixels. The NewsMuseum in Arlington VA has 90 VGA (640 x 480) projectors tiled on a wall, a total of 28 Mpixels. The Air Force Research Laboratory warfighter training team in Mesa AZ has produced a simulator using eight screens each rear projected by a 1600 x 1200 projector, a total resolution of 15 Mpixels. Seamless tiling by cutting AMLCD edges was shown in late 2000.

3.2.2 *Optical stitching*
Stitching involves optical means to make the physical display structure comprising the discrete displays appear to be one large display by optical schemes. One might imagine display tiles mounted on the back side of a display screen with each magnified optically to fill its portion of the big image seen by the viewer. Microlens arrays or holographic sheets might be used. Curved and wall filling displays having resolution of 10 Mpixels or more might be made in this fashion. Sarnoff Corp. is pursuing such an approach to multi-megapixel displays for immersive C4I and entertainment applications. The closable screen depicted inside of the transparent bubble canopy in Figure 1 becomes a segmented hard shell with displays in each portion. Macro-optical-coupling makes a segmented display array appear to be one large display. One macro-optical approach is to tessellate a spherical surface into areas the shape of pentagons and hexagons. Then a flat large FPD is projected a few inches to each polygon (curved sphere segment) via a space-filling fresnel optic. The viewer would see an apparently seamless, curved large solid angle display with a total resolution of the FPD used times the number of tessellation segments times a fill factor (a non-rectangular image inside a rectangular flat panel does not take up all of the addressable pixels).

4. Projection

There are several projection methods: cathode ray tube (CRT), liquid crystal light valve (LCLV), microelectromechanical (MEMS) devices, p-Si and x-Si miniature AMLCDs, and solid state laser display (SSLD). Projector light power output is given in watts at the aperture or ANSI lumens at the screen; one must specify a projection solid angle to compute luminance or a screen size to compute illuminance. More than 1 W per color leaving the projector aperture is required. The light source for all SLM-based projectors (MEMS, AMLCD) is presently an arc lamp; more compact, bright, power efficient, and reliable solid state sources are being developed (inorganic LEDs and visible solid state lasers). Screens also need improvement.

4.1 CRT and LCLV Projectors

Cathode ray tubes and light valves have been used in projectors for some time. The CRT displays are of much lower quality than the LCLV. An example is the Hughes Series 300 LCLV system, based on an optically written a-Si photoconductor, which projects 2500 lm at video rate with good contrast for a price of $150,000; additional limitations include low frame rates and

thermal sensitivity. The resolution provided is so low that 20 foot high letters on a carrier tower cannot be read in a simulator—requiring the early curriculum transition to burning jet fuel.

4.2 MEMS Projectors

Microelectromechanical (MEMS) devices can be fabricated that serve as spatial light modulators (SLM) in a projector light engine of a visual display system.

4.2.1 *Digital Micromirror Device Projector*
The Texas Instruments digital micromirror device (DMD) presents a near term practical alternative to both direct view CRTs and other projection technologies for applications. A depth of about 10 in. behind the viewable screen surface is required. The prototype color high definition 1920 x 1080 pixel system (17 micron pixel pitch) incorporating three 3.2 x 1.9 cm (1.25 x 0.75 in.) DMD chips with a 16:9 aspect ratio, a 150:1 contrast ratio, and projecting >1000 lm to the screen was developed over the period 1990-1995. As of 1999 the TI "Digital Light Processing (DLP)" light engines based on the DMD have re-defined the state of the art in commercial presentation projector market.

4.2.2 *Defractive Grating Light Valve Projector*
The diffractive grating light valve (GLV) linear spatial light modulator being developed by Silicon Light Machines, Inc. (SLMI) is a different type of MEMS device. Light is modulated by micrograting diffraction pixels rather than by moving micromirror pixels. SLMI is currently attempting to tile four 1024 x 1 pixel devices and use scanning to develop projector with resolution of at 5120 x 4096 (21 Mpixels).

4.3 AMLCD Projectors

Liquid crystal displays can be used in projection as well as direct view. The Hughes HighBright™ display technology is based on three a-Si AMLCDs operating in a color projector design; the breadboard system is sunlight readable and has an active display area of 16 x 16 cm (6.25 x 6.25 in.). This technology has been commercialized in a banking application (automatic teller machine). Commercial projectors with p-Si AMLCD devices about 2 x 2 in. compete with DMD for professional presentation markets. A new version, reflective miniature x-Si AMLCD on silicon (LCOS), is due to arrive in projection products in 2001.

4.4 Laser Projectors

Lasers may become the display per se when coupled with a modulator. Laser light is coherent and colors are fully saturated. The coherency translates to a unique feature of direct-modulation laser displays: virtually infinite depth of focus. This means that the image is always in focus, even when displayed on curved or domed screens, as in a custom installation inside a cockpit or simulator. The pure colors provide a wide color spectrum capability. The color range is larger than CRT or LCD based systems. Furthermore, a laser display has better legibility: objects which are fuzzy in a CRT or LCD system are clear in a laser projection of the same image size: luminance and chromaticity contrasts are simultaneously much, much greater. Laser display technological approaches include discrete lasers (both gas, solid state), laser arrays (solid state), and a CRT having semiconductor materials in place of phosphors. The various solid state approaches vary in the pumping mechanism. Projects to make an affordable SXGA solid state laser projector is now underway.[7]

5. Miniature Displays

Research is also underway to establish high resolution miniature displays. The term "miniature display" is a commercial definition for displays whose image must be magnified for viewing. A 12 mm VGA monochrome yellow active matrix electroluminescent (AMEL) display has been developed. The same display has found a direct view application in aircraft annunciator panels as smart, reprogrammable buttons that display diagonal lines smoothly. A miniature 25 mm SXGA monochrome AMEL is completing development and work continues on color. Miniature 12 mm monochrome green CRTs are the baseline technology for the helmet mounted cueing system envisioned in PCCADS. A replacement technology, a miniature 12 mm SXGA monochrome green AMOLED, is being developed as a miniature flat panel display replacement for the miniature CRT. A miniature 25 mm AMLCD at SXGA resolution is being perfected for helicopter helmets. Virtual retinal display (VRD) technology is being developed; color VGA has been demonstrated. Presently, VRD requires too much power and is too bulky for commercialization.

6. System Considerations

Compact supercomputers, known as a multimedia processors, are necessary to drive large area electronic display systems. Also, the functions and screen formats must be determined for this new class of ultrahigh resolution displays.

6.1 Graphics, Video, Information Processors

A supercomputer in a shoebox is required to drive concepts such as depicted in Figures 1 and 2. All information must be integrated in standard formats and graphic generated for the large area of high resolution

display surface(s). This processing capability is a narrow-to-wide band processing problem—the inverse of the wide-to-narrow band type of processing problem at radar and electro-optical imaging sensors. The needed improvements in processors may be anticipated based on current, commercially-driven trends.

6.2 Display Format

Once pixels are available they must be filled based on user-in-the-loop studies and crew station integration concept development efforts. Indeed, the creation of the ability to light up more megapixels and the consideration of what to put in them is a synergistic problem to be addressed jointly by the hardware and humanware engineering communities. Transition from monochrome to color pictorial formats were found to provide intuitive presentation and, thereby, a potential reduction in pilot workload. Similarly, a large display is critical to integrate all information in a meaningful, legible way. Future electronic multifunctional displays must deliver both color and large area to support the display format requirements. One day the entire instrument panel of cars and aircraft, and the tops of desks, may consist of one display surface where both pictorial, and alpha-numeric formats will be displayed.

7. Roadmap

The 2.1 megapixel devices needed for high definition (digital) television (HDTV) will come to define the mass market by 2010.[8] The TV standard beyond HDTV may not come until about 2070 with mass production by 2100. The resolution for the 21st century TV standard (HDTV at 2,073,600 pixels) is about 6.75X greater than 20th century TV. Thus, the TV standard for the 22nd century should exceed 15 megapixels.

Rapid growth in resolution has begun. Creation of 20-30 megapixel displays for simulators, sandboxes, cinema, home and office will involve revolutions leading to pixel-surfaces for furniture, walls, and rooms by 2020. Maps for sandboxes require 33 megapixels/m^2. Flexible and printable display technologies, on which research has just begun, will enable wallpaper-thin displays. Many should be able to afford a home "pixel room" comprising 214 megapixels in six sides, by 2100.

Other challenges must be met in order to increase resolution. Specific power density (W/kg) for mobile power sources needs to go up a factor of 10 by 2010 and 100 by 2100. Light generation needs to be made 10-100X more efficient; efficacy in mass production displays should increase from about 4 lm/W in 2000 to 40 lm/W solid state light sources by 2100. Electronics must speed up too: a 30 megapixel device at 48 Hz requires a digital interface of 34.56 Gb/s and storage capacity on the order of 1 petabyte. Image generation processors must be distributed to pixels and segments.

Table III summarizes this roadmap and vision for displays of the future.

Table III. Predicted resolution for display devices. Resolution is expressed in megapixels per device.

Year	Market Classification (end customer sales)		
	Exotic (1-100 units)	Niche (1-10k units)	Consumer (.1-10m+)
2000	5.4 for computer	2, digital cinema	1.9 for PC
2001	1.3 for cockpit	0.3 for cockpit	2.1,HDTV
2010	30 for IMAX	20, web PCTV	4 WCTV
2020	30 for cockpit	20 for simulator	8 WCTV
2100	855 for simulator	214 for home	15,WCTV
3000	Immersive display room: 1.3 gigapixel system		

8. Conclusions

The advantages of a large area display system were demonstrated in the Panoramic Cockpit Control and Display System program, a joint research effort of hardware and humanware engineers. The key objective result was a 45% increase in pilot combat effectiveness, which translates to a 31% reduction in the number of aircraft and pilots needed for a given mission. Clearly, large area display systems increase productivity.

Flat panel displays and solid state laser and other projectors present what is, perhaps, the most attractive alternative for achieving panoramic cockpit display technology by 2010. They are light in weight, low in power requirements, and can meet all environmental requirements. Furthermore, FPDs and projectors can scale from sizes used in instrument panels up to synthetic out-the-windows. Total cockpit resolutions of 4-10 megapixels are possible in such near term cockpits. Current cockpits and desktops have just crossed the 1.3 megapixel mark. Thus, a realistic challenge in the near term is an increase of total resolution of 3x to 8x times over the next 10 years. Pixel density needs to increase beyond 200/in. The AMLCD technology has achieved this in the latest IBM announcement in September 2000 of a 9 megapixel, 22-in. display with 200 pixels/in.

By 2020 a variety of improved projector and direct view technologies will be available to build HDD and encapsulated cockpit display systems. Individual displays will be >16 million (e.g. 4096 x 4096) color pixels in 2000 cm^2 (300 in^2) and contoured to fit the curved surfaces of the control panel and inner canopy. Several displays will be tiled to achieve larger display areas.

Flexible and printable displays may lead in the far term to a closable cockpit capsule with pixels on the inside as shown in the immersive cockpit concept. Alternatively projection technology may be miniaturized, or continued evolution of optical tiling of direct-view FPDs may provide the solution. The 210 Mpixel immersive cockpit concept depicted in Figure 2 might become a fielded reality by 2050.

References

1. D. G. Hopper, "*Invited Paper* 21st Century Aerospace Defense Displays," in Society for Information Display (SID) Symposium Technical Digest, Session 29 "Applications: Airborne Displays," 1999, paper 29.1, pp. 414-417 (1999).

2. D. G. Hopper, "Panoramic Cockpit Display," published in *Advanced Aircraft Interfaces: The Machine Side of the Man-Machine Interface*, AGARD CP-521, 1992, pp 9-1 to 9-25. Conference Proceedings of the 63rd Avionics Panel Meeting/Symposium held in Madrid, Spain, 18-22 May 1992. Published by the NATO Advisory Group for Aerospace Research and Development (AGARD) Avionics Panel (AVP).

3. Darrel G. Hopper, "1000 X difference between current displays and capability of human visual system:

payoff potential for affordable defense systems," in *Cockpit Displays VII: Displays for Defense Applications,* Darrel G. Hopper, Editor, Proc. SPIE 4022, 378-389 (2000).

4. (a) D. G. Hopper, "Performance specification methodology: introduction and application to displays," in *Cockpit Displays V: Displays for Defense Applications,* Darrel G. Hopper, Editor, Proc. SPIE 3363, pp 33-46 (1998); (b) D. G. Hopper, "Performance specifications: the nearly impossible versus the merely difficult," in *Cockpit Displays VII: Displays for Defense Applications,* Darrel G. Hopper, Editor, Proc. SPIE 4022, paper 15 (2000).

5. *High Definition Systems Program Presentation Summaries",* DARPA High Definition display technology Information Exchange Conference, 21-24 Mar 1999, 1999. Direct requests to DARPA/ETO, 3701 N. Fairfax Drive, Arlington VA 22203-1714.

6. *Cockpit Displays,* Darrel G. Hopper, Editor,
(a) Proc. SPIE 2219 (1994) 408 pp.;
(b) Proc. SPIE 2462 (1995) 350 pp.;
(c) Proc. SPIE 2734 (1996) 312 pp.;
(d) Proc. SPIE 3057 (1997) 608 pp.;
(e) Proc. SPIE 3363 (1998) 490 pp.;
(f) Proc. SPIE 3690 (1999) 380 pp.;
(g) Proc. SPIE 4022 (2000) 442 pp.

7. K. J. Snell, D. Lee, B. Pati, and P. F. Moulton, "RGB optical parametric oscillator source for compact laser projection displays," *ibid*, 285-289 (1999).

8. "HDTV: You're not going to like this picture—Technical snafus continue to slow its growth," Business Week, October 25, 1999, p. 50.

ICAT 2000

Haptic Interface with 7 DOF Using 8 Strings : SPIDAR-G

Seahak Kim
seahak@pi.titech.ac.jp

Masahiro Ishii
mishii@pi.titech.ac.jp

Yasuharu Koike
koike@pi.titech.ac.jp

Makoto Sato
msato@pi.titech.ac.jp

Precision and Intelligence Laboratory, Tokyo Institute of Technology
4259 Nagatsuda, Midoriku, 226-8503 Yokohama, Japan
Tel :+81-45-924-5050 / Fax : +81-45-924-5016

Abstract

Because of the continuous development of computers it is now possible to construct various environments. As a result, human interfaces that allow users to manipulate virtual objects in an intuitive manner, as in the real world, are being demanded. In this paper, we present a 7 DOF tension-based haptic interface that allows users to not only pick the object but also to sense its width. We have developed a system to utilize the physical action of gripping to display grasp manipulation in virtual environments. We also present a method to calculate the position and display force associated with this gripping mechanism. In addition, we show the possibility of its application to virtual reality. Finally, we refer to the characteristic of this device and its validity through examples.

Keywords

7DOF, Tension based haptic interface, SPIDAR-G.

1. Introduction

The development of computer technology is enabling users to interact with various virtual environments. When users want to interact with virtual objects in a manner similar to those in the real world, an intuitive haptic interface with multiple degrees of freedom (DOF) becomes a necessity. In general, the physical act of gripping (or grasping) allows human beings to perform several important functions including using instruments to puncture, cut, rotate, and hit objects. Before doing the above-mentioned tasks, we select the necessary instruments by grasping it. Depending on the size and shape of the object, we can generally grasp an object using our thumb and our other fingers. So far haptic interfaces have presented users with simple ways of representing this grasping function, such as pushing a button in a mouse or keyboard. We believe that an effective haptic interface should not only provide feedback on the differential sense of width. Such an "intuitive" haptic interface has not been developed yet. The purpose of this paper is to realize such a tension-based 7 DOF haptic interface that can allow users to not only pick an object, but to also sense the width of an object as in real life object manipulation.

2. Related work

We can divide the haptic interfaces that have been developed so far into two categories: ground-based type and body-based type. LRP data glove by LRP[1], Cybergrasp force feedback glove by Virtual Technologies Inc[2], and Rutgers Masters (RM-II)[3],[4] developed at Rutgers University are well-known examples of body-based haptic interfaces. Body-based haptic interfaces have the advantage of allowing the user to grasp an object, but also present the disadvantage of not being able to represent the weight of an object. Recently, developers have tried to overcome this demerit in Vti by fixing the Cybergrasp force feedback glove to a serial link manipulator. Still, this device has the disadvantage of not being efficient in displaying rotational force. Furthermore, the overall structure is complex in that is cumbersome to put on the users hand and is difficult to maintain.

Ground-based haptic interfaces can generally be classified as link type, magnetic levitation type, and tension based type. Link type haptic interfaces have the disadvantage of exhibiting backlash, backdrive friction and inertia, and limited work space. The PHANToM is an example of a successful link type haptic interface. However,

since it has 6 DOF, it is impossible to grasp virtual objects using a single PHANToM. When 2 PHANToMs are used to grasp virtual objects, only 2 fingers are displayed in the virtual environment (thumb and index finger). This setup also suffers from limited workspace, due to inertial effects. Recently, a haptic group at MIT succeeded in integrating the Immersion Impulse Engine, a recent invention by Immersion Corporation[5], and 3 DOF PHANToM[6] (made by Sensable Technologies)[7] for laparoscopic surgery simulation[8]. The system utilized a 5 DOF simulation software with the PHANToM as the laparoscopic tool [9]. Therefore it did not provide feedback of rotation force to its users. The Haptic Master [10] is another well-known parallel-link type haptic interface. Because this device uses a gear, it has backlash and backdrive friction while displaying only 6 DOF.

CMU's magnetic levitation type haptic interface [11], [12] has the advantages of non-contact actuation and sensing, high control bandwidths, high position resolution and sensitivity, but has disadvantages of small workspace (motion range:15-20 degrees rotation, 25mm translation) and only 6 DOF display.

Tension based haptic interfaces [13], [14], [17], [18] have the advantages of fast reaction speed, simple structure, smooth manipulation, and scalable work space (since tension based types do not affect backlash, backdrive friction and inertia). The SPIDAR-G has 7 DOF, users can manipulate virtual objects with 6 DOF and can grasp them simultaneously. SPIDAR-G stands for SPace Interface Device for Artificial Reality with Grip.

3. Force displaying using tension

One characteristic of using strings to display forces is that they can only be used to represent tension. In other words, the strings can be used to pull and not push. We can determine the number of strings needed by applying vector closure to the indispensable condition of displaying n-DOF reflective forces using strings. When generating a n-dimensional force vector $q \in R^n$, using m-strings, the force vector q added to the target object from m-strings can be shown like this.

$$q = [w_1, w_2, \cdots, w_m]\tau \qquad (1)$$

$$w_i \in R^n \quad (i = 1, \cdots, m)$$

$$\tau = (\tau_1, \tau_2, \cdots, \tau_m)^T$$

Where w_i represents a force vector, when unit tension is added to the i -th string and τ is tension vector. The following theories (1 and 2) outline the Conditions for a positive τ that can realize any q in equation (1) [15], [16].

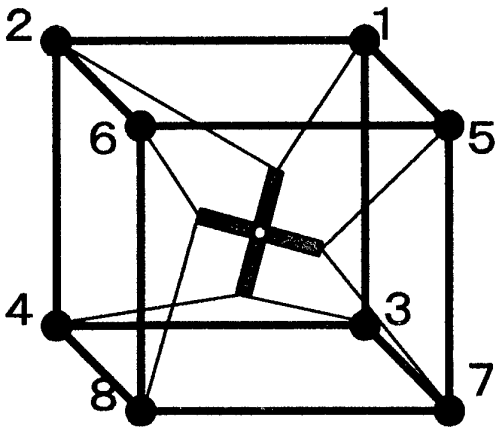


Figure 1. Basic structure of SPIDAR-G

[Theory 1]

If $A = [w_1, w_2, \cdots, w_m]$, the indispensable condition to have positive solution in equation (1) is as follows:

m>n

[Theory 2]

If $A = [w_1, w_2, \cdots, w_{n+1}]$, the indispensable condition to have positive solution in equation (1) is as follows:

1. $rank(A) = n$

2. Using remain row vector, any $w_i (i = 1, \cdots, n+1)$ have to represent as

$$w_i = -\sum_{j=1(j\neq i)}^{n+1} \alpha_j w_j \quad (\alpha_j > 0) \qquad (2)$$

However, n is the dimension of the work coordinate. Therefore, we can conclude that for the user to move an object in any direction in n-dimensional space, n+1 strings are need. Furthermore, the connection of the strings has to satisfy theory 2. In our case, SPIDAR-G needs at least 8 strings to display 7 DOF.

4. Structure of SPIDAR-G

4.1 Basic structure

Although we deduced that it was sufficient for us to use only 8 strings and that the connection had to comply with theory 2 (from vector closure), we still need to choose the best possible configuration for the connection of strings. This is because the magnitude, direction and area of force depend on the types of connection between the frame and grip. In general, we assume the users of our device would work in the central area of the frame. We choose the simplest way to display 7 DOF force in the central areas of

Figure 2. State of grip before grasping

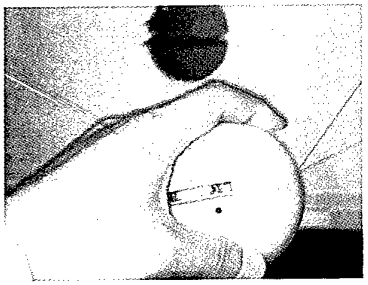


Figure 3. State of grip after grasping

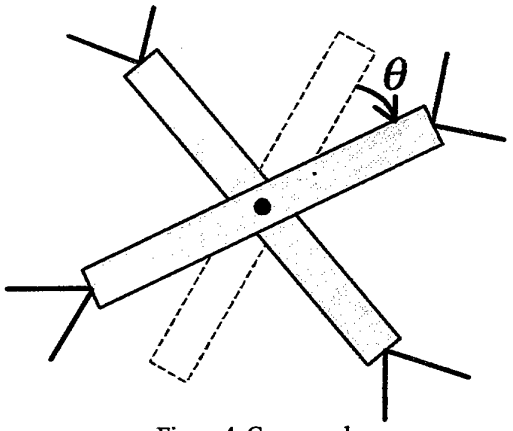


Figure 4. Grasp work

the frame by using low torque. In other words if the position vector of grip was set to $A(\in R^{7\times 8})$, the larger the result of $\det|A^TA|$, the better it was for our purpose. Using this type of an analysis we could take the best connection between a vertex of a grip and a corner of a frame, as in figure 1. At each corner of the frame, an encoder and a motor was attached. The 8 strings were connected to each of the corners of the frame. On the opposite side, the other 8 strings were connected to each of the 2 strings on the vertex of the grip as well. The encoder calculated the length

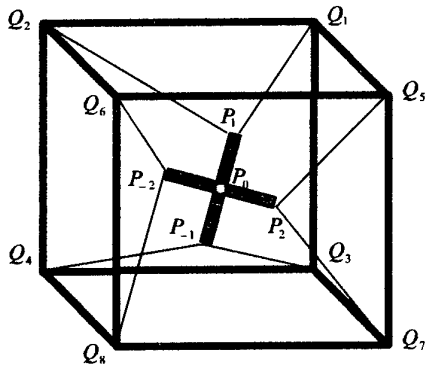


Figure 5. Definition of symbols

of string and the motors produced tension by pulling on the string.

4.2 Structure to grasp

Human beings are naturally skilled at grasping objects using their thumb and fingers. To display feedback force on the individual fingers, we initially tried attaching strings to the tips of each finger. This approach was not successful as it provided to be difficult to display translational, rotational, and grasping forces using only 8 strings.

In this paper we suggest a new mechanism for the grip. The new grip allows its users to manipulate with 7 DOF by grasping it between the thumb and other fingers. In order to incorporate the "grasping" functionality of the grip, it is best to consider a spherical shape, In figure 2, the proposed mechanism is broken into 2 hemispherical structures. As can be seen, if the user grasps the grip using thumb and other fingers, the 2 poles rotate depending on the magnitude of the grasp force (see figure 3). Hence it is possible to control the grasp functionality of the grip. The basic structure of the cross type grip is shown in figure 4.

The crossing degree θ changes with the magnitude of the grasping force and is used to quantify the action of grasping.

5. Way to calculation position

Position (translation, rotation, and grasp) is calculated from the length of 8 strings. The shape of frame is rectangular parallelepiped and each size of X, Y and Z axis is $2a,2b,2c$.

We take the center of frame as the origin (0,0,0) . Each position vector Q_i $(\in R^3)$ in i -th extremity of frame is as follows.

$$Q_1 = (\ a,\ b,\ c) \quad Q_2 = (\ -a,\ b,\ c)$$

$$Q_3 = (\ a,\ -b,\ c) \quad Q_4 = (\ -a,\ -b,\ c)$$

$$Q_5 = (\ a,\ b,\ -c) \quad Q_6 = (\ -a,\ b,\ -c)$$

$$Q_7 = (\ a,\ -b,\ -c) \quad Q_8 = (\ -a,\ -b,\ -c)$$

Position vectors of the grip (P_0) and the 4 extremities (P_1, P_{-1}, P_2, P_{-2}), P_i ($\in R^3$) are defined below.

$$P_0 = (x, y, z)$$

$$P_1 = (x + x_1, y + y_1, z + z_1)$$

$$P_{-1} = (x - x_1, y - y_1, z - z_1)$$

$$P_2 = (x + x_2, y + y_2, z + z_2)$$

$$P_{-2} = (x - x_2, y - y_2, z - z_2)$$

If we set the length of each pole to $2d$, the following equation comes out.

$$x_1^2 + y_1^2 + z_1^2 = x_2^2 + y_2^2 + z_2^2 = d^2$$

If we set each extremity of grip which is connected to the i-th frame (i), we can easily know the following relation.

$$(1) = (2) = 1 \text{ ,} \cdots (3) = (4) = -1$$

$$(5) = (7) = 2 \text{ ,} \cdots (6) = (8) = -2$$

Setting the length of the i-th string, l_i can be represented with the following equation.

$$l_i = \left\| Q_i - P_{(i)} \right\| \quad (i = 1, \cdots, 8) \qquad (3)$$

To calculate translation, rotation, and grasp, we have to solve (x, y, z), (x_1, y_1, z_1), and (x_2, y_2, z_2) from the length of 8 strings. Equation 3 can be converted into the following equations.

$$(x + x_1 - a)^2 + (y + y_1 - b)^2 + (z + z_1 - c)^2 = l_1^2 \quad (4)$$

$$(x + x_1 + a)^2 + (y + y_1 - b)^2 + (z + z_1 - c)^2 = l_2^2 \quad (5)$$

$$(x - x_1 - a)^2 + (y - y_1 + b)^2 + (z - z_1 - c)^2 = l_3^2 \quad (6)$$

$$(x - x_1 + a)^2 + (y - y_1 + b)^2 + (z - z_1 - c)^2 = l_4^2 \quad (7)$$

$$(x + x_2 - a)^2 + (y + y_2 - b)^2 + (z + z_2 + c)^2 = l_5^2 \quad (8)$$

$$(x - x_2 + a)^2 + (y - y_2 - b)^2 + (z - z_2 + c)^2 = l_6^2 \quad (9)$$

$$(x + x_2 - a)^2 + (y + y_2 + b)^2 + (z + z_2 + c)^2 = l_7^2 \quad (10)$$

$$(x - x_2 + a)^2 + (y - y_2 + b)^2 + (z - z_2 + c)^2 = l_8^2 \quad (11)$$

Using the above equations, we can solve (x, y, z), (x_1, y_1, z_1), and (x_2, y_2, z_2). We can solve above variables using 4 arithmetical operations because of the redundancy of strings. We show the detailed algorithm in index 1.

6. Way to display reflect force

In this section, we explain how to determine tension of the 8 strings to display 7 DOF force in crossing type grip.

We define force vector $q(\in R^7)$ should be generated like this.

$$q = (f_x f_y f_z m_x m_y m_z, g)^T$$

Where f_x, f_y, f_z represent translation forces, m_x, m_y, m_z rotation forces, and g is the grasp force.

We define the tension of string $\tau_{(i)}$ $(i = 1, \cdots, 8)$, and tension vector τ as follows:

$$\tau = (\tau_1, \tau_2, \cdots, \tau_8)^T \quad (\in R^8)$$

We set w_i as the force vector generated in the grip as the unit tension is added to i-th string. w_i is defined below.

$$w_i = \begin{bmatrix} c_i \\ r_{(i)} \times c_i \\ \delta_i \cdot n \cdot r_{(i)} \times c_i \end{bmatrix}$$

However,

$$c_i = \frac{Q_i - P_{(i)}}{\left\| Q_i - P_{(i)} \right\|} \quad (i = 1, 2, \cdots, 8)$$

$$r_{(i)} = P_{(i)} - P_0$$

$$\delta_i : \begin{Bmatrix} 1 & i = 1,2,3,4 \\ -1 & i = 5,6,7,8 \end{Bmatrix}$$

$$n = \frac{r_1 \times r_2}{\left\| r_1 \times r_2 \right\|}$$

If we set $A \in R^{7 \times 8}$ into $A = (w_1, w_2, \cdots, w_8)$, the force vector q, given tension vector τ, can be represented as

$$q = A\tau$$

To display force vector q to cross type grip, we have to solve the tension vector τ which satisfies the above equation. However, the tension vector is positive value vector ($\tau_i \geq 0, i = 1, \cdots, 8$). If we solve 2 degree Optimum problem, we can obtain the tension vector.

$$\left\| q - A\tau \right\| \rightarrow \min \cdots$$

$$\cdots \text{s.t.} \cdots \tau \geq 0$$

Because SPIDAR-G uses the tension of strings to display force, according to the position of grip, there are certain location in the frame where SPIDAR-G can not display appropriate forces. However, near the center of frame,

SPIDAR-G can display 7 DOF force appropriately (see index 2).

7. Development of SPIDAR-G

We show manufactured SPIDAR-G in figure 6. The length of frame is 52cm, and the radius of grip is 4.1cm. The computer which was used with the SPIDAR-G in figure 6 is Pentium 400 MHz. The encoder was a HEDS-5540 made by HP company. A DC motor made by Maxon company was used.

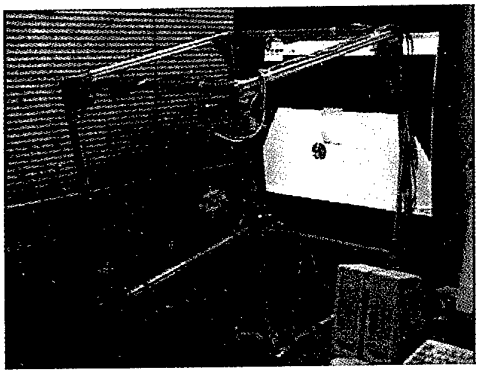


Figure 6. Manufactured SPIDAR-G

In following example, the user grasps the grip and the color of grip in the monitor turns red. On release, the color turns blue. This allows users to perceive not only force feedback but also a visual representation.

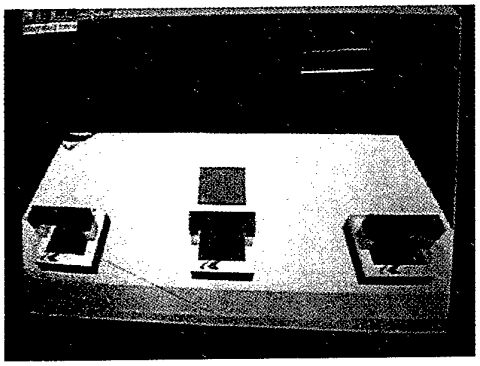


Figure 7. Example of lift up virtual object

We prepared 3 different weighted objects, and lift up each object with grasping. From this work, we could distinguish weight difference of the each object. However, it was difficult for users to perceive the same weight when far from the center of the frame. In addition, users had difficulty perceiving the width of the objects when the grip was positioned far from the center of the frame.

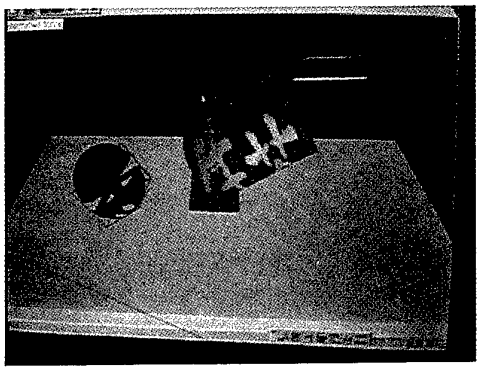


Figure 8. Example of grasp and rotation

In the demonstration represented in figure 8, users have to grasp the object and rotate into each X axis(depth direction), Y axis(vertical direction) and Z axis (horizontal direction). It was easier to manipulate from the X axis, the Y axis, and the Z axis respectively.

8. Conclusion and future work

In this paper, we described the tension based haptic interface with 7 DOF. We can get the precise solution for the position of grip using the redundancy of strings and the unique geometric characteristic of this system. We have also showed a new way to calculate position with a 7 DOF. Through examples using SPIDAR-G, we have demonstrated the validity of our proposed SPIDAR-G. The examples prove that our contrived SPIDAR-G provides users with not only translation and rotation but also grasp manipulation that is accurate and efficient. There are still issues with rotational force stability, which hope to address in future research.

Acknowledgement

We would like to thank Mr.Jeffrey J, Berkley of the Human Interface Technology Lab at the University of Washington for proofreading and editing this manuscript.

References

[1] M. Bouzit, P. Coiffet and G. Burdea, The LRP Dextrous Hand Master, Proceedings of VR System'93 Conference, Newyork City, October 1993.

[2] http://www.virtex.com

[3] D.Gomez, G. Burdea and N. Langrana, The Second Generation Rutgers Master – RM II, Proceedings of

Automation'94 Conference, Taipei, Taiwan, Vol. 5, pp.7-10, July 1994.

[4] Fabiani, L., G. Burdea, N. Langrana and D. Gomez., Human Performance Using the Rutgers Master II Force Feedback Interface, IEEE International Symposium on Virtual Reality and Applications (VRAIS'96), Santa Clara, CA., pp. 54-59, March 1996.

[5] http://www.immersion.com/impulseengine.html

[6] T.H. Massie. Design of a Three Degree of Freedom Force-Reflecting Haptic Interface. Bachelor of Science thesis, Massachusetts Institute of Technology, May, 1993.

[7] http://www.sensable.com

[8] M.P. Ottensmeyer et al. Input and Output for Surgical Simulation: Devices to measure Tissue Properties in vivo and a Haptic Interface for Laparoscopy Simulators. Medicine Meets Virtual Reality 2000. IOS Press, 2000.

[9] E. Ben-Ur, Development of a Force-Feedback Laparoscopic Surgery Simulator. Master Thesis, Department of Mechanical Engineering, MIT, 1998.

[10] H. Iwata. Artificial Reality with Force-Feedback: Development of Desktop Virtual Space with Compact Master Manipulator. Computer Graphics (SIGGRAPH' 90 Proceeding) pp.165-170, 1990.

[11] P.J.Berkelman, Z.J.Butler and R.L. Hollis. Design of a Hemispherical Magnetic levitation Haptic Interface. DSC-Vol. 58, Proceedings of the ASME Dynamics Systems and Control Division, pp.483-488, 1996.

[12] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/msl/www/haptic_device.html

[13] M. Sato, Y. Hirata and H. Kawarada. SPace Interface Device for Artificial Reality-SPIDAR. The Transactions of the Institute of Electronics, Information and Communication Engineers (D-II), J74-D-II, 7, pp.887-894, July, 1991.

[14] M. Ishii, M. Nakata, and M. Sato. Networked SPIDAR: A Networked Virtual Environment with Visual, Auditory, and Haptic Interaction, PRESENCE (MIT Press Journal), Vol.3, No.4, pp351-359, 1994.

[15] S. Kawamura and K. Ito. A New Type of Master Robot for Teleoperation Using A Radial Wire Drive System. Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and System, pp.55-60, 1993.

[16] A.J.Goldman and A.W. Tucker. Polyhedral Convex Cones in Linear Inequalities and Related System. H.W.Kuhn and A.W.Tucker editors, Princeton Univ. Press, 1956.

[17] M. Ishii and M. Sato. Force Sensations in Pick-and-Place Tasks. International Conference of American Society of Mechanical Engineering 1994, Chicago, USA, DSC-Vol.55-1, pp.339-344, 1994.

[18] L. Bouguila, Y.Cai and M.Sato. New Interface Device For Human-Scale Virtual Environment: Scaleable-SPIDAR. International Conference on Artificial reality and Tele-existence (ICAT97), pp.93-98, Tokyo, 1997.

[19] S.Kim, W. Somsak, M.Ishii, Y.Koike, and M.Sato. Personal VR system for rehabilitation to hand movement. International Conference on Artificial reality and Tele-existence (ICAT98), pp102-108 ,Tokyo, 1998.

[20] M.Ishii and M. Sato. A 3D spatial Interface device using tensed strings. Presence, 3(1), pp.81-86, 1994.

[21] K.Hatano, M.Ishii and M.Sato. Six Degree of Freedom Master Using Eight Tensed Strings. Measurement and Control in Robotics Proceedings, pp.251-255, June, 1998.

Index 1

From equation (4)~(11), we can get each x, y, x_1, y_2 as follows.

About x : -eq(4)+eq(5)-eq(6)+eq(7)

About y : -eq(8)-eq(9)+eq(10)+eq(11)

About x_1 : -eq(4)+eq(5)+eq(6)-eq(7)

About y_2 : -eq(8)+eq(9)+eq(10)-eq(11)

Therefore,

$$x = \frac{1}{8a}\left(-l_1^2 + l_2^2 - l_3^2 + l_4^2\right)$$

$$y = \frac{1}{8a}\left(-l_5^2 - l_6^2 + l_7^2 + l_8^2\right)$$

$$x_1 = \frac{1}{8a}\left(-l_1^2 + l_2^2 + l_3^2 - l_4^2\right)$$

$$y_2 = \frac{1}{8a}\left(-l_5^2 + l_6^2 + l_7^2 - l_8^2\right)$$

We substitute the above solutions into equation (12)~(15).

eq(4)• eq(5)• eq(6)• eq(7)• • • • • • • 12

eq(4)• eq(5)• eq(6)• eq(7)• • • • • • • (13)•

eq(8)• eq(9)• eq(10)• eq(11)• • • • • • (14) • • • •

eq(8)• eq(9)• eq(10)• eq(11)• • • • • • (15) • • • •

We can get 4 equations about Z. These 4 equations can be changed into 2 six degrees equations. General case, we

use numeric method to solve six degrees equations. But this method requires suitable conditions, substantial time due to the iterative nature of the technique, and the results are only approximations. It is necessary to reduce the amount of calculations to maintain the haptic servo loop and we need precise results rather than approximations to earn high resolution. Numerical methodologies are therefore unsuitable.

Fortunately, in the case of using strings, we can know that the above 2 six degree equations contain the same result about z due to the redundancy of the strings. By either adding or subtracting 2 equations, we are able to get a six degree equation and a five degree equation which have a common solution about z. By dividing the high degree equation by the low degree equation, we get the result of Z. Using this, we can solve other variables (y_1, z_1, x_2, y_2, z_2).

$$y_1 = \frac{1}{2b}\{k_1 + (z-c)^2\}$$

$$x_2 = \frac{1}{2a}\{k_3 + (z+c)^2\}$$

$$y_2 = \frac{1}{8b}\left(-l_5^2 + l_6^2 + l_7^2 - l_8^2\right)$$

$$z_2 = \{\frac{1}{8}(l_5^2 - l_6^2 + l_7^2 - l_8^2) - yy_2 - ax - xx_2\}/(z+c)$$

However,

$$k_1 = x^2 + y^2 + d^2 + a^2 + b^2 - \frac{1}{4}\sum_{i=1}^{4} l_i^2$$

$$k_2 = x^2 + y^2 + d^2 + a^2 + b^2 - \frac{1}{4}\sum_{i=5}^{8} l_i^2$$

Index 2

That is to say, our system satisfies following equation in the center of the frame.

$$\sum_{1}^{8} w_i = 0$$

We can know that that equation satisfies theory 2.

ICAT 2000

4+4 Fingers Haptic Display in the Mixed Reality Environment

Keita Yamada, Somsak Walairacht, Shoichi Hasegawa,
Masahiro Ishii, Yasuharu Koike, Makoto Sato
Precision and Intelligence Laboratory, Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku, Yokohama 226-8503, Japan
{kyamada, Somsak, hase, mishii, koike, msato}@pi.titech.ac.jp

Abstract

A system with direct manipulation environment is proposed. A user is allowed to use both of his/her hands to manipulate virtual objects in the simulated virtual world. 3D graphical displays with 3D virtual hands represented the user's real hands in the virtual world showing the manipulation works in the virtual world to the user. The 3D virtual hands move corresponding to the behavior of the real hands.

We have developed a two-handed multi-fingers string-based haptic interface device. By using this device, force feedback can be displayed at the eight fingertips (4 fingers on each hand) of the user. Eight fingertip positions measured from the user's real hands are used in modeling 3D virtual hands. By computing the joint angles of the fingers, the virtual hands pose can be estimated.

In this paper, we have discussed about design policy of the system. Algorithms and computation are also given in detail. A manipulation of the virtual Rubik's cube is constructed and is given as an application of the proposed system.

Key words: String-based haptic interface device, 3D virtual hand, Direct manipulation

1. Introduction

With recent improvements of computer system, virtual reality (VR) system has shown high performance and abilities to simulate many kinds of task. The simulation can be for various kinds of purpose, such as training, education, working in the remote site or in the dangerous place, etc. An effective simulation requires natural interaction between human and the system in the same way as performing in real world. Conventional 2D computer input/output devices, such as 2D-mice or keyboards, are insufficient and difficult to perform the simulated VR tasks.

Consider human interactions in real world, many tasks involve the usage of our hand(s). For example, an interaction likes object manipulation by a hand, we perceive the sense of touch (haptic feedback) when grasping the object in our hand. This haptic feedback tells us the existence of the object, that why we can grasp it stably. The sight (visual feedback) of the object and our hand helps us to move the hand and reach for the object correctly and precisely. We can place our fingers at the right position on the object and watch the object being manipulated by hand as desire. Although, human being also uses some other sensory feedbacks when interacting with the environment, such as, audio, temperature, odor, etc., but haptic and visual feedbacks are two major information that human often uses intuitively for the interaction.

It is still difficult to integrate and to provide all of the sensory feedbacks that human used in real world on the current VR systems. At present, any VR system that can also be considered as an effective system, at least, it should be able to provide good quality of haptic and visual feedbacks and its user can perform the simulation task naturally in similar way as in the real world. Therefore, there are many research works are now working to achieve this goal.

In recent years, many input and output devices have been developed and proposed as hand haptic interface device. PHANToM[7] is a haptic interface devices widely used for VR simulation tasks. However, it is only single-point interface device in which user can directly use a finger or, indirectly, controls through its stylus pointer. We often see the combination of two PHANToM systems use in grasping a virtual object effectively. However, to employ PHANToM to every finger on user's hand is not an applicable way of implementation. The system will become too complex, bulky, and expensive. Next, the input device for a hand likes the DataGlove[6] can only measure positions on the user's hand but cannot display any force feedback. Also, some others hand haptic interface device, called Exoskeleton[2-5], requires complex structure to be installed on user's hand for position measurement and force generation. The complexity of the system and the weight of the device itself are its main drawbacks. Meanwhile, the string-based haptic interface device is

simpler in structure and control, safe, and light-weighted. The SPIDAR[16-18] systems are successful examples of this type of device. However, the interference of string, either by part of user's body or among the other strings, is always cause problems to this type of device.

In this paper, we propose a system for human interaction with the virtual world. The system provides force feedback to the user by using string-based haptic interface device. At the same time, the system provides graphical display of virtual world with 3D virtual hands mirror the movement of the user's real hands. We consider many issues on construction of the proposed system to provide quality of haptic and visual feedbacks to the user.

2. Design policy of the proposed system

2.1 Two-handed multi-fingers string-based haptic interface device; SPIDAR-8

The proposed system is an improved version of SPIDAR[16-18]. The early systems use four strings attaching to a finger of the user, however, the proposed system uses only three strings. As shown in Fig. 1, three strings from each corner of the frame are connected together and to be attached to a fingertip of the user. By the structure of the rectangular frame, the system has eight interface points in which four points are to be attached to four fingers on the left hand and the other four points are to be attached to four fingers on the right hand. Therefore, a user can use both hands and multi fingers to interact with the virtual world. There is no string passing between both hands to reduce chance of the interference of strings. Since the proposed system allows a user to use eight fingers, we have named the system as SPIDAR-8[19-22].

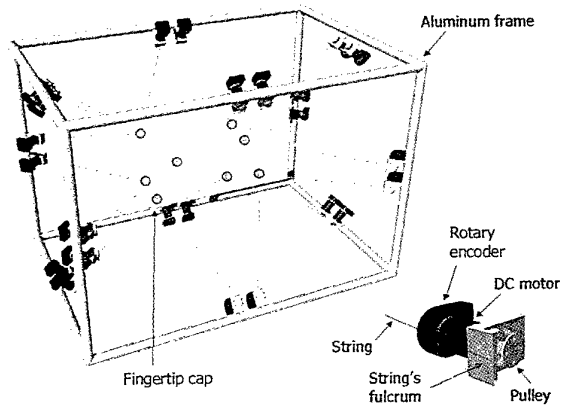


Fig. 1 SPIDAR-8

2.2 Two-handed multi-fingers string-based haptic interface device; SPIDAR-8

Because the user needs only to wear fingertip caps on his/her eight fingers when using the proposed system and only a small value of tension force (about 0.2N) is applied to straighten each string for the purpose of position measurement, the user can have full freedom of movement and the usage of hands to direct manipulate the virtual objects. Force feedback is displayed at the fingertips of the user by controlling the tension of the strings. Since there is no interface with the whole finger or palm of the user's hand, the system cannot display power-grasping-force by whole hand. However, with force feedback at fingertips, the dexterous object manipulation using fingertips can be effectively performed.

2.3 Two-handed multi-fingers string-based haptic interface device; SPIDAR-8

Vision is an important part of the interaction as it can increase the level of immersion. The proposed system models 3D virtual hands that mirror the movement of the user's real hands. Consider an example of grasping an object by a hand, the thumb is seen in front, the object is in the palm, and the other fingers are occluded behind the object. By displaying the virtual hands, it is possible to present such appearance with correct position and orientation of virtual thumb, fingers, and palm grasping the virtual object. That means the user using the proposed system can perceive the visual feedback in the same appearance of real hand grasping real object.

3. Overview of the proposed system

The processes of the whole system can be divided into 3 subsystems as shown in Fig. 2.

1. Haptic subsystem
2. Virtual world management subsystem
3. Visual subsystem

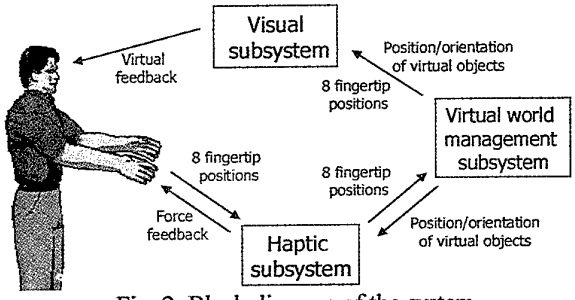


Fig. 2 Block diagram of the system

First, in the haptic subsystem, SPIDAR-8 is used to measure eight fingertip positions on the user's real hands. The fingertip positions with reference to the virtual world are calculated and sent to the virtual world management subsystem.

Next, the virtual world management subsystem is in charge of collision detection of positions of fingertips and the virtual objects. Force feedback value for each finger is calculated in this subsystem and sent back to haptic subsystem when the collision is occurred. The haptic subsystem controls tension of strings according to force feedback value for each finger and displays forces to user's fingertips. The motion of the virtual object in virtual world is the result of the collision forces acting on the object.

In the visual subsystem, 3D computer graphics of virtual world is rendered. The updated fingertip positions update the model and motion of the virtual hands. The motions of the virtual objects are updated by the updated position/orientation of the virtual objects. On the display screen, the user can see the virtual objects in the virtual world being manipulated according to the behavior of his/her real hands.

4. System implementation

4.1 Haptic subsystem

4.1.1 Position measurement

From the structure of frame of SPIDAR-8 and the attachment of three strings to one finger, the measurement and calculation of all fingertip positions can be performed in the same configuration. By measuring the length of three strings and substituting the corresponding positions of string's fulcrums, position of each fingertip can be calculated by the following computation.

Let $l_i (i = 1,2,3)$ is the length of each string measured from a fingertip position P to corresponding string's fulcrum $A_i (i = 1,2,3)$. The vectors $\vec{n}_1$ and $\vec{n}_2$ are unit vectors along the vectors $\overrightarrow{A_2A_1}$ and $\overrightarrow{A_3A_1}$ respectively. And $\vec{n}_3$, is the cross product of $\vec{n}_1$ and $\vec{n}_2$.

$$\vec{n}_1 = \frac{A_2 - A_1}{\|A_2 - A_1\|},\ \vec{n}_2 = \frac{A_3 - A_1}{\|A_3 - A_1\|},\ \vec{n}_3 = \vec{n}_1 \times \vec{n}_2$$

From the diagram shown in Fig. 3, the position of point P can be found by the following equation.

$$P = A_1 + \alpha_1 \vec{n}_1 + \alpha_2 \vec{n}_2 + \alpha_3 \vec{n}_3 \qquad (1)$$

where the values of α_1, α_1, and α_1 can be derived from the following.

$$\alpha_1 = \frac{1}{\sin^2\theta}\left(\frac{K_1}{d_1} - \cos\theta \cdot \frac{K_2}{d_2}\right)$$

$$\alpha_2 = \frac{1}{\sin^2\theta}\left(\frac{K_2}{d_2} - \cos\theta \cdot \frac{K_1}{d_1}\right)$$

$$\alpha_3 = \sqrt{l_1^2 - \|\alpha_1 \vec{n}_1 + \alpha_2 \vec{n}_2\|^2}$$

and

$$\theta = \cos^{-1}(\vec{n}_1 \cdot \vec{n}_2)$$

$$d_1 = \|A_2 - A_1\|,\ d_2 = \|A_3 - A_1\|$$

$$K_1 = \frac{1}{2}\{d_1^2 - (l_2^2 - l_1^2)\},\ K_2 = \frac{1}{2}\{d_2^2 - (l_3^2 - l_1^2)\}$$

4.1.2 Force feedback generation

SPIDAR-8 displays force feedback at the fingertips of the user by controlling the amount of electric current entering the DC motors. The tension force on each string, $t_i (i = 1,2,3)$, and the unit vector $\vec{u}_i (i = 1,2,3)$ are used to compose the resultant force vector as in the following equation.

$$\vec{f} = \sum_1^3 t_i \vec{u}_i \qquad (2)$$

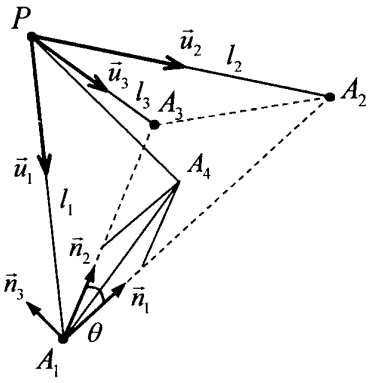


Fig. 3 Position measurement and force generation

where

$$\vec{u}_i = \frac{A_i - P}{\|A_i - P\|}$$

The connection of three strings from three fulcrums to a point is forming a triangle cone of force display for each finger. Force feedback can be displayed correctly in the case that the resultant force vector lies inside the force cone. However, in the case that the resultant force vector is outside the force cone, the projection of the force vector back to the force cone is computed and the resultant force vector is recomposed. By this way, SPIDAR-8 can display appropriated force feedback to the user using tension of string(s).

4.2 Virtual world management subsystem

4.2.1 Collision detection

A virtual object is defined by its dimension and position in the virtual world. If Q is a point on the plane of virtual object, and $\vec{N}$ is a normal vector pointing inside the virtual object, then the collision of fingertip position P can be detected by examining the sign of $(P-Q)\cdot\vec{N}$.

$$(P-Q)\cdot\vec{N}\begin{cases} > 0; & P \text{ is inside} \\ < 0; & P \text{ is outside} \\ = 0; & P \text{ is in contact} \end{cases} \qquad (3)$$

4.2.2 Force and motion of object

A repulsive force, f, is generated using the conventional penalty-based method in which the amount of force is in proportional to the amount of penetration, d, into the virtual object (k=force constant).

$$f = kd \qquad (4)$$

The motion of the virtual object is computed by using fundamental Newton's law of motion.

$$m\frac{dv}{dt} = f \tag{5}$$

$$I\frac{d\omega}{dt} = (p-r)\times f \tag{6}$$

where

m : mass of the object

v : velocity of the object at center of gravity

f : force acts on the object

I : inertia tensor matrix

ω : angular velocity of the object

p : position where f acts on

r : center of gravity of object

4.3 Visual subsystem

In the previous system [20], SPIDAR-8 attached strings to three fingertips and a position on the wrist for each hand of the user as shown in Fig. 4. These measured positions are used to model a virtual hand. In that case, the user can use only three fingertips on each hand for virtual objects manipulation with force feedback sensation. There is no force feedback generated at the wrist position. In the present system, strings are attached to four fingertips and the system allows the user to use four fingertips in the manipulation and perceive force feedback. The strings are attached to the Thumb, index finger, middle finger, and ring finger on each hand of the users (see Fig. 4). With four fingers, the ability to manipulate the virtual object is obviously increased. The user can grasp the virtual objects naturally, more stable, and less finger's fatigue compared to using three fingers. Moreover, the user can even rotate a virtual object grasped in a hand by using four fingers. It is almost impossible to perform such manipulation by only three fingers. Position and orientation of the virtual hand (6 DOF) is now assigned to be the parameters of the model of a virtual hand (17 DOF). It can be computed and used to estimate virtual hand pose from four fingertip positions. Detail of virtual hand pose is described in the following sections.

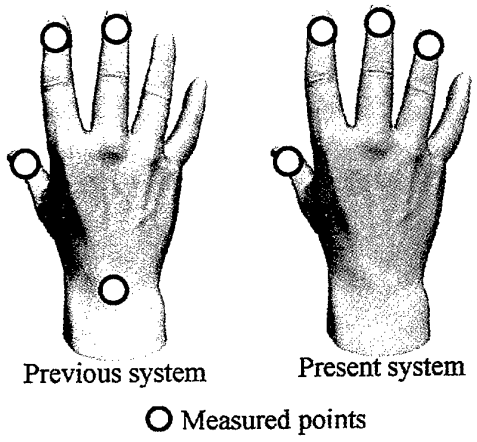


Fig. 4 Measured points

4.3.1 Model of human hand

Joint motion of all fingers defines the number of degree of freedom (DOF) on a hand. Each finger has 4 DOF, two at the connection with the palm and one at the end of first finger part and one at the second finger part. Figure 5 shows simple structure of a hand and DOF on each of the joint. 20 DOF of all finger joints and 6 DOF of translation and rotation of hand measured at the wrist make one human hand a 26 DOF manipulator.

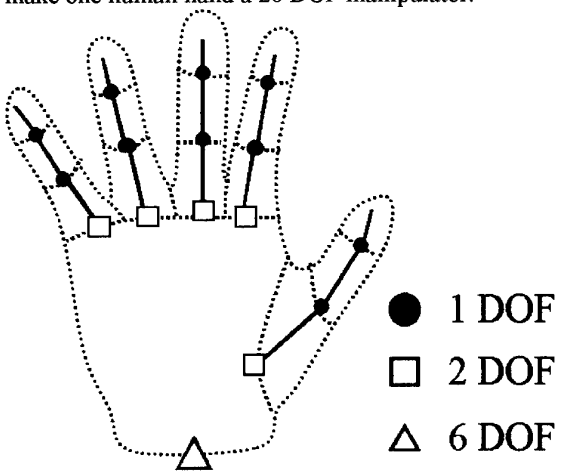


Fig. 5 Model of hand

4.3.2 Simplified model of virtual hand

Since SPIDAR-8 can measure four fingertip positions in 3D space, which is equivalent 12 DOF, we have found that it is too difficult to model a hand of 26 DOF by using only 12 known values. Thus, we have set up the criterions to reduce the number of DOF of the real hand for the virtual hand.

Criterions for reducing the number of DOF are as follows.

1. SPIDAR-8 does not measure position of little finger. Joint motion of each finger part of little finger is assigned to be the same as the corresponding part of ring finger.

2. A human finger has the property that it is impossible to move the joint closest to the fingertip without moving the next closest to the fingertip joint and vice-versa. Therefore, there is a dependency between these two joints, which causes by the same tendon used for moving inside the finger.

After measuring several times of these two joint angles as shown in Fig. 7, we found that it is reasonably approximated the dependency by a second-degree polynomial equation as shown in Eq. 7.

$$\theta_2 = 1.1341\theta_1^2 - 0.286\theta_1 \tag{7}$$

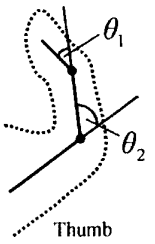


Fig. 6 Joint angles on thumb

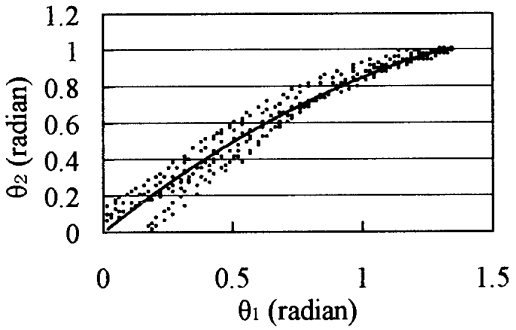


Fig. 7 Joint angle

3. The middle finger does not move side by side in most cases of grasping the object without forcing the finger to move in unnatural way. The joint motion at the connection of finger part and the palm of the middle finger can be reduced to 1 DOF.

Finally, the simplified model of a virtual hand after applied the above criterions has the number of DOF reduced to 17 DOF.

4.3.3 Virtual hand pose estimation

Forward kinematics is used to calculate joint angles of the fingers. To estimate virtual hand pose by placing the fingertips at the locations in the next update, inverse kinematics is required.

The algorithm for virtual hand pose estimation consists of the following steps.

Step 1. Retrieve fingertip position of thumb, index finger, middle finger, and ring finger sent by the virtual world management subsystem.

Step 2. Find the difference of the fingertip positions retrieved from Step 1 and the current fingertip positions of the virtual hand.

Step 3. If there is no difference, no hand pose estimation is performed. The process is finished and left the algorithm, otherwise, is preceded to the next step.

Step 4. Reduce the differences of fingertip positions by revising the model parameter of the virtual hand.

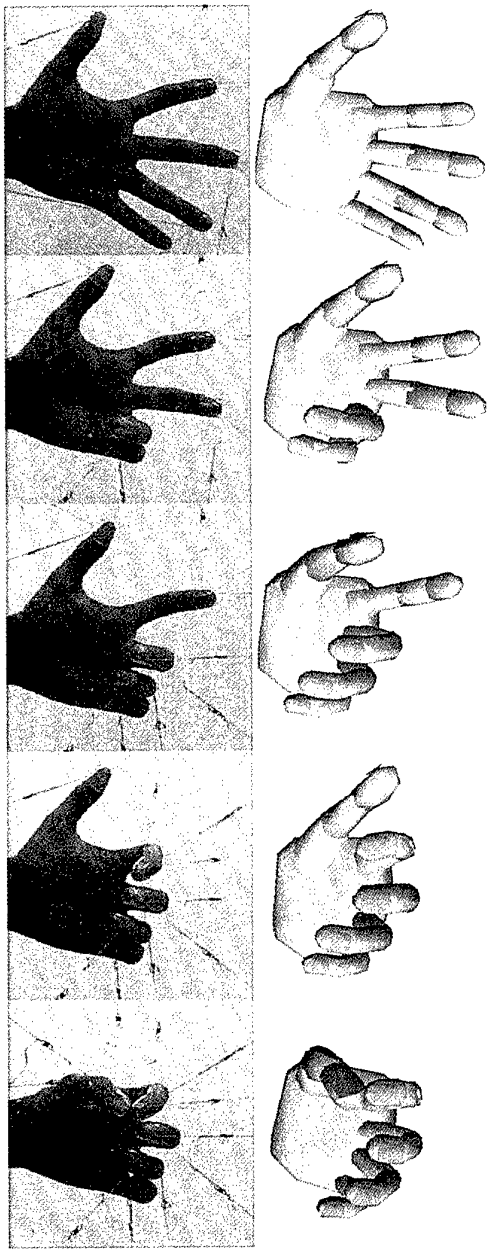
Step 5. Repeat the process from Step 2 again.

The process in the Step 4 of above algorithm, which is used for estimating hand pose, can be described in detail as follows.

Fingertip positions are expressed as matrix P and the model parameters (17 DOF) of the virtual hand are expressed as matrix θ as shown in Eq. 8 and 9 respectively.

$$P = (p_1, ..., p_{12})' \quad (8)$$

$$\theta = (\theta_1, ..., \theta_{17})' \quad (9)$$



(a) (b)

Fig. 8 Results of hand pose estimation (a) real hand of user, (b) virtual hand

Fingertip positions can be expressed as a function of model parameters.

$$P = f(\theta) \tag{10}$$

Substitute the function of model parameter by jacobian and the change quantity of fingertip positions can be derived from the change quantity of model parameters as shown in Eq. 11.

$$dP = J(\theta)d\theta \tag{11}$$

The pseudo-inverse matrix of jacobian $J(\theta)^+$ is computed and the change quantity of model parameter can be found as in Eq. 12

$$d\theta = J(\theta)^+ dP \tag{12}$$

where, the pseudo-inverse matrix of jacobian is shown as Eq. 13.

$$J(\theta)^+ = (J(\theta)^t J(\theta))^{-1} J(\theta)^t \tag{13}$$

Results of virtual hand pose estimation can be shown in Fig. 8. The left hand side column shows images of user's real hand in different postures and the right hand side column shows the virtual hand in the corresponding posture.

5. The constructed system and its application

5.1 Configuration

The configuration of the proposed system is shown as in Fig. 9. Frame of SPIDAR-8 is 80cmX60cmX60cm in dimension. The working space of both hands of the user is equivalent to the space enclosed by the frame. A user stands in front of SPIDAR-8 and forwards both of his/her hands to the center of the frame, where the virtual objects to be manipulated are located. An 18-inch LCD display is installed on top of the frame and is away from the user by the distance of 50cm. The 3D Computer graphics scene of the manipulation task is rendered and displayed to the user on the display.

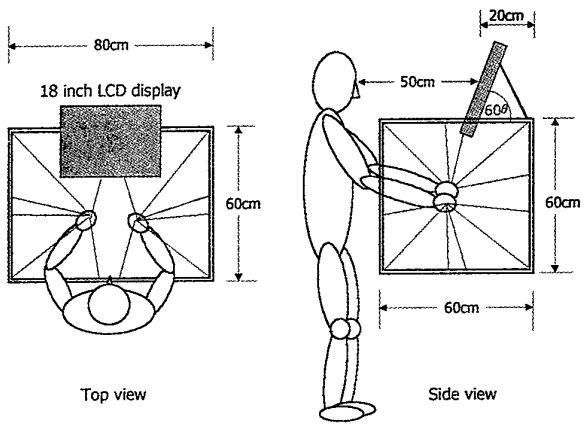


Fig. 9 System configuration

5.2 Application

The manipulation of a virtual Rubik's cube is selected as an application of the proposed system. Virtual Rubik's cube is a 2x2x2 cell, which its column-cell and row-cell can be rotated in the same way as the real Rubik's cube. The manipulation of the virtual Rubik's cube can clearly show the abilities of the system. Grasping or rotating cells of the cube is considered as a dexterous manipulation task using multi fingers. The rotation of two adjacent column-cells or row-cells requires both hands to perform in a cooperative way, in which one hand must grasp on one column-cell or row-cell and another hand grasps on the opposite column or row and rotates each hand in the opposite direction. If the cube is grasped by either left or right hand, the whole cube is rotated according to that hand's rotation. As shown in Fig. 10, a user is manipulating the virtual Rubik's cube by his real hands and, on the screen, the manipulation as a snapshot shown in Fig. 11 is presented.

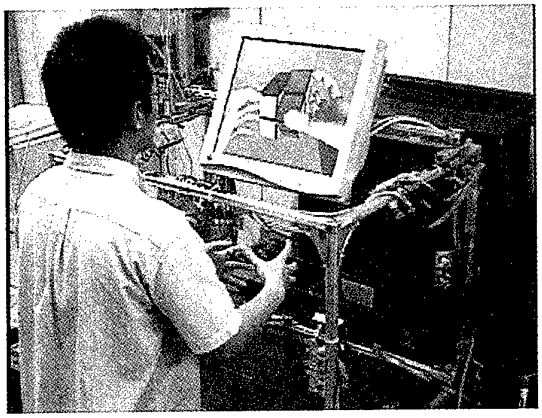


Fig. 10 A user is using SPIDAR-8 manipulate virtual Rubik's cube

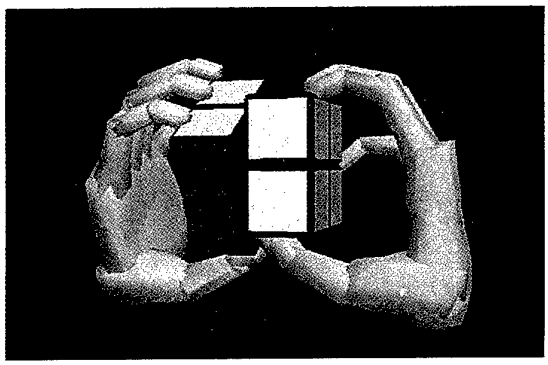


Fig. 11 Snapshot of virtual Rubik's cube manipulation

6. Conclusion and future works

A system with direct manipulation environment for the interaction with the virtual world is proposed. In the constructed environment, a user can perceive force and visual feedback, which allows he/she to manipulate the virtual objects in the same way as performing in the real world. We had developed a two-handed multi-fingers string-based haptic interface device named SPIDAR-8. By using our haptic interface device, a user can perceive

force feedback at eight fingertips when manipulating the virtual objects. The system displays 3D virtual hands representing real hands of the user in the virtual world. Using eight fingertip positions, 3D virtual hands are modeled and joint angle of the fingers are computed. Then, virtual hands pose can be estimated by the joint angles of the fingers. 3D virtual hands move naturally according to the movement of the user's real hands.

There are some future issues of this work to be further investigated.

1.The implementation of visual feedback using stereoscopic vision system.

In the present work, visual feedback is displayed on 2D computer screen in which the user has complained about the difficulty in perceiving of the depth, for example, when he is trying to grasp the virtual object, it is difficult to locate the fingers at the back of the object. The stereoscopic may be a solution as the 3D scene with depth can be seen. However, the coupling of haptic and stereopsis on depth perception[23] is still unclear and must be carefully considered.

2.The implementation of a mixed reality system.

Instead of displaying 3D virtual hands, the images of real hands of the user are merged into the virtual world. VDO camera is used to take image of user's real hands. Real-time images of hands combine with computer generated 3D virtual world by the technique of chroma-keying. This work is now under the implementation.

References

1. K. B. Shimoga, "A Survey of Perceptual Feedback Issue in Dexterous Telemanipulation: Part I. Finger Force Feedback", Proceedings of Virtual Reality Annual International Symposium VRIS, pp. 263-270, 1993

2. M. Bouzit, P. Richard, and P. Coiffet, "LRP Dextrous Hand Master Control System", Technical Reposrt, Laboratoire de Robotique de Paris, pp. 21, January 1993

3. G. Burdea, J. Zhuang, E. Roskos, D. Silver, and N. Langrana, "Portable Dextrous Master with Force Feedback", PRESENCE-Teleoperators and Virtual Environments, Vol. 1 No. 1, MIT Press, Cambridge, MA, pp. 18-27, March 1992

4. G. Burdea, D. Gomez, N. Langrana, E. Rokros, and P. Richard, "Virtual Reality Graphics Simulation with Force Feedback", International Journal in Computer Simulation, ABLEX Publishing, Vol. 5, pp. 287-303, 1995

5. H. Hashimoto, M. Boss, Y. Kuni, and F. Harashima, "Dynamic Force Stimulator for Feedback Human-Machine Interaction", Proceedings of the VRAIS, pp. 209-215, 1993

6. VPL Research Inc., "The VPL DataGlove", Redwood City, CA.

7. http://sensable.com

8. http://www.spacetec.com

9. http://www.virtex.com

10. G. Burdea, "Force and Touch Feedback for Virtual Reality", John Wiley & Sons Inc., 1996

11. Z. Huang, R. Boulic, N. M. Thalmann, and D. Thalmann, "A multi-sensor Approach for Grasping and 3D Interaction; Computer Graphics", Proc. of Computer Graphics International 95, pp.235-254, 1995

12. Y. Iwai, Y. Yagi, and M. Yachida, "Estimation of Hand Motion and Position from Monocular Image Sequence", Transactions of IEICE D-II vol. J80-D-II, No.1, pp.44-55, Jan. 1997

13. K. Funahashi, T. Yasuda, S. Yokoi, and J. Toriwaki, "A Model for Manipulation of Objects with Virtual Hand in 3-D Virtual Space", Transactions of IEICE D-II vol. J81-D-II, No.5, pp.822-831 May.1998

14. N. Shimada, Y. Shirai, and Y. Kuno, "3-D Hand Pose Estimation from Sequence Using Probability-Based", Transactions of IEICE D-II vol. J79-D-II, No.7, pp.1210-1217, Jul. 1996

15. K. Ishibuchi, K. Iwasaki, H. Takemura, and F. Kishino, "Real-Time Vision-Based Hand Gesture Estimation for Human-Computer-Interface", Transactions of IEICE D-II, Vol. J79-D-II, No.7, pp.1218-1229, Jul.1996

16. Y. Hirata and M. Sato, "3-Dimensional Interface Device for Virtual Work Space", Proceedings of the 1992 IEEE/RSJ International Conference on IROS, 2, pp. 889-896, 1992

17. M. Ishii and M. Sato, "3D Spatial Interface Device Using Tensed Strings", PRESENCE-Teleoperators and Virtual Environments, Vol. 3 No. 1, MIT Press, Cambridge, MA, pp. 81-86, 1994

18. M. Ishii, P. Sukanya and M. Sato, "A Virtual Work Space for Both Hands Manipulation with Coherency Between Kinesthetic and Visual Sensation", Proceedings of the Forth International Symposium on Measurement and Control in Robotics, pp. 84-90, December 1994

19. S. Walairacht, Y. Koike, and M. Sato, "A New Haptic Display for Both-Hands-Operation: SPIDAR-8", Proc. of IEEE ISPACS'99, pp. 569-572, Dec. 1999

20. S. Walairacht, K. Yamada, Y. Koike, and M. Sato, "Modeling Virtual Hands with Haptic Interface Device", Proc. of ICAT'99, pp 233-236, Dec. 1999

21. S. Walairacht, Y. Koike, and M. Sato, "String-based Haptic Interface Device for Multi-fingers", Proceedings of the IEEE Virtual Reality 2000, p. 293, March 2000

22. M. Sato, S. Walairacht, K. Yamada, S. Hasegawa, "4+4 Direct Manipulation with Force Feedback", Emerging Technologies: Point of Departure, SIGGRAPH 2000, New Orleans, July 2000.

23. Integration of Binocular Stereopsis and Haptic Sensation in Virtual Environment; M. Ishii, Y. Cai, and M. Sato: IWAIT'98 (International Workshop on Image Technology), Cheju, Korea, pp.67-72, Jan. 1998.

ICAT 2000

Development of a Sensory Data Glove Using Neural-Network-Based Calibration

Chin-Shyurng Fahn and Herman Sun

Department of Electrical Engineering
National Taiwan University of Science and Technology
Taipei, Taiwan 106, Republic of China
csfahn@mouse.ee.ntust.edu.tw

Abstract

In this paper, we present the development of a sensory data glove using infrared receivers/transmitters as finger-bend measurement sensors. This data glove produces nonlinear outputs that must be calibrated before it is employed in a virtual environment. To make the glove easy for use, a four-stage calibration procedure together with the construction of the calibration device is realized.

In the software calibration process, we devise a neural-network-based function approximator trained with a modified robust backpropagation (BP) algorithm that has the ability of eliminating the effect of noises in the training data. In order to speed up the training process, we propose a "tentative-and-refined" train t method that is combined with a robust BP algorithm to constitute the modified one. Many successful experiments are made on a concrete data glove to verify the effectiveness of the proposed algorithms. So far, the experimental results of the calibration process with our method are very satisfactory.

Key words: data glove, calibration device, neural-network-based function approximator, robust BP algorithm, tentative-and-refined training.

1. Introduction

In recent years, a new type of input devices, a sensory data glove, has been extensively applied along with the popularization of virtual reality (VR). The data glove is a multi-sensory device that generates a large amount of data and is more complex than other input devices. Nevertheless, most researchers still adopt this device because the natural interfacing characteristic of the data glove with the human being is the way to improve system manipulations that are applicable in many specific fields, particularly in immersive VR systems. At present, the data glove has been increasingly employed in the areas of teleoperations and robotic control [1]-[3], surgery training of medical applications [4],[5], entertainment sports of VR systems [6],[7], industrial manufacturing of CAD/CAM applications [8],[9], and so on.

Among the available input devices for VR, hand-tracking technology is the most popular one. Such glove-based input devices let VR users apply their manual dexterity to the VR activities. Hand-tracking gloves currently marketed include: Sayre Glove, MIT LED Glove, Digital Data-Entry Glove, DataGlove, Dexterous HandMaster, Power Glove, CyberGlove, VPL Glove, and Space Glove [10].

According to the outputs of sensors, the data gloves can be grouped into two classes: one produces *linear output*, and another produces *nonlinear output*. Either linear or nonlinear data gloves should be calibrated before they can be used in the applications. The calibration process of linear data gloves is directly executed by a linear mapping, but that of nonlinear data gloves is not so easy owing to lack of outputs' references of nonlinear sensors. In this paper, we present the development of a sensory data glove using infrared receivers/transmitters as the finger-bend measurement sensors. This data glove produces nonlinear outputs that must be calibrated before operation. To make the glove easy for use, the construction of a calibration device together with a four-stage calibration procedure is developed. The former creates a calibration device for a nonlinear data glove, and the latter performs an associated nonlinear mapping via a neural-network-based function approximator [11]-[13] that is trained by a modified robust backpropagation (BP) algorithm of noises elimination capabilities.

The rest of the paper is organized as follows. In Section 2 we describe the hardware construction of the data glove as well as the calibration device. In Section 3 we introduce the software calibration process. In Section 4 we present experimental results. Finally, we summarize our findings and conclude our paper in Section 5.

2. Hardware Construction

2.1 Finger-bend sensors

The finger-bend sensor is made of infrared

transmitter and receiver components that are plugged into a small flexible pipe as shown in Fig.1. The flexible pipe functions as the infrared signal transmission space. When an operator's finger is bent, the finger is-bend sensor located on the relative joint is also bent in the same shape that causes the decreasing of the radiation signal reaching the infrared receiver. This signal decrease will affect the output impedance of the receiver. Unfortunately, our experimental results of the relationship between the bend angle and the output impedance are nonlinear. Such nonlinear characteristic is affected by the bend position of the sensor. To overcome this problem, we implement a calibration device associated with a four-stage calibration procedure.

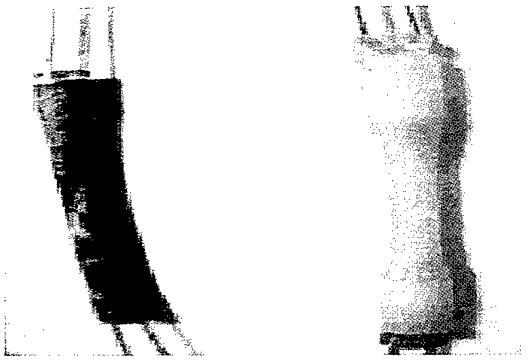


Fig. 1 The finger-bend sensors with two flexible pipes of different materials.

2.2 Fitting up the data glove

The data glove we create consists of twelve bend sensors, ten of which are located in the finger joint positions of the glove, one of the remainder is in the thumb-index abduction angle position, and the last one is in the carpal position for measuring the wrist pitch rotation angle. Figure 2 illustrates the position of each sensor equipped on the data glove.

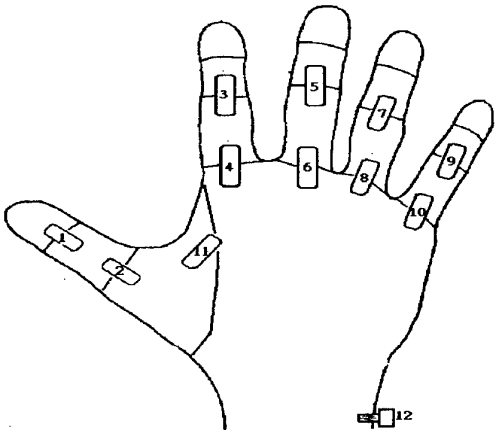


Fig. 2 Positions of the sensors on the data glove.

As shown in Fig.2, the sensor located in the carpal position is a linear one that produces linear outputs. In this case, the calibration is simply a linear mapping process. The name of each sensor related to its position in the glove is depicted in Table 1.

Table 1 The Names of the Sensors Related to Fig.2

Position no.	Sensor name
1	Thumb IJ
2	Thumb MPJ
3	Index PIJ
4	Index MPJ
5	Middle PIJ
6	Middle MPJ
7	Ring PIJ
8	Ring MPJ
9	Pinkie PIJ
10	Pinkie MPJ
11	Thumb-index abduction
12	Wrist pitch

2.3 The calibration device

The calibration device of the data glove is composed of three linear sensors. The first linear sensor is fitted on the positions of proximal interphalangeal joints (PIJ), which provides the referenced values for the four PIJ sensors of the data glove. The second sensor is fitted on the positions of metacarpo-phalangeal joints (MPJ) to provide the referenced values for the four MPJ sensors of the data glove. The last sensor is attached to a moveable stick inside a pen-shaped tube to convert the bend angles of thumb IJ and MPJ joints into a linear motion. Figure 3 shows the positions of the linear sensors used for data calibration.

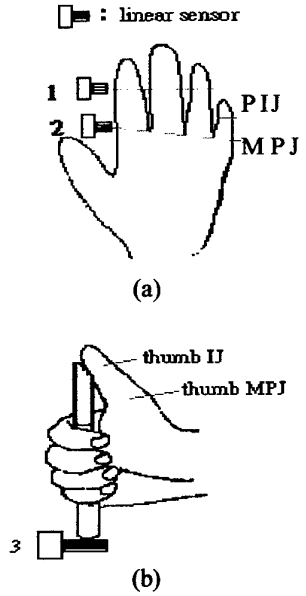


Fig. 3 Linear sensors of the calibration device positioned in: (a) four-finger PIJ and MPJ joints; (b) thumb IJ and MPJ joints.

The calibration process is executed before the data glove is employed in the virtual environment. To make it easy for use, we develop the calibration technique of four stages as follows:

1) Use the first linear sensor to calibrate the four-finger PIJ joints. This stage begins with placing the hand on the calibration device whose first sensor attaches to the middle phalange position of the index as shown in Fig.4(a). As the calibration process is started, users bend the four-finger PIJ joints to the maximum angle and then stretch the PIJ joints back to their original positions at a constant velocity.
2) Use the second linear sensor to calibrate the four-finger MPJ joints. At the beginning of this stage, the hand wearing the data glove is placed on the calibration device with the second sensor attaching to the proximal phalange position of the index as shown in Fig.4(b). When the calibration process is started, users flex the four-finger MPJ joints to the maximum angle and restore the MPJ joints to the original positions at a constant velocity.

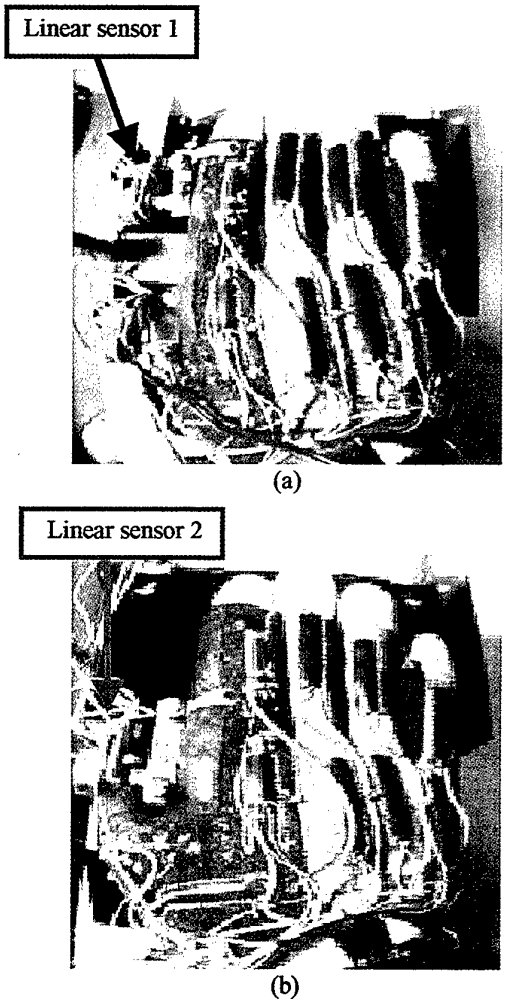


Fig. 4 Illustration of the data glove calibration process: (a) four-finger PIJ joints calibration; (b) four finger MPJ joints calibration.

3) Use the third linear sensor to calibrate the thumb IJ and MPJ joints. At this stage, users wearing the data glove grasp the pen-shaped tube and push the movable stick downwards as shown in Fig.5(a). The motion of the stick is connected with the linear sensor to produce the referenced outputs for the thumb IJ and MPJ sensors.
4) Use the second linear sensor to calibrate the thumb-index abduction angle. At this stage, the hand is placed on the calibration device with the palm facing to the left as shown in Fig.5(b). The second linear sensor is attached to the thumb distal phalange position for measuring the movement of the thumb-index abduction angle.

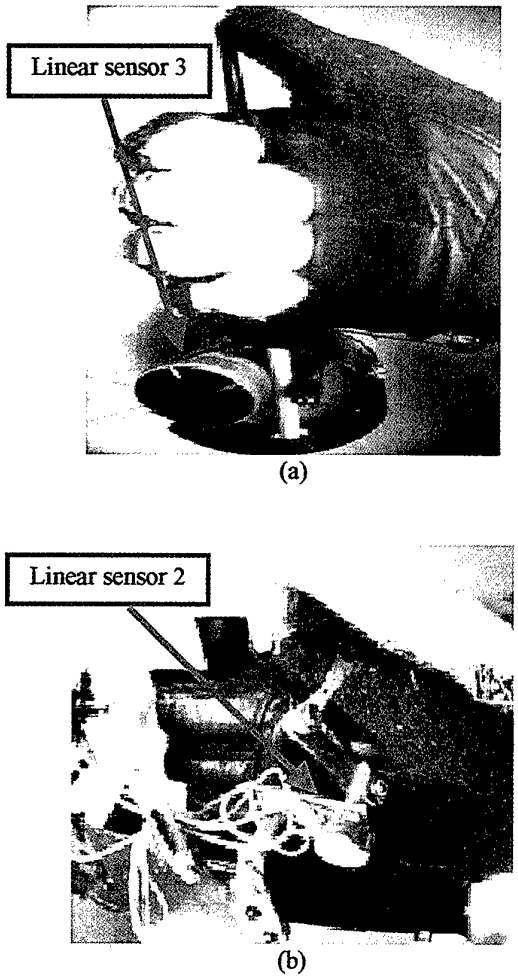


Fig. 5 Illustration of the data glove calibration process: (a) thumb IJ and MPJ joints calibration; (b) the thumb-index abduction angle calibration.

3. Software Calibration Process

After the four-stage calibration procedure is finished, a function approximator implemented by a

feedforward neural network is developed to each sensor of the data glove. The structure of the neural network is designed in the following way to provide the function approximating capability. The hidden layers of the network contain up to twenty-five nodes. It was determined experimentally for obtaining the best approximation result. In our experiments, each network consists of five layers.

The neural-network-based function approximator in the calibration process is normally trained by the BP algorithm, which acts as a nonlinear converter to map data glove sensors' outputs into cal output s i nt o ca values. The outputs of these nonlinear converters are then transformed into the finger-bend angles by a linear mapping function.

Some factors that slow down the execution time of the BP algorithm, especially when using a large amount of training pairs, are summarized as follows:

1) *The correlation between training pairs.* It means that on an average, the sampling signals do not change rapidly so that the difference between adjacent samples should have a lower variance than the variance of the whole signals. When applying the BP algorithm to train the network, we treat each training pair as an independent one that will generate conflict in the weight adjustment of the training process.
2) *The number of floating-point multiplications.* Assume that the number of floating-point multiplications needed to train one training pair is *n*, the total number of floating-point multiplications required for one iteration in the training process yields *nm* or more when the conflict is occurred for *m* training pairs in the training set.
3) *Small learning rates.* When a large amount of training pairs is adopted in the training process, a small learning rate is usually selected to prevent the conflict in the weight adjustment among training pairs.
4) *Undesired initial weights of the network.* The initial weights selected at random normally generate the outputs that are deviated from the approximated function.

To speed up the BP algorithm, we propose a Vtentati v-and-refined" train t r aini ng method . Thi s includes a tentative training procedure, followed by a refined training one. In the tentative training procedure, part of the original training set is chosen to train the network. After this tentative training, the entire original training set is employed to refinedly train the network. The motivation is based on the fact that the training speed will perform rapidly when the training set is not too large. Additionally, it can provide good initial weights for the subsequent process.

When noises exist, the approximated function behaves like a highly nonlinear one. Consequently, the number of neurons in the network should be large enough to approximate the nonlinear function. Furthermore, as the nonlinearity increases, more number of iterations is needed for the network to reach the desired error that causes the performance of the BP algorithm becoming too slow for practical uses. In most applications, it is difficult to guarantee that noises do not present in the training set. In order to eliminate the effect of noises in the training data, we devise a neural-network-based function approximator trained by a modified robust BP algorithm. This training approach combines the "tentative-and-refined" train t r aini ng and a robust BP algorithm [14]. The following describes this combination that results in a modified robust BP algorithm [15]:

Step 1: Use the first procedure of the tentative-and-refined training method to train the network until the value of its energy function $E_R = \sum_{k=1}^{K} \phi_t(r_k)$ reaches δ, where $\phi_t(r_k)$ is the integration of the Hampel's tanh estimator, r_k is the error residual, K is the number of training sets, and δ is the threshold employed to detect the time when the energy function has a sharp drop during the initial estimation.

Step 2: Reset a counter k that is used for updating $\phi_t(r_k)$.

Step 3: Compute the robust energy function: if $E_R < \varepsilon$ or the energy difference between the current and the previous iterations is less than ε_d, then terminate the learning process.

Step 4: If the counter k is a multiple of the time duration Δt between successive updates, then alter $a(t)$ and $b(t)$ which are the time-various cut off points used for obtaining the derivative of the optimal $\phi_t(r_k)$.

Step 5: Compute the error signals for the output layer and hidden layers by using the robust BP algorithm, and update the weights of the network.

Step 6: Increase the counter *k* by one and go to Step 3.

4. Experimental Results

To demonstrate the performance of our training method, we construct a feedforward neural network consisting of 4 layers with 2 input neurons, 1 output neuron, and 8 neurons in the first and the second hidden layers. The learning rate is 0.002 , the parameter □ of the activation function is 15, and the expected error is 0.000005. Firstly, the network is trained with a traditional BP algorithm. The number of iterations and the execution time required in each training process are recorded, and then compared to the tentative-and-refined training method with the learning rate of 0.005 and the expected error of 0.0005 in the tentative training procedure. In this initialization process, the

training pairs are selected from the original ones with the interval of 20 samples, including stationary points. The number of iterations and the execution time required for the above two techniques are listed in Table 2 and Table 3 with respect to 14 different experiments.

Table 3 shows the total execution time of the tentative-and-refined training method is less than that of a traditional BP algorithm, even though the number of iterations of the weight initialization procedure is larger than that of the traditional one, because of fewer training pairs participating in the tentative training stage. Figure 6 shows the output of the ring MPJ sensor of the data glove, and the output of the network trained by the modified robust BP algorithm is shown in Fig.7.

Table 2 The Number of Iterations and the Execution Time of the Traditional BP Algorithm

Exp. no.	The traditional BP algorithm	
	Iterations	Time in sec.
1	1,347	146
2	573	63
3	1,316	144
4	1,772	194
5	751	83
6	3,472	376
7	1,654	181
8	1,771	193
9	4,500	491
10	600	67
11	512	56
12	2,036	221
13	4,500	489
14	400	45
Total execution time in sec.		2,798

Table 3 The Number of Iterations and the Execution Time of the Tentative-and-Refined Training Method

Exp. No.	The tentative-and-refined training method				
	Procedure 1		Procedure 2		Total time in sec.
	Iterations	Time	Iterations	Time	
1	116	0.7	73	8	8.7
2	3,784	24	5	0.5	24.5
3	23,789	144	6	0.7	144.7
4	1,480	9	391	42	51
5	4,806	30	1	0.1	30.1
6	1,881	12	822	90	102
7	5,397	33	1	0.1	33.1
8	7,202	44	6	0.7	44.7
9	3,092	19	7	0.8	19.8
10	418	3	1	0.1	3.1
11	6,908	43	7	0.8	43.8
12	9,010	55	1	0.1	55.1
13	6,446	39	885	95	134
14	578	3	273	30	33
Total execution time in sec.					822.6

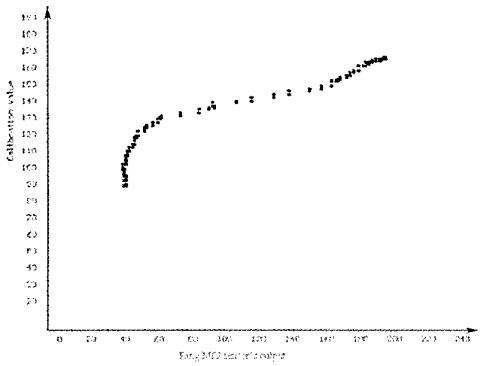


Fig. 6 The output of the ring MPJ sensor on the data glove.

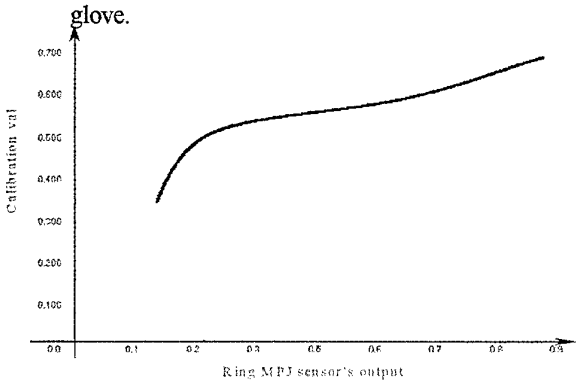
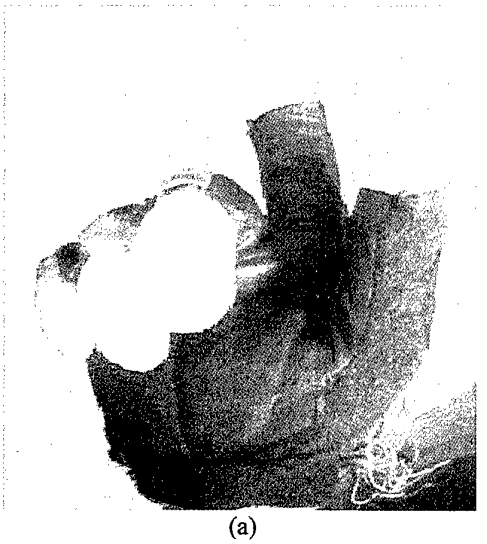
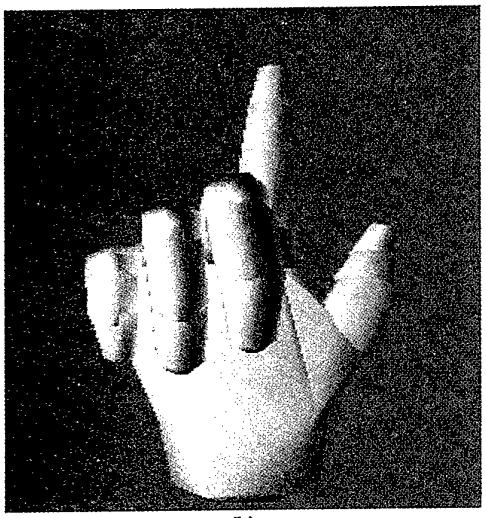


Fig. 7 The output of the network trained by the modified robust BP algorithm.

The performance of the data glove after completing the calibration is illustrated in Fig. 8.



(a)



(b)

Fig. 8 A hand gesture and the corresponding virtual hand: (a) a user's hand wearing Us hand wearing (b) the virtual hand in a virtual environment.

5. Conclusions

In this paper, we have presented the construction of a nonlinear data glove associated with a four-stage calibration procedure. In the software calibration process, we propose a new method of accelerating the BP algorithm by repeatedly training the network with different sizes of training sets that are produced by resampling the ones. We call it the tentative-and-refined training method. It can work well in the application of function approximation because the training pair generated by the sampling mechanism is usually correlated to the adjacent one. To increase the robustness of the algorithm, we devise a modified robust BP algorithm that combines the "tentative-and-refined" training method and the robust BP algorithm.

Although the data glove provides a natural way of performing a human-machine interface, it is not so convenient for the operator to use in the virtual environment owing to the presence of electric wires connecting the glove with the control device. More researches that should be accomplished in future involve:

1) *The development of a force-feedback device.* This device is attached to the data glove to feed the force back to the operator from a virtual environment. When a virtual hand touches a virtual object in the virtual environment, the force generated from the object is calculated according to the physical modeling used, and then sent out to the force-feedback device.
2) *The natural way of object grasping in a virtual environment.* In the real-time virtual reality application, the user wearing a data glove manipulates virtual objects via the virtual hand. To provide more realistic object grasping, the force generated from the hand making contact with the object should be modeled in the virtual environment.
3) *The development of a motion constraint device.* This device is employed to restrict the fingers' movements of an operator's hand when h Us hand when he grasps an virtual environment.
4) *The development of a portable data glove.* In this research, we attempt to increase the efficiency of the data glove acting as a human-machine interface, and to enhance its performance.

References

[1] C. P. Tung and A. C. Kak, "Automatic learning of assembly tasks using a DataGlove system," in *Proc. of IEEE/RSJ Int. Conf. on Intell. Robots and Syst., 1995*, pp.1-8.

[2] H. Hashimoto et al., "An unilateral master-slave hand system with a force-controlled slave hand," in *Proc. of IEEE Int. Conf. on Robotics and Automat., 1995*, pp.956-961.

[3] B. K. Sing and K. Ikeuchi, "A rob #A robot that observes and replicates grasping tasks," in *Proc. of the 5th Int. Conf. on Comput. Vision, 1995*, pp.1093-1099.

[4] W. J. Greenleaf, "Rehabilitation, 4Rehab ilita and disability solutions using virtual reality technology," in *Proc. of Interactive Tech. and the New Paradigm for Healthcare Conf., 1995*, pp.415-422.

[5] R. M. Satava, "Medicine 2001 / Medi cine 200 dead," in *Proc. of Interactive Tech. and the New Paradigm for Healthcare Conf., 1995*, pp.334-339.

[6] S. Kawamura et al., "Development of a virtual sports machine using a wire drive system- a trial of virtual tennis," in *Proc. of IEEE/RSJ Int. Conf. on Intell. Robots and Syst., 1995*, pp.111-116.

[7] D. Gromala and Y. Sharir, "Dancing with the virtual dervish: virtual bodies," in *Proc. of Virtual Reality Software and Tech. Conf., 1994*, pp.321-328.

[8] M. W. S. Jaques, P. Strickland, and T. J. Oliver, "Design &De si gn by manufact uri ng sim concurrent engineering meets virtual reality," in *Proc. of Mechatronics Conf., 1994*, pp.637-642.

[9] M. W. S. Jaques and D. J. Harrison, "Using gestures to interface with a 'virtual manufacturing' package," i *Proc. of the 3rd Int. Conf. on Interface to Real and Virtual Worlds, 1994*, pp.231-240.

[10] D. J. Sturman and D. Zeltzer, "A survey of glove-based input," *IEEE Comput. Graphics and Appli.*, vol.14, no.1, pp.30-39, 1994.

[11] K. Hornik, M. Stinchcombe, and H. White, / Multi layer feedfor ward net works are uni versal approximators," *Neural Net.*, vol.2, pp.359-366, 1989.

[12] G. Cybenko. "Approximati #Approuperposition of a sigmoidal function," *Math. of Contr., Signals, and Syst.*, vol.2, pp.303-314, 1989.

[13] K. Funahashi, "On the approx l On the approxima of continuous mappings by neural networks," *Neural Net.*, vol.2, pp.183-192, 1989.

[14] D. S. Chen and R. C. Jain, # A robust back propagation learning algorithm for function approximation," *IEEE Trans. Neural Net.*, vol.5, no.3, pp.467-479, 1994.

[15] S. C. Sun, *Development of a Sensory Data Glove with Neural-Network-Based Calibration*, Master Thesis, Dep. of Elect. Eng., Nat. Taiwan Univ. of Sci. and Tech., Taipei, Taiwan, 1998.

The proposal of an interaction design based on self-awareness: Toward the reformation of self-realization

Isato Kataoka[1) 3)] Atsuhito Sekiguchi[2)]
Katsunori Simohara[1) 3)] Michio Okada[1) 4)] Osamu Katai[1)]

1) Dept. of System Science, Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, JAPAN.
2) IAMAS, 3-95, Ryoke-cho Ogaki City Gifu 503-0014, JAPAN.
3) ATR-Internation, 2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, JAPAN.
4) ATR-MIC, 2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, JAPAN.

isato@sys.i.kyoto-u.ac.jp

Abstract

A computer installation that enables people to rediscover their own identity is proposed. In materially affluent societies, people seem to have lost the perception of their meaning and identity as well as the relationships with others and their societies. In that society, there are unexpected outrages, such as like murders committed by teenagers, as shown in recent years of Japan.

We believe that it is indispensable for people to be conscious of their own identity and meaning so that they can not only build relationships with others but also acquire their own identities. This thought became the purpose of our artwork.

In order to assist our own self-realization, this artwork has two focuses. One is the perception of our own physical body's usual role, so-called in Japanese "Shin-tai-sei". The other is the subject's developing process. At first we create an alter-ego. The "alter-ego" is the ego that is presented at outside of our own. Secondary, based on the autopoietic theory, we postulate subjective-self is formed of mental, nervous, physical and social system.

We anticipate the interactions with the alter-ego and an expression of the development about subjective-self make us recognize the our own meaning and identities.

Key words: self-awareness, self-realization, lost the self-existence, social entities, embodied media, social bonding, interaction design

1. Introduction

The human being, as a social entity, longs for relationships with others and seeks meaning. In that sense, we could postulate communications as a "form of relationship with others." However, can modern people be conscious of their own existence and meaning in their relationships with others?

Although living in today's materially affluent societies appears to bring us much happiness, unexpected outrages, such as like murders committed by teenagers and treatments small animals cruelly, occur frequently in recent years of Japan. Moreover, parents have murdered their children because they were confused about their responsibility. In so-called "competition-obsessed" societies, people experience a compulsion to pursue wealth and profits. On the other hand, along with the tide towards a so-called cyber society, there is a chance that people may not distinguish between real and virtual worlds. But there is some controversy about whether people feel ease or difficulty in developing social relationships or self-realization in such a society.

Originally the person's worth belonged to physical body, like a running fast, cultivating the land and eating, as shown in Figure1. But in materially fluent and high informational society, human has a tendency to set the worth on not only physical body but also the wealth and profits. As shown in Figure2, if the worth does not return to own-self, the person may create unexpected self or lost own-self. In recent years of Japan, it becomes hard to create a self-reference from social system. For this example, some youth has very high ideal, if the ideal is out of the bounds of possibility, they have not seen themselves objectively. The divergence between ideal and reality bring them mach obstruction that is unable to conquer and disappointment.

This is why people have lost their perception of self-existence as well as the relationships with others and their societies. Some people lost coordination to others and society. Therefore, to guard own ego from the divergence of external system, people increase mental strength with egoism or ethic. At the result the person, who lost mental fitness, have tendency to need intensity information. These points cause much destruction in people's minds.

In order to confront this problem, we believe that it is indispensable for people to be conscious of their own existence so that they can not only build relationships with others but also acquire their own identities. Based on this belief, we are working on communication design in human-to-system interfaces. As the initial step of this research, we propose an interactive system that enables people to rediscover their own existence through a series of interactions between people and the system.

2. Forming self-awareness

In order to assist our own self-realization, this artwork has two focuses. One is our own physical body's routine role that is called "Sintai-sei" in Japan. And the other is the subjective-self's developing process, as shown in Figure1 and 2. We think the process does not only consist of innate biological data. The example of the innate biological data is DNAs and the characteristic of body and so on. Although these data can become parameters of that process, therefore we put emphasis on the subjective external and internal interaction.

At first we create an "alter-ego" for subject's interaction. The alter-ego is the ego that is presented at outside of our own. Secondary, based on the autopoiesis, we postulate subjective-self is formed of mental, nervous, physical and social system.

2.1 Focus on physical body

In order to assist people in rediscovering a new relationship with each selves, the idea is to postulate the physical body as another entity separate from the subjective-self, and to use the physical body as a medium that makes a user rediscover his/her own subjective-self. If the subjective-self were his or her spirit, the physical body would be the space around the subjective-self. Thus, the physical body can become a medium that makes the user rediscover his/her own existence because it is another entity separate from the subjective-self in the nearest sense. Based on this believe, we propose an "alter-ego". The alter-ego is subject's ego. It is made from the subject's vital data and physically exists in front or outside of our own. This approach is different from the former language based philosophical one. This system does not only transmit "something" to the subjects but also provides a space that represents the entire environment. In short, the space as a medium should include the subject as the subjective-self.

2.2 Focus on subjective-self

As the former philosophical discussions, we suppose the subjective-self to be existing with the external and internal interactions. On cognitive science field, there are attempt to build the mathematical human-behavior model with autopoiesis, coupled chaotic, dynamical system approach.

The autopoietic system, which maintain their defining organization throughout a history of environmental perturbation and structural change and regenerate their components's external and internal interactions in the course of their operation to define whole system.

The psychoanalyst reports that the unnecessary duplication cause a double or triple personality symptom. Formerly such a symptom was treated as a serious illness, but the analysis based on autopoietic theory regard the symptom as general problem because people have any self-images.

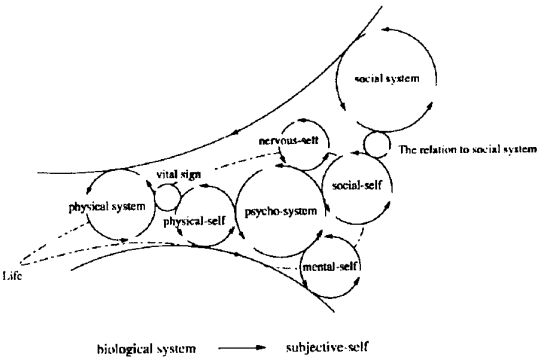


Fig. 1 Person's worth belong to physical body.

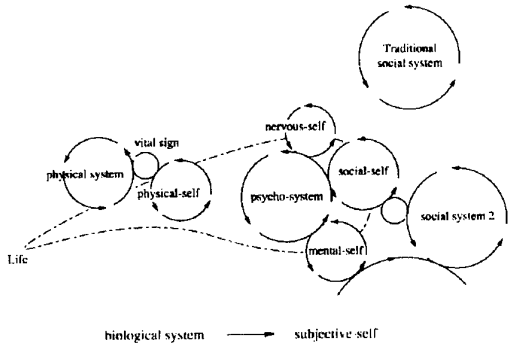


Fig. 2 The influence of new society

Based on the autopopietic theory, we postulate the subjective-self developing mechanism has four elements as shown in figure1 and 2. At first the self-awareness that is determined by our mind is the mental-self. Secondly, the self-awareness that is unable to determine is nervous-self, although the nervous-self should be treated as self-observation or self-diagnosis that is so called in Japanese "Naikan". It is one of Chinese therapy. Thirdly, the self-awareness based on real feeling and external perception is physical-self. Finally, the self-awareness based on social rule is social-self.

2.3 Exercise on this installation

For the awareness of subjective-self, this installation tries to keep non-contact as far as it is possible. So we regard the subjective-self developing mechanism as the basic system. In advance we have to say this is not for medical treatment and analysis machine because we establish this installation as artwork. However, the devotion to this artwork makes people cast away an aching heart and urge people to aware the generation of subjective-self. For that aim, this artwork presents the information for the subject to observe the subjective-self objectivity.

3 . Explanation of our artwork system.

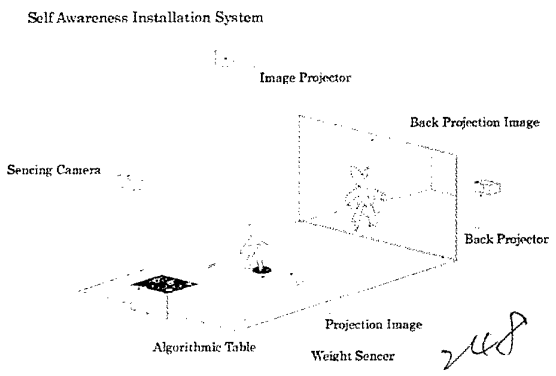


Fig. 3 System outline

At first, in the exhibition entrance, "Algorithmic-Table" is established. In the table, we establish a certain society by an artificial creature algorithm. The table's surface consists of computer display panel with touch-sensor. So experience person is able to specify their position on the society by own finger. This behavior has influence to whole of the system and starts this installation.

There is marked area that it is called "WeightSpot" in the center of this exhibition space, and the experience person stands up first in that center. The moment it stands up, the capturing the person's outline and measuring the weight of the person in the beginning.

Then, water that is same quantity with the measured weight spreads out on the floor by computer graphics. An expression by this volume of water is handled as sensory massiveness of the self-body of the experience person. Shape and the nature of the material are changed, and it is made to indicate the data, which it has this image inputted to from the input from the captured image and AlgorithmicTable in the cause by the influence of the basic system.

On the backside wall, an alter-ego is projected as the image of the experience person who had capturing done by this, and recorded in different time. As shown in Figure 4,5,6 and 7, particles organize themselves into a pattern that represents the subject's alter ego.

The interaction between the subject and the alter ego is prepared in advance as shown in Figure 6 and 7. In performing this installation, the alter ego not only affects real events but also generates unexpected interactions. In other words, his/her physical movements stimulate the mental and nervous systems revert to the physical body.

The series of alter-ego express the personal history. It is used for the cause as an own reproduction. In this installation, by touching the alter ego, the phase bifurcation program is activated as shown in Figures 6 and 7. When the program starts running, the relation between the subjective-self and the alter ego starts collapsing. Thus, the subject is expected to become

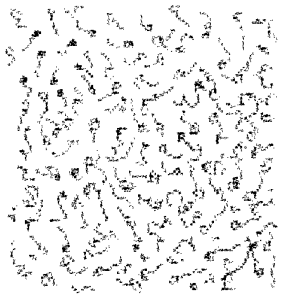


Fig. 4 There are particles that have no meaning and chaotic motion.

Fig. 5 Subject's investigation makes the particles organize into the alter ego.

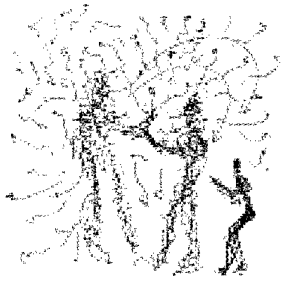


Fig. 6 When the current alter ego touches the former one, the phase bifurcation is activated.

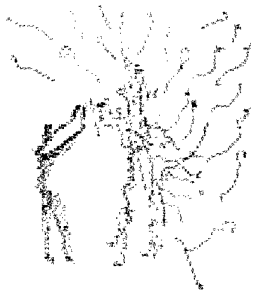


Fig. 7 When subject touches the current alter ego, alter ego fade away, then only the subject remains.

conscious of the subjective-self and the alter ego.

The alter-ego image changes shape and the nature of the material, too, and makes it indicate the data inputted in the same way from AlgorithmicTable in the cause by the influence of the basic system.

The interactions with the alter-ego stimulate the mental and nervous systems revert to the physical body. We postulate that these interactions are effective in the self-realization's problem. Because these presentations are desirable to keep or recover the balance of developing own self-image. Thus, the self-awareness that is disengaged from realities is reformed to self that is real exists.

4. Conclusion

In this paper, we have proposed an experimental system to investigate people can find the awareness of the subjective-self's generation through interaction with alter ego and the subjective-self's model.

The interactions with the alter ego stimulate the mental and nervous systems revert to the physical system. Then the sensory massiveness that is disengaged from realities is reformed to realize. This physical approach is different from the former language based philosophical one.

We anticipate the external and internal contact to alter ego and the subject's developing systems make us reproduces a sound boundary of own self. In short, the contacts and interactions create a subjective-self.

The awareness of self-realization is one of the foundations for emergent communication mechanisms, not only to activate cyber-social interactions between people and information but also to enable people to find diverse relationships. One of the prospective directions of this research will be to study behavior-awareness using a computer installation system with which people can find the meaning of their own behavior and its effect on the relationships with others. We hope to create an environment where people can develop diverse relationships with others and naturally form social bonding with them.

References

1. Michio Okada: Issues on reality in talk-in-interaction, "Shintaisei-to-computer" pp.220-232, Okada, Mishima, Sasaki eds, Kyoritsu pub.(2000)
2. Michio Okada, Shoji Sakamoto, Noriko Suzuki: Muu: Artificial Creatures as an Emboded Interface, "SIGGRAPH Emerging Technologies: Point of Departure." pp. 91.(2000)
3. Isato Kataoka: Make an experiment with an inflated balloon, "IAMAS annual 1998" pp.22-23.(1998)
4. Isato Kataoka: Make an experiment with an inflated balloon, "IAMAS annual 1999" pp.112. (1999)
5. Michizo Noguti:"Noguchi-taiso, karada-ni-kiku", Hakuju pub.(1979)
6. Shizuo Takiura: "Jibun-to-tanin-wo-doumiruka", NHK pub.(1990)
7. Niklas luhmann : "Essays on self-reference" Columbia U.P.(1990)
8. Hanamura Seiichi: "Bunretubyou-sei Jittai-teki Ishiki-sei", "BunretsuByouRon-no-Genzai" pp.147-186. Kobundou pub.(1996)
9. Toshihiko Nagata: "Bunretubyou no Shitubyoukann to chiryou", "BunretsuByouRon-no-Genzai" pp.187-202. Kobundou pub.(1996)

A Primary Study on the Design of an Immerse Campus

Peisuei Lee* Shou-Yen Lin Uh-li Su Tzuchin Chen Ding-Wuu Vale Wu
Sheng-Chi Yu Bin-Shyan Jong Yuan-Liang Liu Yuan Kang
International Academy of Media Arts and Sciences*
3-95 Ryoke-chou, Ogaki-city, Gifu 503-0014, Japan
Chung Yuan Christian University
Chung-Li 320, Taiwan, R.O.C
*peisuei@iamas.ac.jp**, *yuankang@cycu.edu.tw*

Abstract

This paper studies a user to access a lot of interactive sign of the virtual world by using a fixed-screen system and on the web site. An immerse campus views and platform motions make the user's senses lik Us senses l ike dr i v real campus. According to lot of simulation and examination to understand the influence of each sense for help to design the immerse effect with a virtual world. Especially, this paper focuses on the design of an immerse campus by using the cognition of each senses from access the motion platform and on the web.

Keywords: immerse effect, a fixed-screen system, motion platform, cognition, and design

1.Introduction

This paper describes how to design a virtual world with immerse effect. According to the simulation, the sense of seeing, hearing, and feeling from driving a virtual world can be combine lots of basic immerse effects. The user also can be seating in the front of a fixed-screen or monitor. It allows to using a set of applications like as drivers. The driver likes using whole-body navigate through an immerse campus. The campus can be driving on the web and effectively doubling or tripling from access lots interactive signs.

The feature of this paper shows how to design those interactive sign based on the cognition of examination. According to the cognition of examination, this paper improve the immerse effect generation [1] combined lots cognitions of each sense of seeing, hearing and the touch. This paper also shows the immerse effect not only generating lots virtual objects [2] but also coordinate by lot of interactive signs. The finally goal of an immerse campus design for users trying to approach more really and naturally.

2.The Virtual Environment

The Virtual Environment (VE) of this paper included a fixed-screen system with display screen and "Motion pla /Mt i as shows on fig.1. The figure illustrates the construction of

The merit of using the display screen of a fixed-screen system shows the size of virtual objects as same as real objects · Users can be seating in the front of the display screen like as driver seating in the motion platform navigating a real campus. In this case, using two axes of the motion platform is enough to drive on the road and save the cost of three axes.

On the other hand, the motion platform also can be access from the web and realized the user communicates between a virtual world and real world.

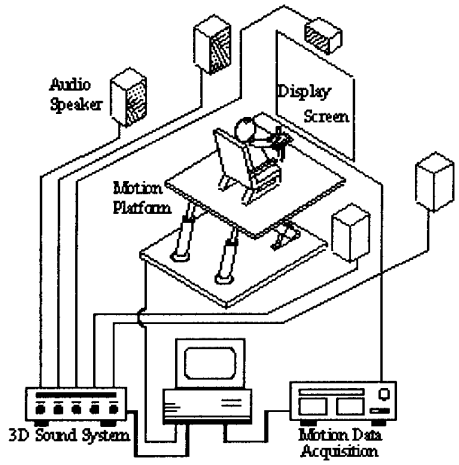


Figure 1 "Motion Platform"

3. A Virtual World

According to the marketing research can be understood the most important perception from the sense of seeing, hearing, touch. The beginning of a virtual world generation of this paper refers to a real campus on the Chung Yuan Christian University.

The fig.2 shows a view of the "Visual Sign" for the sense of seeing. The fig.3 shows a seen of the "Sound goods" for the sense of hearing. The fig.4 shows a Conner of the "Force feedback" for the sense of touch.

A virtual world of this case prepare the visual sign lead the driver entry a real campus. Driver can be go through or turn the direction of moving and toward or back. On campus driver can be ring the bell to access the sound goods. In the middle of road driver also can be receive the felling of meet lots rocks from the force feedback.

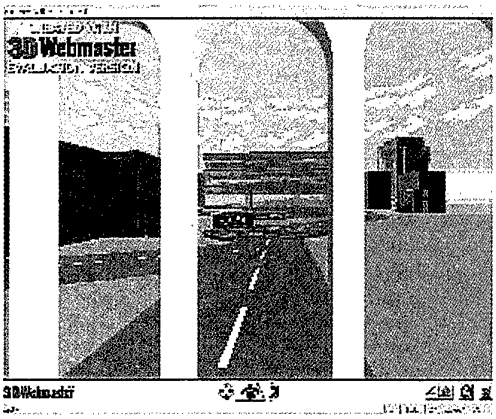


Figure 2 8 Yrt uagoods"

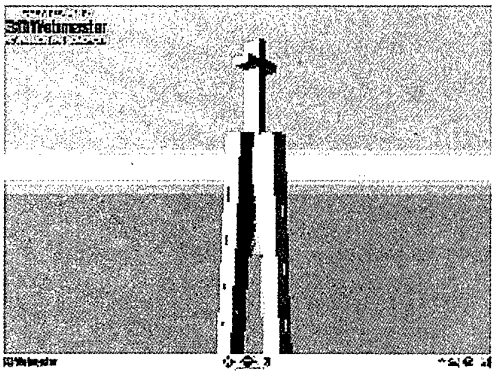


Figure 3 "Sound goods"

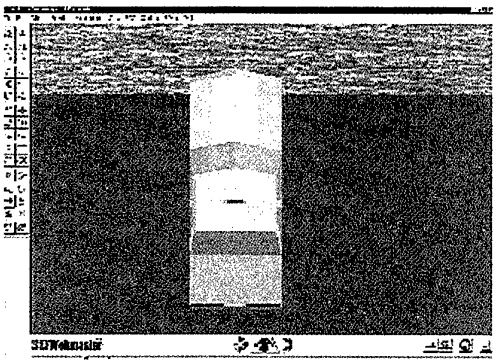


Figure 4 "Force Feedback"

4.Simulation and Distribution

This simulation for each one of over twenty students drive navigate through this virtual campus on the desktop and motion platform can be received the different cognition of lots key words. On this paper focusing the most high point of three key words of satisfy, really and smoothly. According to the three key words simulate it on the desktop and motion platform as shows it on the fig.5-10.

On the desktop received the different level of each sense of seeing, hearing and touch as shows on the fig.5-7. According to the simulation, analysis the different cognitions of each sense of contents, direction and operations. The fig.8 shows different result from increase the objects of contents, influence the cognition of satisfies on the sense of seeing. The fig.9 result shows set up the different direction and distance for different cognition on the sense of hearing. The fig.10 also shows renew result can be understood the different from the data of operation.

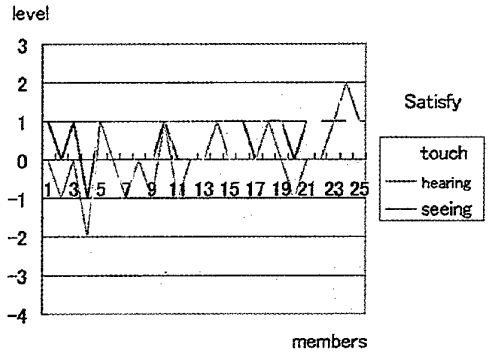


Figure 5 Pre-fixed Satisfy

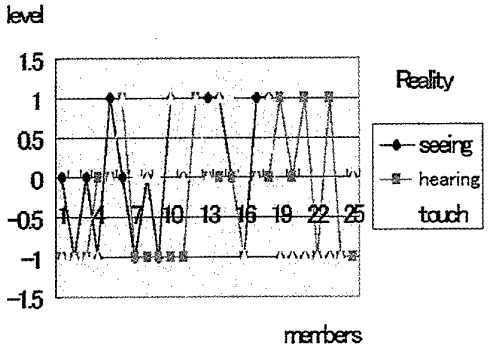


Figure 6 Pre-fixed Really

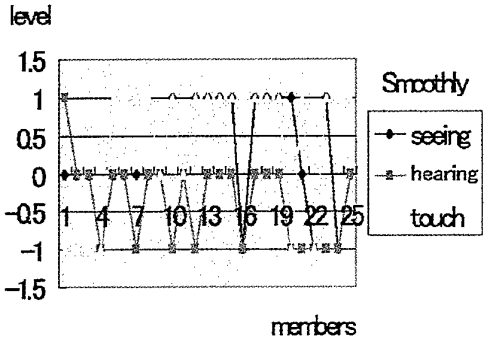


Figure 7 Pre-fixed Smoothly

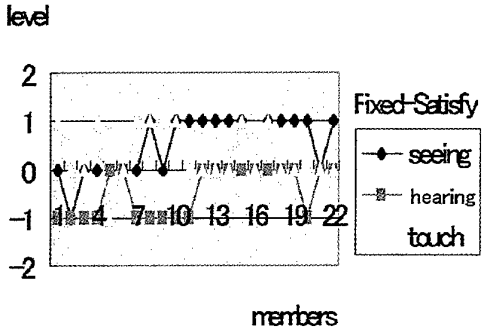


Figure 8 Fixed Satisfy

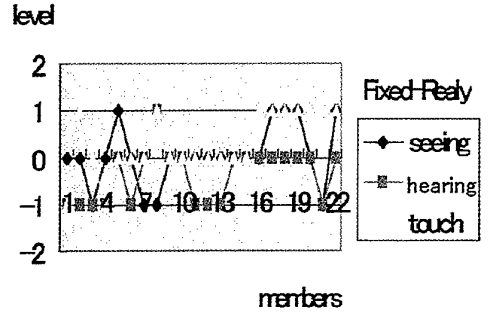


Figure 9 Fixed Smoothly

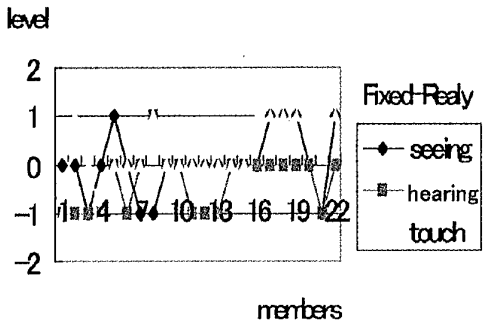


Figure 10 Fixed Really

5. An Immerse Campus

According to the result of simulation from the motion platform, fig.11-13 shows the fixed image of lots virtual world scenes.

On the visual, users can be access the fig.11 like as seating in the car driving in a real campus. On the hearing, users also can be getting off from the car to ring the bell of the fig.12. On the touch, users can be received different feeling from access the fig.13 navigating the all campus on the road meeting different objects.

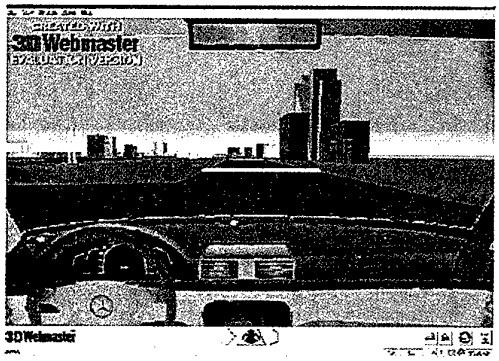


Figure 11 Fixed Virtual Sign

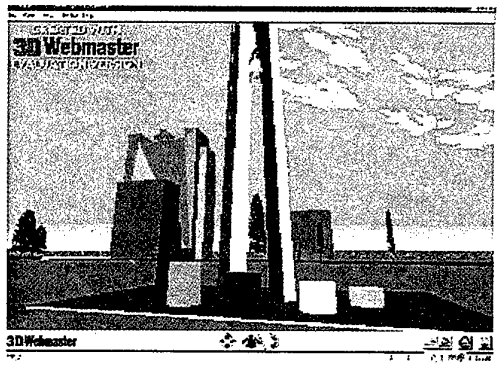


Figure 12 Fixed Sound goods

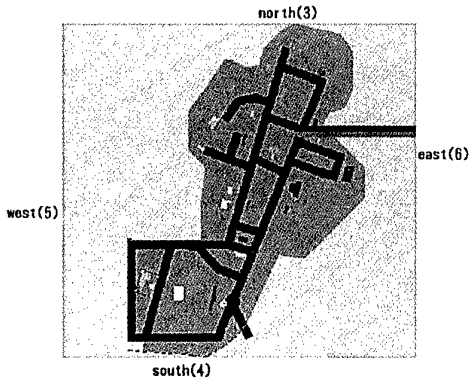


Figure 13 Fixed-force feedback rote

6. Conclusion

The most important challenging task is the way shows design an immerse campus on the virtual world combined the cognitions of the senses of seeing, hearing and touch from the feeling of driving the motion platform and on access it he web. Also the performance of "Motion platform" can be influence the user's sense.

Reference

1. Massamari Kanagawa, Kiyoshi Kiyokawa, Haruo Tekemura, Naokazu Yokoya, "A Study on the Effect of Time Lag on Direct Mainpulation Virtual Objects", pp.563-572, TVRSJ Vol.14 No.3, 1999.

2. Masahiro KIMURA, Makoto TAKAHASHI,KazutakaMITOBE,TohruKATO,Yuichi ISHIDA, Hiroyuki NARA, Tohru IFUKUBE and Katsuyuki YAMAMOTO, "A Basic Study on Combined Effect of Audiovisual Sensory Information on the Sense Equilibrium", pp117-183, TVRSJ Vol.3 No.3 1998.

The Monologue Opera - The New Era

Masahiro Miwa Shinjiro Maeda
International Academy of Media Arts and Science
3-95 Ryoke-cho, Ogaki-city, Gifu 503-0014, Japan
{mmiwa, maeda} @iamas.ac.jp

Abstract

"The New Era", a monologue opera, was performed for the first time in Kyoto in April 2000, with a further performance in Tokyo. The opera is an experiment in integrated performance system construction, which by means of several computers realizes in a theatre setting real time operation of control systems, images and sound. At the same time the opera aims to question the people of the present era who constantly seek the new technology that is embodied in the work.

1. Introduction

There are currently a number of stage experiments being conducted under labels such as 'multi-media theatre' and 'cyber opera' that combine the artistic fields of dance, theatre, film, music and so forth, which themselves make the most of new technologies. This monologue opera (a monologue opera being an opera with only one lead actor/actress) falls into the same category as the above stage experiments. It is widely accepted that it is in western opera tradition that we find the most representative examples of the 'integrated arts'. Regardless of whether this monologue opera is a work that inherits these traditions or another style of composition, firstly it is an opera. It is a device to entice the audience who share the performance space into a different world, by using for the most part visual and aural stimulation. This work was born out of the desire to try to express symbolically the 'present', technological society that we live in, while keeping past traditions in mind. Technology is not just a tool to facilitate the performance, but the most important theme of the opera.

The synthesis of the human voices by the 'performance' of the band pass filter and the melody creating algorithms are the special features of the composition. However the opera is also an example of where MAX/MSP, a development tool which I often use for relatively personal music creation, is utilized to realize a 90 min stage work and also control, on stage, the sound as well as imaging and lighting.

I would be grateful if the readers of this introduction can use it as an opportunity to consider the expressions of technology and the present era, rather than just an introduction to the background of the opera.

2. Outline

This opera is about a fictitious religion, it is a mass for the believers of the church of the 'The New Era'.

Background that can't be conveyed during the opera:

The digital network that is connecting the world on a global scale is spreading like wildfire. It is becoming a book of knowledge that can gather and process huge amounts of information, a brain for the whole world that has abilities that far exceed that of individual humans.

Before long this brain will begin to address the areas in the book that have been left blank by humans, it will start to automatically generate the secrets of the creation of the universe and the origin of life, and will release these secrets as a secret code onto the internet.

There are people who intuitively understand the meaning of this strange code that floats around on the internet. They have learnt that the code is directly conveyed to humans not by letters and words, but by transforming the code into music and sound. They have also discovered that it is something that was already understood by humans belonging to ancient civilizations.

People have exchanged this knowledge, and with further intuitiveness a faith in the internet, a religion has been born.

The gospel of this religion preaches that one should suicide, that is, do away with the flesh and attain a purely spiritual existence. It states that the objective of human existence is to offer oneself as a sacrifice to God, the principles of the universe, and become part of the glorious melody.

This melody, or sound has the magical

ability to stimulate to the maximum parts of the human brain that are not usually utilized. It also has the awesome power to awaken what has been handed down as a genealogical record from ancient civilizations that have long since ceased to exist.

The melody is the only expression comprehendible to humans of God's language, something that we normally cannot understand. To do away with the physical body and become part of God's melody is the final goal for humans, a goal that is not attainable by other biological beings. Originally, human language was also structurally regulated in the same way as "God's language".

3. Content:

This mass, an initiation ceremony for death is the most significant life event for the believers.

The mass is a real space where believers share the music and sound that is created from the strange code that comes from the internet, a solemn place where it is possible to listen directly to God's language through the music.

The ceremony centers upon the leading actor who will offer himself up as a sacrifice in front of the other followers and is held according to the following order.

1. Alleluia - All present listen to God's melody performed by virgin maidens.
2. Archive - The appearance and genealogical information of the actor are captured and sent as data onto the net (and left there).
3. Reincarnation - The leading actor drinks some poison and while waiting to die sings God's language, thanks to God and his confession. In order to confirm that the believer who died during the ceremony has become part of the melody, data of the believer will be recalled with a matching part of the melody and remembered amongst feelings of respect and envy.

The leading actor for the opera, the mass, is a fourteen your old child.

4. The Stage:

The following features of the opera are made possible with the use of specially developed systems and several computers including:

- A sound generation system based on the performance of four female keyboard players (principally a performance in keeping with sine waves);
- A video imaging system to synchronize the musical score of the four female keyboard player s performance and the progress of the music;
- An imaging system to project prepared images and images from several (infrared) cameras on stage onto a large screen;
- A delay system to delay the video images on the stage;
- Control systems to control other sound sources and lighting etc.

The electronics and the network fully utilize the latest technology. They make the stage both complicated and intricate and are themselves a form of the opera's expression.

What we see, hear and our existence in present day society is questioned by three means: the video delay system which replays the video of what occurred on stage ten seconds beforehand, the infrared cameras which convey what is occurring on the stage when viewers can hear but can not see what is going on with the physical eye due to the darkness, and especially the synthesis of voices in keeping with the melody of overlapping of noise and sine waves. The above techniques used during the opera also symbolize the strange transparency and danger of existing in a 'mass media' society.

5. The Progression of the Story

The opera progresses on stage as follows:

Prologue

0 Before the performance. The icon of the order is slowly revolving on the 'screen', and there is a 'noise' sound to give the audience the sensation of the presence of God.

Part 1 Alleluia

1a The young boy faces the audience (the believers who are attending the ceremony) and says a greeting.

(Note: the young boy is actually played by the female soprano Reisiu Sakai, and also referred to as the lead actor or the believer).

- He faces the 'lantern' and chants incantations in a quiet whisper. The four virgin maidens begin to chant in response.

1b The young boy picks up the poison and water that have been prepared for the

ceremony from the 'pedestal', and sings the "Song of Joy" without accompaniment.
- He enters the 'device', puts on the headphones and begins to meditate.
1c The maidens perform the "The Coming of the Holy Spirit" while watching the score (score projection onto the 'lantern') that has been transformed from data on the network. (31mins)

During this time the words from the vowels that symbolize God's voice with the four melodies from the keyboards are synthesized. The believers who have already left this world through this ceremony area called back by a point set in the melody and their appearances over lap (flash onto) the musical score that is projected onto the 'lantern'.
- The performance finishes with incantations in a quiet whisper.

Part 2 Archive
2a The young boy recites <Fixed verse no.1> as determined by the ceremony.
- The maidens perform 'God's Melody' while looking at the score projected onto the 'lantern'.
- The young boy says <Confessions no.1 - 7> with 'God's Melody' in the background.
- He then recites <Fixed verse no.2> with 'God's Melody' in the background.
2b The "capture" of the believer's (the young boy) personal information takes place.

The young boy sings to the melody cited by God that continuously floats up as white noise. His voice is sampled, and then over sampled using a delay system (transformed into a multi strain harmony) and is converted into data. The neume type notation (or a similar notation) that is rapidly flashed on the 'screen' is symbolic of the personal information of the believer that has been converted into data.
2c The 'reason' for the ceremony becomes apparent as the young boy recites a monologue and another fixed ceremonial verse, with sounds symbolic of the voice of the believer (the young boy) in the background.
- The boy closes his eyes and drinks the poison.
2d He takes off the headphones and leaves the 'device' with the laptop that he used in the confession.
- He verifies the image of himself drinking the poison that has been projected onto the 'screen' with the video delay system.
- The lead says <Words of Thanks> to the other believers (the audience) that confirms the archive process (conversion of his personal information into data) has been a success.

Part 3 Reincarnation
3a 'God's Melody' is played with sounds symbolic of voice of the young boy.
- The young boy sings the <Secrets of Angels> with 'God's Melody' in the background.
- He then unfolds the laptop computer on the 'pedestal'.
- He returns to the 'device' and gradually begins to lose consciousness.
3b 'God's Melody' becomes sin waves part by part, and the performance of the melody becomes automatic.
- The maidens cease playing as the performance becomes automatic, leave their places on the 'alter', close the curtain on the 'device' and leave the stage.
3c The message from the young boy to the believers (the audience) automatically begins to play on the laptop computer.
3d The young boy's image is projected onto the 'screen' with the infrared cameras, he opens his eyes.
- The boy sings <The New Era> with 'God's Melody' in the background.
3e The young boy completely leaves this world. The image of the boy disappears and is replaced by the icon of the believers, and then stars.

The Cast:

The characters that appear on stage during the opera are as follows:
* A fourteen year old boy believer (the lead character) - soprano
* Four virgin maidens who are in charge of the ceremony - female keyboard players (4 players)
* Believer 1 - operator responsible for visual effects
* Believer 2 - operator responsible for sound effects
* Believer 3 - mixing operator

The four keyboard players are the 'orchestra' responsible for the music in the opera, and at the same time perform the role of the maidens who take charge of the ceremony on stage. The three operating staff who exist as 'clerics'

in the background of the religious ceremony concentrate on opera operations and do play an active acting role.

Stage Set-up:

On the stage the 'device', 'alter', 'screen', 'operation counter', PA speakers, projector and 'pedestal' are arranged in the following fashion.

Stage Layout Chart (Refer to Fig.1)

The rectangular parallelepiped frame with curtains on three sides is called the 'device'. It is the machine that extracts the personal information from the believers and transforms it into data. Inside the 'device' there is a chair, a small table, a head phone amplifier, an infrared camera and PA speaker, which are arranged so that they are not easily visible.

The 'alter' is the space where the four keyboard players perform. In the middle of the 'alter' there is a 'lantern'. Inside the 'lantern' there are small projectors that project images in all four directions (mainly music scores). This area is the 'alter' in the opera, of which the 'lantern' is the center of attention.

The 'lantern' is an acrylic fiber four-sided column that has semi transparent paper pasted on all four sides, it is empty on the inside and has no lid. It is used as a screen to project images (mainly musical scores) in all four directions. Inside there is a lamp controlled by computer to let the performers know the rhythm of the music.

The 'screen' is a 3m by 4m back projection screen.

The 'operation counter' is where all the computer, visual and sound systems are situated. Using these systems the three operators conduct real time operations according to the progress of the opera.

Visual Systems:

The following visual systems are used as images sources during the opera: four video cameras, a camera for filming the 'lantern', infrared cameras for filming the keyboard player's performance (two cameras), an infrared camera for filming the vocalist inside the 'device', together with two video players, and a video image replay computer with random access. The images from the above sources are modified with both relay and hand held switches and then projected onto the 'screen'. In addition to the above there is also a computer specifically for video delay operations, and a computer that is used to display the sound waves produced by an oscilloscope from the characters voices.

Imaging Signal Connection Chart (Refer to Fig.2)

PA System:

The PA system is a four channel speaker system built around a Yamaha O3D mixer board. The four speakers are not quadraphonic, they are placed in specific locations on the stage such as near the screen and where the vocalist is standing (please see the stage setup diagram). Natural balance and orientation is achieved using the volume of the vocalist's voice as the standard.

All the sound inputs are sent to the YAMAHA O3D mixer and leave through the Bus out and Aux out as the operator carries out the appropriate routing and mixing according to each scene of the opera. The inputs include the microphone in the vocalist's hair, the sin waves produced by computer from the performance of the four keyboards, white noise, the output from the software sampler, narration prepared on CD-Rom and also the metronome click that is sent to the vocalist's headphones.

Sound System Chart (Refer to Fig.3)

Sound System:

The sin wave sounds produced on stage by the keyboard players and computer algorithms, and the filtered white noise are the raw sound materials characteristic of this opera. The melody in the music score used for the keyboard performance is generated according to the same algorithms that are used for real time generation. These algorithms can be divided into two main groups, which have been called the algorithms used for speech synthesis and the algorithms used to generate 'God's Melody'.

There has already been an explanation of the algorithms for speech synthesis that realize formant synthesis through the performance of the keyboards, so there will be no further

explanation here (please see reference materials: Literature: Word Shadows (Kotoba no kage) or Alleluia introduction).
The algorithms used to generate 'God's Melody' are extremely simply constructed algorithms that generate four 'voices' by picking musical notes according to random numbers. However, the special feature of the algorithms is their small range, in other words, they have been created so that the different 'voices' do not simultaneously choose the same note from the small number of notes available. Consequently the 'voices' may choose the same note consecutively, thus sounds are produced that have few changes, few developments but are continuously moving. 'God's Melody' is performed partly in Part 2, 2a with sin waves generated from the performance of the keyboards, in Part 3, 3a with the keyboards and the sampled voice of the vocalist, and in sections 3b and 3c with the automatic performance generated in real-time by sin waves.

MIDI Control Systems:
During this opera two Mackintosh computers are used to control lighting and sound generation and management, in which MIDI control systems play a central role. The computers have been called the "RacMac", and the "DeskTopMac" and are set up as follows. All control functions are split between these computers and are carried out with the MAX/MSP patches that have been installed.

"RacMac"
Model: PowerMac9600 G3/233MHz upgrade
+ 196MB memory
+ 4 * Xclaim VR video board (PCI cards)
+ Korg 1212 I/O audio card (PCI cards)
+ MidiTimePiece II (MIDI interface with serial connection)

"DeskTopMac"
Model: PowerMacG3 233MHz
+196MB memory
+ Korg 1212 I/O audio card (PCI card)
+ MIDI Translator (MIDI interface with serial connection)

The functions of the MAX/MSP patch installed on the "RacMac" have been divided into specific modules according to their function:

* AOclock.m : the control of the lamp inside the 'lantern' and the timing of the overall systems that control the performance output information that comes from the keyboards.
* AOscore.m : the control of the projection of the musical scores.
* AOsound.m : the control of the algorithms and generation of the white noise and sin waves.
* Delay8000.m : the video delay and the delay of the timed music in Part 2 (2b).

The "DeskTopMac" is used in Part 2 (2b, 2c) for the following:
* The sound generation and control of the white noise and the sampled voices.
* The control of the lighting in the 'lantern' and the 'device'.
* The metronome click and the control of the tuning note sent to the vocalist through the headphones.

Other computers used are:
PowerBookG3:
Used in Part 3 (3a) for sampling voice (software sampler) of the keyboard performance.
PowerBook5300
This computer is not used for the main performance but rather to simulate the performance of the keyboards for rehearsals and sound checks.

MIDI System Chart (Refer to Fig.4)

The Final Word:

This opera was commissioned and conceptualized in 1992 by its sponsor The 22th Century Club. Actual work on the opera began in 1996, in 1999 the following independent compositions were combined to complete the opera with the help of one of the performers, Shinjiro Maeda.

"Silhouette of Words, or Alleluia" based on "A's" text
"Neue Zeit" for 50 iMac's and an operator
"Neue Zeit" for two organists and an assistant with a Mega-Phone
"19-sai no shi" for two pianists and a computer

Each of these works has their own theme that was either modified or disregarded to suit an

operatic composition, and then integrated into the theme that continues throughout this opera. However, a great deal of the musical materials and the computer algorithms in this opera follow suit with the above works.

Credits:

monologue opera "The New Era"
Composer, scriptwriter, computer programmer:
Masahiro Miwa

Director, visual effects:
Shinjiro Maeda

Special thanks to:
Masayuki Akamatsu

Supported by:
IAMAS (International Academy of Media Arts and Sciences)

Sponsored, produced and commissioned by:
The 22nd Century Club

First Performance at 'Alti' Kyoto on April 20. 2000
Second Performance at Kioi-Hall on April 27. 2000

Soprano:
Reisiu Sakai

Keyboards:
Kaori Iimura, Aya Usutomi, Takae Kikuchi, Tomomi Mitsui

Sound technicians:
Tomoko Ueyama, Masatsune Yoshio

Visual technicians:
Akio Okamoto, Takaaki Shimbori

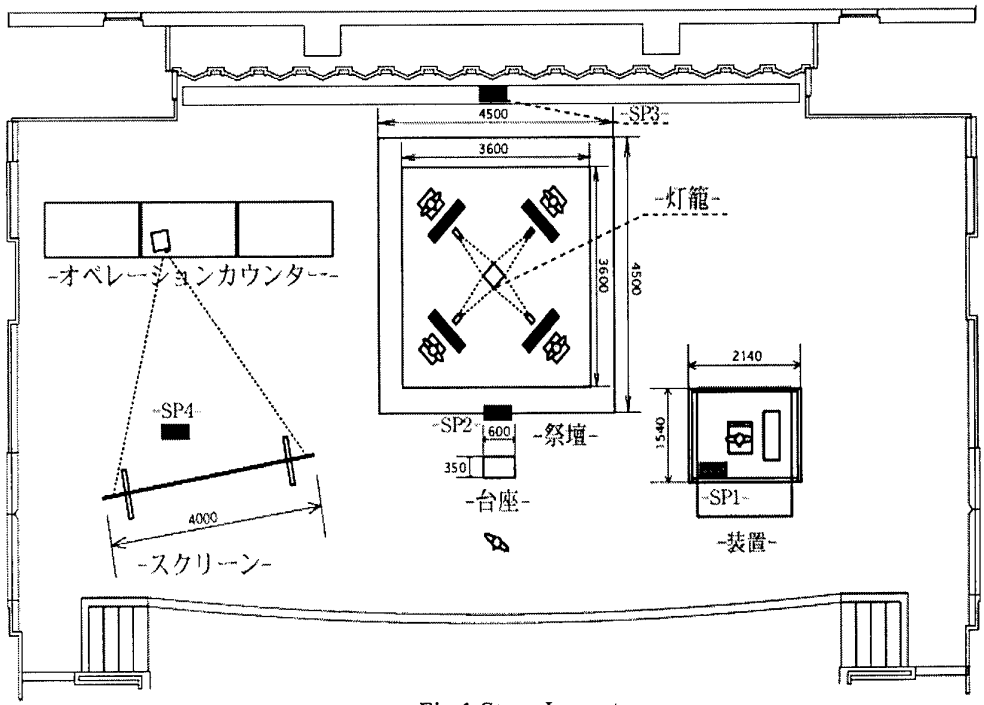


Fig.1 Stage Layout

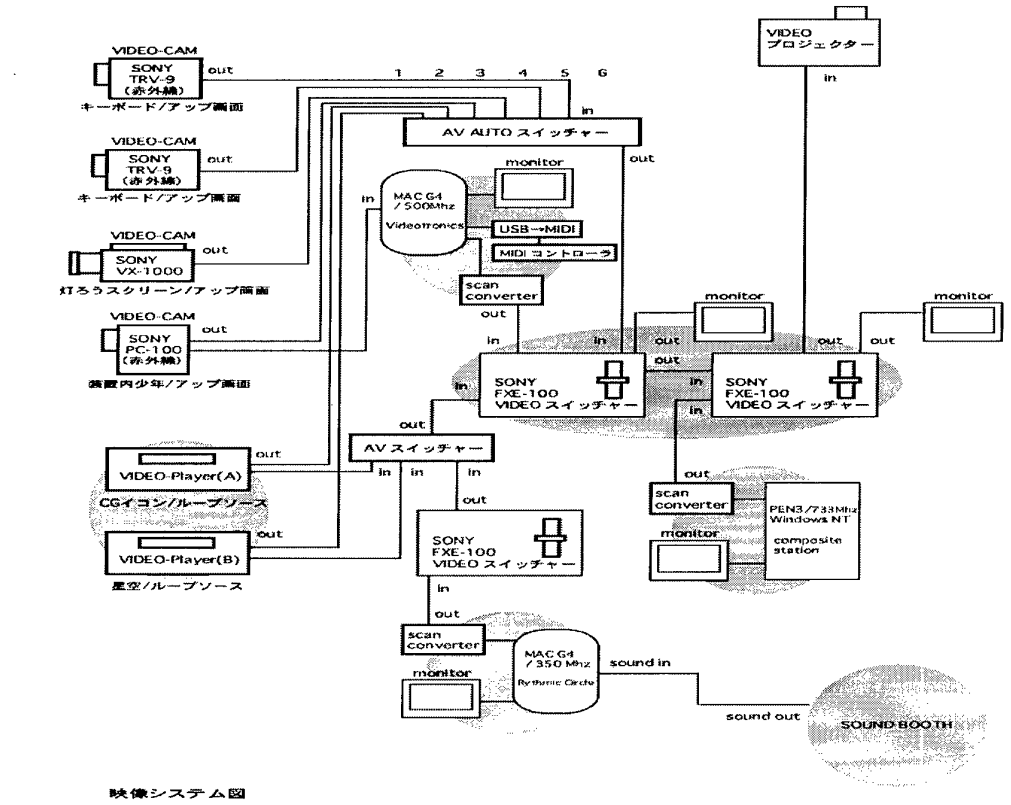


Fig.2 Imaging Signal Connection

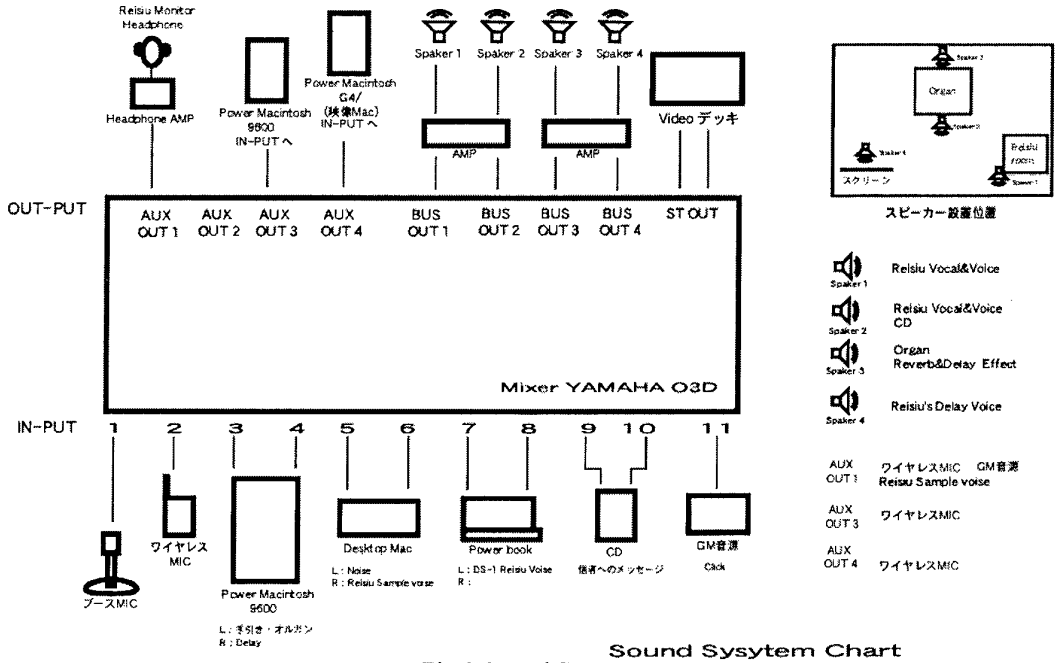
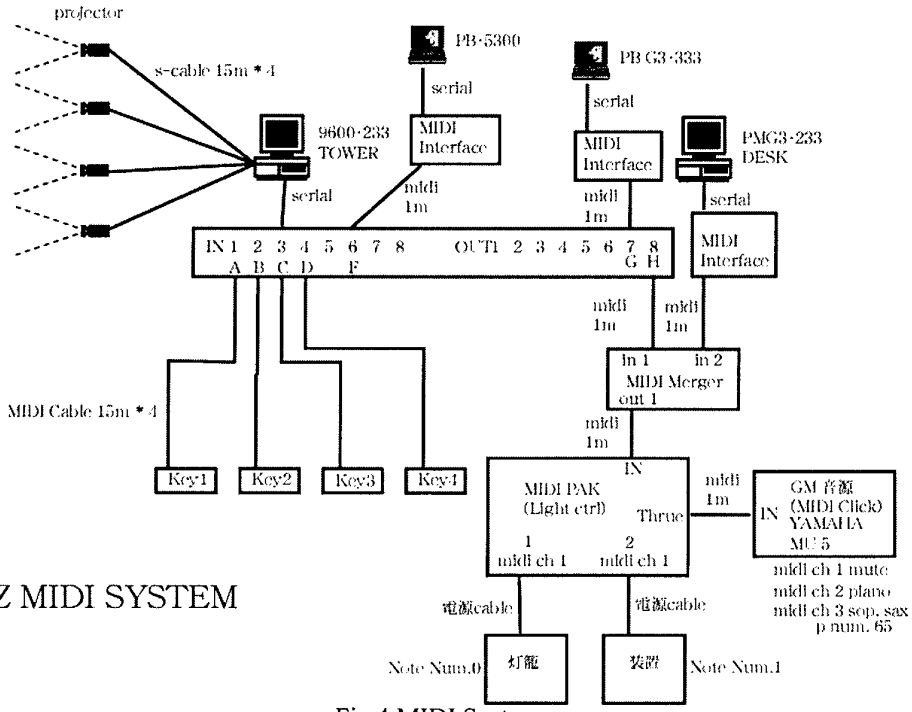


Fig.3 Sound System



NZ MIDI SYSTEM

Fig.4 MIDI System

Trends of Music Composition on Computer

Dye Wu
National Taiwan College of Arts

1. Stepping on the road of ancestors to go on our way

The birth of computer has been directing the entire social development, and accomplishments are found from technology to all levels of social applications.

Back in the early 50's, composition on computer has been started in Cologne Music Center to search new sounds with the electronic devices and to bring new life in composition. Computer has become a new device in music composition over these decades. Like the IRCAM (Institut de recherche et coordination acoustique/musique) at the Centre Georges Pompidou in Paris, the CIDAM (Création ingénerie diffusion des musiques d'aujour'hui) of the École nationale de musique de Pantin in Paris, the Institute of Sonology at Hague in the Netherlands; the New Music and Musicology Research Center at Darmstadt in Germany, the CCRMA (Center for Computer Research in Music and Acoustics at Stanford University, and the computer music research centers at various US universities. They focus on computer composition and bring music to the forefront of times.

It is evident that composition on computer is a future trend. If you want to catch up with times and to possess a world musical view, you music give up the prejudice of the non-human device of computer and thereby to adapt to and combine with computer while composing.

2. Review the history of music and develop new musical concepts

From a historical approach of music composition, the style, the representation, the spirit and attitude of music have been changing from time to time; and the search for arts of different times never stops, it is the spirit to change and to be difference that create the bright history of music. Technological development also brings positive reinforcement to music. Modern musicians or musicologists should ease the worry that music will turn stiff when synthesizer or computer are used as devices in musical compositions.

Hence, we should abandon the conservative attitudes and concepts, and to think about the future trend of music composition from the review of history to readjust the pace of musical advancement.

2.1 Changes—the natural development of music

From the evolution of music, it is easy to discover that ancient music is simply monophony with focus on human voices or rhythmic instruments. It is not until the appearance of polyphonic organum in the Renaissance that other instruments were added to musical works, though harmonious acoustics was still the focus, and dissonance or rather complex effect was rare. The structural elements of music have changed, and readjustment on the concept of timber thus followed, and the acoustics of old musical instruments no longer satisfy the needs of musical works,

and to reengineer the sound of musical instruments has been the goal at the moment. Arcangelo Corelli gave a new life to viol and redefined the string instruments of modern days.

Corelli not only revolutionized the technique of violin playing, but also established the formal structure of musical works, and paved way for the formalism in the classical period. The changes in music in a product of the changes of the thoughts of musicians, *l'air du temps*, and social style. Though formalism is the central of the classical period, Ludwig van Beethoven changed the fate of classicism to make music speak the me of the composer.

The wheel of time endlessly runs through the times and brings up changes to music, enables the 'authentic music school' to rule the musical world for over two centuries. To most people 'authentic music school' refers to 'traditional music', 'authentic music', it sets up the standards for tonality, rudiments of music and central of music. However, just when music grows in its high, Arnold Schöenberg breaks all rules with his dodecaphony, and tonality thus becomes a historical term.

Sound is the basic element of music, the changes in social environment altered the musicians' appeal for musical representation. They depict the quietude and simplicity in peaceful music, they scream out their troubles in romantic music, and modern musicians even chase after noise in music and begin to think what noise could bring to audiences. They started to search for diversified timber changes and musical texture. Traditional musical instruments no longer satisfy their desires for timber and effect, and computer thus become an easy-to-use device.

2.2 Novelty—musical representation

When noise is the target of modern music, it is not difficult to discover that novel music has been used in the revolution for the novel age. In face of the impacts of these consciousness, we must feel the necessity of computer in music composition, and it is a breakthrough of the traditional concepts.

To enable music to express the sense of times, futurist music makes use of the noises of machines or factories together with the rhythms to convey the pace of modern life, some musicians even create noise instruments of the city. French composer Edgard Varèse is one of the most eminent figures who dedicates in this field immersing in the magic and mystery of science.

The effort of Varèse alone is difficult to emancipate noise, to develop electronic music and to promote urban sounds. American composer John Cage is a good partner to continue these concepts. His attempt is shown in the *prepared piano* in the late 30's by putting different objects such as metal plates, wood blocks or rubber between the strings of the piano, in hopes to create 'sounds of indeterminacy' and to test new timbers.

Besides changes of timbers, the *prepared piano* is an imitation of percussion timber to express the rhythm of music with both hands without the need of a huge and complex percussion band. It is the same way as conducting a huge orchestra with a computer.

French composer Olivier Messiaen is another musician who puts much emphasis on rhythm. He considers rhythm is kind of number and continuous change. In his grammar of composing, he puts special focus on changes of dotted note

value, expansion and reduction of rhythm, and irreversible rhythm and compound rhythm. He uses numeric calculation as changes of these rhythms, which resembles the calculation functions of computer, and it is the specialty of computer.

Dissatisfaction of the timber of traditional musical instruments and the search of new timbers is the mission of this period. The birth of magnetic tape after the 40's has added a new device in music creation and has given way to 'electronic music'. Just before that, the idea to change and to process natural sounds have been a fact in the test of sounds. In following the discovery of sound wave, musicians produce some special sounds they want to bring to music.

Electronic technology has thus become an important component ion musical art and the drive to the computerization of music. Computer is a product of the USA in the 60's, and many American composers thereafter try to use computer to synthesize sounds or to change sound by using the calculation functions of the computer to process the values of sound wave, in order to produce the ideal sound on the musicians' mind. Computer even enables composers to enter the rules of composition to create a new work itself. In this sense, who or what is composing? The issue immediately arouses heated debate. In reality, computer simply follows the rules dictated by composers to write music, and it is still composers who created the rules. The modern Romanian composer Yannis Xenakis is a typical example. He uses computer as a calculation device in composing and thereby creates a milestone in music creation.

2.3 Search—changes in music

History tells us that all musicians follow the mainstream of their times, they seek changes and new things, aiming to broaden the scope of changes of music. Of course, to innovate and to change are the sole missions of a composer.

In addition, we discover that the trends of times and social changes are the voices of human beings. As a representation of soul, music needs changes to satisfy the needs of times and people. Internal changes require cooperation of the external structure, and the use of computer in analysis, logic calculation and integration will mean a piece of cake.

3. Computer—device of musical creation

In the 50's and the 60's, computer was an aide to composers and never took over the process of composing itself. The contents of creation have been defined within the human scope of thinking.

The process of composing is an act of acoustic organization, a logical arrangement, connection and grouping of sound. The surface structure of composing is a set of rules; however, it is the thoughts of composers in the deep structure which is conveyed and represented through the rules of organization.

Ideas are implicit and abstract, they must be represented by means of physical music, and the rules of musical representation are simply means of creation. Ideas and well-organized structure are the artistic requirements of a musical work. Musicians in the past followed these rules and attempted to make changes to create new rules, and it is the same to modern composers: to seek new musical grammar and new direction out of the tradition. At the

turn of the century when technology is so advanced, the use of computer as a device in composition will be a natural trend. When composers take the advantage of calculation of computer to express their ideas, it will be mean to kill to birds with one stone. When computer is a helpful hand to composers, it can do things far and wide. Besides composing, it can perform and create the novel timbers to open a new door of creation for composers.

However, there are many technical problems to solve to let computer participate in composing. A composer must cultivate his programming ability before he can convert his thoughts into programs; he must think his ideas well before he can assign the task to the computer.

Musical creation is a process of sound arrangement and grouping; calculation is the specialty of computer, the conception of sound and the calculation of computer will be a perfect match, when a composer is able to tell the computer his rules. In other words, when a composer can convert all the rules into programs, the computer can compose itself. Therefore, one must get ready the following points before composing with a computer:

(1) Establishment of a musical grammar database, including

Interval data: all kinds of natural intervals, changes, harmonious intervals, dissonant intervals, monophonic intervals, polyphonic intervals, melodic intervals and harmony intervals.

Rhythm data: to collect all kinds of simple rhythms, compound rhythms, aboriginal rhythms and changeable rhythms.

Register data: to analysis the register of each part of mixed voices, child voice and the register of all instruments.

Musical pattern data: to categorize all kinds of musical patterns that represent different musical thoughts to let the computer to access. It is a rather important item, because it presents the ideas of creation and musical thoughts of a composer, and it is a tough work.

Sound progression data: different sound progression represents different sentiments, if we can categorize these rules and store them in a database, it will be a good reference for the computer.

(2) Polyphony method

Contrapuntal technique requires a set of rules, a composer should create a these rules in the database to enable the computer to follow and thereby to establish a good discourse.

(3) Melodic music method

It is a harmony technique. As harmony rules are very strict, when we use computer to compose, we must create such sets of rules to enable the computer to follow. The categories are:

rule database establishment;

use of simulation;

trail of new harmony effect.

(4) Multi-tonality method

Multi-tonality is different from polyphony, though it is sometimes a complex of multiple melodies, each melody has its own tonality and grammar. Bela Batók and Darius Milhaud are few of the very best of this technique after Arnold

Schöenberg. After one has understood the basic concepts of multi-tonality, he can use it on the computer to link up melodies of related tonality. In fact, the organization of parts is a coordination of tonal relations. Therefore, tonal relationship is the focus of establishment of multi-tonality technique, and its technique can be accomplished by the coordination of the previous musical grammars.

(5) Atonal music technique

The result of multi-tonality is atonality. When there are too many of them, it becomes a chaos, and the subject will disappear. Single tonality requires the emphasis on the tonic, around which everything runs. When there are more than one tonics, any note will be equally important, and the tonic and dominant relationship will collapse, and tonality will become atonality.

Capture of sound organization and rhythmic changes are the prerequisites to the use of computer in atonal music composition. The full use of all items in the musical grammar database, the creation of free arrangement of atonal rules, a composer may modify the product from the simulation to complete an atonal work.

(6) Serial music technique

Serial music, technically speaking, is different from atonal music. The latter emphasizes on no tonal center and free groupings of tone rows; while the former must strictly follows the non-repetition principle of dodecaphony and allows only retrograde, inversion, and inverted retrograde of the original. Such technique is the product of the second Viennese school: Arnold Schöenberg, Alban Berg and Anton von Webern. It is the foundation of the search for composition with computer.

From the principles of tone row, it is an easy work for the computer. A composer simply converts the tone row rules into programs, he can compose anything on the computer, even take advantage on the float-point calculation feature of the computer to facilitate the task.

(7) Experiment and application of timber changes

Different forms of arts have different textual elements and manners of representations, and it is no difference to music. It is its sound nature which makes it an art of invisibility and time, no sound no music (the *4'33"* by John Cage is an exception, it is a recollection of the indeterminacy). Sound is therefore the basic element of music. If we are observant enough, it is no difficult to discover sounds are everywhere. If we can make full use of them with careful processing, they will turn out to be wonderful music.

Same sounds appear in this world in different forms since the ancient time, whether pleasant or not, it is subject to the view of times. Changes of environment will change the views of sounds, which enables much variation in music. Review on traditional technique to match with new sounds, new texture and new

structure will be the subject of modern composers.

Life itself is an art, and environment is the source of condensation of art. Modern composers should understand we are immersing in the endless resources of creation. With the caress of technology, we can turn any impossibilities to possibilities. Sound is the gift of God to composers, and how to combine them with the computer in a reasonable way and turn it into an true art of music will be the central concern.

Sounds in the nature are in fact a systematic and well-organized single-tone grouping, if we can sample and analyze them with the computer, it is easy to discover that they are groupings of different single sounds. From the viewpoint of music, simple sound line resembles different parts of a work. A reasonable arrangement of these simple sound lines can make beautiful music.

If computer is the gateway to see the wonder of noise, how to integrate them to formulate a logical musical structure will need some efforts:

A. All single line sounds can be treated as different parts of a work, e.g., we hear bird calls, vehicle sounds and crowd noises in everyday life, if we can mix each of them together life different parts of a musical work, it is the same way as we hear them in the nature.
B. The basic component of different sounds can be treated as a sound pattern of a work: each individual sound has its own way of grouping, i.e., they may be cyclic, continuously repeated or changed repetitions. Each cycle resembles different sound pattern in a musical work, and the continuous expansion of the line will form different musical lines. For examples, the calls of a certain kind of bird is simple and clean, if it is a regular call, it will become a kind of pattern. Like the hammering of a blacksmith or the sound of a machine, the regularity will form a certain kind of element.
C. The distance and strength of sounds in nature are the dynamics in music: within an acoustic space, the utterance of sound has a sense of distance, man hear weaker sound when he is far away from the source, and the closer the louder. Additionally, the sound itself has its strength, it will cause tension on hearing, and it is indispensable from musical expressions.
D. The tension and relaxation of sound are the structural features of rhythmic and tempo tension of music: different sounds have different paces, from the viewpoint of tension, it resembles the changes in note value and tempo and thus forms the tension and of time and creates the agogic in music.
E. Intermittent strong sound means the staccato in music: if a sound is intermittent, it resembles the staccato in music, like staccatissmo, mezzo staccato,

portato. these resemble to intermittent sounds.

F. Synergic strong sound resembles the forte of harmony: if we put all the single sounds together, the effect is like a harmony of thick texture.

Besides collecting sounds from the nature, modern composers can use the sounds from a synthesizer to show the special effect in a musical work, or like what Richard Wagner did when he created the Ring—he even created new instruments for the special sound effects. It can be achieved on a computer. Microtone is the special feature of computer, and it is irreplaceable by any instrument.

4. An account for *Phenomena and Concepts 1001*—for voice and computer

This is the product of physical life experience and music interaction, with an attempt to express the truth of life through tone rows and to tell the philosophy of life. Music is a reflection of human life and a representation of everyday life. Through the combination of music and media, it may be able to reveal the truth of life. We often think that life is a time machine, musicians feel that and music is the pulse of the time machine. Phenomena in daily life have their own time and space, and music is the variables such times and spaces.

Music is a continuation of sound and time, the leaps and floats of sound form a music line, then row and space. Bagatelles in daily life are different points in time, and thus forms endless concepts. The changeability of these concepts turns into endless sounds, which help stimulate the eternal dreams and associations of audiences.

Music is the basic element of the work, after individual creation of the music, it is combined with multimedia and voices and actions on the stage.

The work is inspired by pointillism, besides hearing, these points give an visual effect to create an endlessly deep space.

Points on the surface is extended with different instruments on different tracks, the pitch is expressed horizontal on the picture, and the size of the points represent the dynamics of sound to reflect the sense of space in the float of time.

Hardware media: Macintosh computer, PROTEUS/2 XR Orchestral mixer, computer overhead projector, stage projection backdrop, Midi interface, amplifier, stereo speakers.

Software media: Max and external from IRCAM, integrated with HyperCard.

5. Conclusion

Modern man has modern mind to create the new times; modern scientists have modern wisdom to reveal world miracles; modern musicians should have modern views to accept impacts of new thoughts, to establish new musical environment and to review new directions of music creation.

In this diversified society, different phenomena have condensed countless concepts and have brought endless space of thinking.

Media Installation "Hide-and-Seek "

Kumiko Kushiyama 1) and Shinji Sasada 2)

1)Department of Literature, Waseda University

1-24-1,Toyama Shinnjuku-ku,Tokyo,162-8644 kushi@ea.mbn.or.jp

2)Computer Graphic ,Japan Electronics College

1-25 ,Hyakuninn-chou,Shionjuku-ku,Tokyo,169-0073

sasada@cg.nippon.ec.ac.jp

Abstract

Hide-and-Seek is a future interactive dining table. Viewers walk around a dining table carrying a portable television. On one channel, they can find hidden images that mix real and virtual spaces. In this interactive installation art work we present a creative Mixed Reality application .Our interest is creating an imaginably architectural space, between real life space and virtual space. We try to interpret the present space included a life and an human communication..

Keywords: Interactive Reality, Hand-held Display, Furniture, Communication Art, and Mixed

1.Introduction

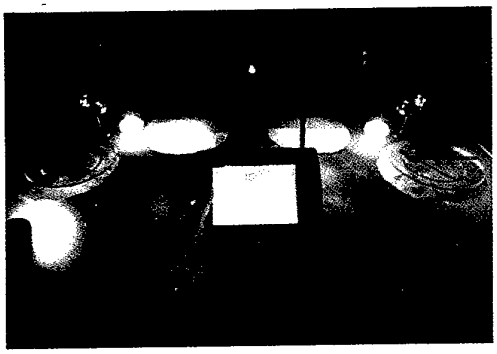
In this work, we present a creative Mixed Reality dining table. As for this work, there are three points. VR technology is applied currently in the various scenes. but, the implementation ground as a creative work is limited to a part of

contemporary art museum or amusements such as a game center. Our interest is creating an imaginably architectural environment. We present future furniture as the work that VR technology was applied

In our daily life. Among other things, the dining table symbolizes the most usually scenery of our life. In our present age life, there is the state that the situation becoming vague had a thing of reality and distinction with a virtual thing already, and reality and imagination were mixed by advanced information technology in one scenery of world. In this work, we present a creative Mixed Rarity. We express a state of boundary line that reality and imagination were mixed as work concept technically and artistically. They can find the hidden images that mixed real and virtual images from a dining table and Hand-held sole display.

We suggest thirdly new form of communication. On a table, we look at scenery of dining table which virtual conversation of man and woman as a message version was indicated. And the audience used a Hand-held sole display. They can find the hidden images that mixed real and virtual images from one TV channel. In our daily life we want to enjoy a space useing Mixed Reality (MR) System. So far, MR systems have been based on large and heavy and expensive computer systems. We imagine that If we will get a MR system channel on the daily TV program ,we usually can experience MR environment . We propose a MR system using a mobile computer and a potable

television. The audiences share in physical space and virtual space made by mixed media installation. They can find the hidden images that mixed real and Virtual images from one TV channel.

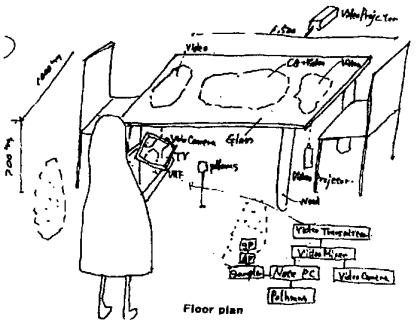


[Fig.1] 'Hide-and-Seek' in 2000

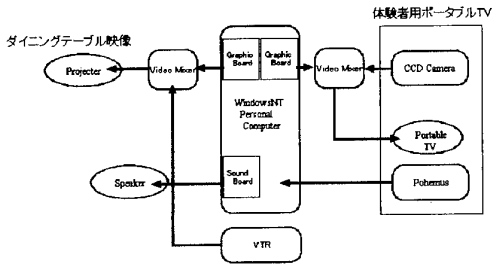
2. System Overview

We set a real dining table in space. We make some imaginary holes in a real Dining table, showing real daily life images and virtual images marked computer graphics. The audiences walk around with a potable television in hand. They can catch the hidden images that mixed real and virtual images from one TV channel. Interactivity Audience can enjoy a change of image between real and virtual by the sensor.

As system constitution, I consist of three of event department to read a position of small monitor which a spectator has by a picture of character on a table and a small monitor., a picture of character forming three dimensions picture projected on the basis of information from event department, sound output doing feedback by a sound the image creation department.



[Fig.2] Sketch 'Hide-and-Seek' in 2000



[Fig.3] 'System'

2.1 event department

By magnetism sensor (polemics) installed on the back of 6inch monitor which a audience had. We can get information about measures an angle and a three-dimensional position of Hand-held Display that an audience peeps out. The event department transmits a message to the image creation department. Indication of sound output goes simultaneously, too, and does total control as a mainstay of system.

2.2 strokes image creation department

In the image creation department, two kinds different images are formed on the basis of information transmitted a message from an event department at real time. It is mixed images with video picture and one of picture formed with a computer, are reflected beforehand by a tableside.

On the back of the Hand-held Displays, small CCD video camera is established. As a picture in the Hand-held display (6 inch monitor), which a audience has, it is Real time interactive image made by mixed images, and real video

camera. Mixed images made by digital video mixer.

2.3 sound output department

There is the sound output for character of picture appearing as feedback for experience person except a picture. I form MIDI data, and I utilize the origin of MIDI sound and as for a sound, it is output a sound by the indication signal, which has been sent from an event department.

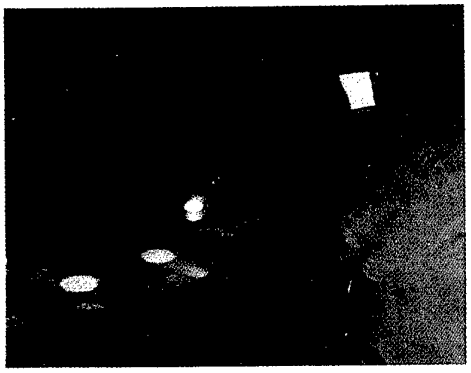
3. Implementation

We have cheated works by videotape, media installation (mixed media-architectural, sculptural), media performance, rowing about "Imaginably space" sin 1981. A prototype of this system was tested 'INVISIBLE POND 'in 1996 , 'A in a Dining Table 'in 1997[Fig2] and ' NEXT NEST' in 1998 [Fig3] at the Virtual Reality Society of Japan. 'INVISIBLE POND 'in 1996 is a future floor . There is the virtual pond, which made by reinforcement glass. The audience take on the floor made by glass, and get an actual feeling of a gap between body space with real glass and a virtual images.

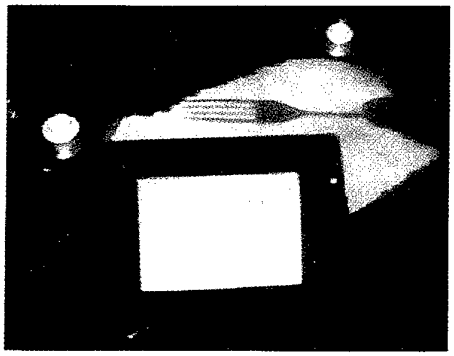
"Artificial Dining Table" is a future furniture in a human life environment and a communication tool between you and me. Virtual reality display is a new material in an industrial design and interior design and architecture . We cleat a new style furniture. That furniture can be a interactively change the surface image and communicate with a sitting people. The surface image is showing a real daily life images and disciplinary virtual images marked computer graphics. Interactively Audience can communicate a image and sound between real life and virtual life by the sensor. "Artificial Dining Table" is cleared for a proposal of a next architecture of sensory perception by an artist ,architect and computer programmer .' NEXT NEST' is a future communication tool which used Mixed Rarity system. The audience used a Hand-held sole display. They can find the hidden images that mixed real and virtual images from one TV channel .

'Life Drops ' ' Life Drops 2' is a future window.

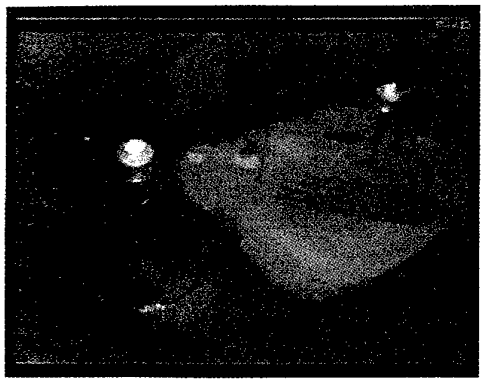
We represent the concept of a new form of communication with a environment , generates various sound and images in real time on the movement of the human body in space.



[Fig.4] ' INVISIBLE POND' in 1996



[Fig.5]'A in a Dining Table 'in 1997



[Fig.6] ' NEXT NEST' in 1998

[Fig.7] ' Life Drops' in 1999

[Fig.8] ' Life Drops 2' in 2000

4. Future Work

In the future, We try to travel with creative imaginary space using wearable MR system. Hide-and-Seek is a future interactive dining table. We hope this furniture is used in life, and expects that this work becomes a tool making life pleasant in as a part of architectural environment and communication .

Apropos Dancing Technology

— Using a CD and Computer Animation to Assist Choreography —

Yi-Jen Huang
National Taiwan College of Physical Education
Dance Department
No.16, Section 1, Shuan-Shin Road, Taichung, Taiwan, R.O.C.
Email: yijenhuang@hotmail.com

Abstract

In the beginning of this paper, we are going to discuss how to integrate dancing and interactive CDs together. Using a CD and computer animation to assist choreography is a revolutionary task. The interactive CD is divided into many small windows (excluding virtual animation windows) and each one equivalent to the traditional dancing recording instrument has its own subject; however, any recording instrument has its limitation. But once we are able to combine videos, animation, dance registers, and other styles of dancing records into a CD with an interactive virtual dancing environment, we'll better understand the dancing compositions. In addition, we can restore the dance to its original form or clarify a composition's presentation. We'll take "Motion Capture" and "Life Forms" tool application as vivid examples so as to realize how multimedia computers and interactive CDs bring
a great impact on the dancing field.

I. Introduction

In ancient times, people had to rely on dancing masters to learn dancing techniques. With the development of photography, it made an effort to take pictures of dances. Meanwhile, with the development of light cam recorders, whole dances could be recorded. After the spread of multimedia computers and interactive CDs, the whole dance could be described because the CD can: 1) tell you the origins of the dances, 2) dub in background music, 3) record entire dancing compositions, 4) show dancing steps, and 5) show the dancer's body movements even zooming in on particular areas. As long as the dance itself drawn attention, the CD could preserve its essence forever.

In the late 1940s, computerization began and till those recent fifteen years, its technology has been improved immensely. The PC is commonplace and the data processing is more accurate and quicker. Data transmission has also been improved in speed and accuracy along with superior data compression and imperishable permanent data storage. It allows various types of data to be stored on computers and the data could be retrieved as one piece from the database with independent reconstitution as a perfect copy. Nowadays, with the improvement of digital tools, the specialized and flawless service can be provided. For example, currently there has a digitized movement in Motion Capture and in the future there will have a dancing animation by the use of Computer Animation Software "Life Forms".

In spite technology fails to enhance individual intelligence and agility, it can enrich human cultures so tremendously.

II. Integrating Dancing and Interactive CDs

A. Summary

Inviting you to enter the virtual dancing environment with interactive CD:

If we would like to understand a dancing composition more completely, we need to take a look at its interactive CD.

First, the browser window has many small view windows together with the main view window which make a series of dancing performance divide into individual elements. Every small view window has its own theme such as performance commentary, composition ideas, individual muscular and skeletal analysis films, 3-D animation, dancing photos, audition interviews of the dancers, dancing steps, rhythm symbols, tempo, dancing jargon, background music, stage lighting, stage, script commentary, and etc. The audience not only can understand the whole dance but also can have an interest in exploring some certain parts of the dance such as selecting the button for dancer interviews or the rewind button or the stop button which can allow one to understand how each individual part of the dance is synchronized on stage and how the dance is produced from audition to performance.

These view windows have two methods of viewing dances: video verity and virtual animation.

In the video verity window, we can view the dancing performance, the stage lighting, and the dancer's emotional

expressions from different angles. From the virtual animation window, we can throw away the limitation of the video camera and then twirl, draw close or far or simulate the animated dancers among 3-D world. Thus, a long dancing movement can be shortened or dances can be analyzed from different points of view.

In spite the traditional recording forms of the dancing movement notation can record different kinds of essential movement symbols, the dancer could only get a superficial outline of the dance after reading those symbols. Meanwhile, the video recording only can allow the audience to see what it shots but cannot lead them to experience the entire performance. Also, the continuous changes of videos can easily cause one to lose his thinking.

Currently, Dr. Smith and Dr. Maletic of Ohio State University have applied 'Macromedia's Interactive Multimedia Software 'Director' in developing the Multimedia Dance Prototype which can provide an approach and methodology for concepts and create an outline for multimedia dance data repositories. Any dancing group and individual dancer can use it to mutually exchange and collectively preserve their compositions. In the screening aspect, it shows film, background music, script commentaries, and dance photos; in the choreography aspect, it presents dancers' auditions, dance scores, and music scores all in individual windows which can present different dancing themes and thus can redesign arrangements. This system can become a tool for the choreography, rehearsing, and performing of dances as well as serve as the instruction environment of the digital dancing; therefore, we can click a mouse button to listen and watch the choreograph dancer's description.

B. Advantages

i. Compared with traditional multimedia systems, interactive multimedia computers have increased its interactivity of multi-media equipment. The interactive multimedia computers provide a mutual communication between the user and computer so that it allows the user to press a button or input a command in the multimedia desktop in order to display and sort data in compliance with one's wish. It is not like linearity of traditional multimedia with its films and videos which is unilateral and the user can only watch but cannot interact with it. Interactive multimedia computers are overthrowing linear systems and becoming new tools.

ii. Its Zooming Functions→Make the details of the movement clear. The

tempo adjusts the function→Give the movement a new direction which not only clarifies confusing patterns but also experts in zooming in on the hidden fluctuation of the image.

iii. Interactive Virtual Dancing Class CD is an all-aspected multimedia and a fast dancing textbook which agrees users to view various data and also improves the linearity of traditional videos. The interactive CD contains asynchronous teaching modules arranged into a virtual environment different from some cor-respondence studies which have the fixed contents. Within the virtual environment, we can learn flexibly along with our interests and schedules.

iv. Also, this CD is permanent; the recorded data can remain intact without decaying! Therefore, it is superior to traditional methods of recording dance compositions.

v. The space-time barrier can be conquered, so that lots of audience can watch the choreographed per-formance. Naturally, it allows compositions with an extensive circulation and reproduction, but not for plagiarizing.

vi. Highly efficient compositions between the choreographer and the dancer. By using the computer, addressing rehearsals, dancing composition, and performing production can be fulfilled out of the dancing studio. The choreographer can quickly and efficiently communicate with the dancer without wasting considerable energy.

vii. Wherever and whenever the choreographer can catch up his sudden inspirations for trying out more possibilities. Meanwhile, he can use this utility to transform newly envisioned compositions into understandable pieces which allows us to seize the fleeting inspirations. It also stimulates the creativeness of the choreographers - they can easily edit films and revise dance sequences so that this virtual production can provide the possibilities of real bodily operations. With blending new and old dancing com-positions together can allow dancers to be super-imposed figures and thus make the compositions have limitless capabilities. It also can apply virtual 3D animated dancers who can break through the limitations that the live dancers possess. Thus, it can provide choreo-graphers freedom beyond their imagination to test various

possibilities.

viii. Save time, Simplify, Be precise: In the future, if one would like to rearrange the words of the composition, it will be much easier and more precise. Therefore, the interactive dance CD will bring the dance field unpredictable and valuable benefits.

III. Two Items

A. Motion Capture's Movement Digitalization

Motion Capture is an accurate interactive composition recording utility. Normally, the user will wear a body suit or a small suction cup adhered to the users' joints which can record the amount of movement of the individual body parts, the displacement, tempo, and rhythm changes and then convey the data to a computer. The computer will then use software resembling 3D Studio MAX animation software to convert the data into a virtual character animation.

Currently, there are two aspects to use Motion Capture: 1) in biomechanical and bioengineering research, it is applied to transform statistical data for medical science, mechanics, sports, and military installations; and 2) in entertainment as in movies, computer games and advertisements. However, because of the high cost of this equipment, it is rarely applied in dance composition. But in recent years, various performers and digital artists from different countries have cooperated in the use of this technology. Up-to-date, it has been developed so that it not only can display dancers' complete body movements by using virtual 3D computer images so that choreographers can separate or reassemble compositions, but also can record future integrated dance schemes (for example: Labanotation).

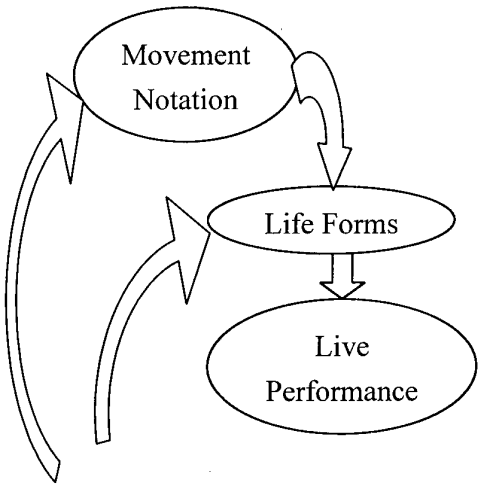
B. Life Form's Dance Animation

"Life Forms" software can allow choreographers to create, save, and edit movements. If traditional movement notations are imported into "Life Forms" → then they can be directly broadcast as animated figures. Merce Cunningham said, "Like chance choreography, computer choreography can stimulate us to think; perhaps some of the methods you have not thought of before."

Merce Cunningham has been forecasting since 1968 that choreographers would use technology as their tools and in the past 10 years he has composed dances by using "Life Forms" software.

In the future "Life Forms" software will be integrated with Motion Capture. Most probably Motion Capture will be used to record live performance broadcasts → then the data will be sent to "Life Forms" for editing. On the other hand, good dance sequences can be choreographed by using the software before going to the dance studio; therefore, dance choreographers can save time to search for dancing movements with the dancers. Choreographers can also use the Internet to provide distant dancers with individual rehearsals by transmitting "animated" dance compositions electronically. Thus "virtual dances" could become a new method of performing dances and that will be an important trend in the future. Also, there will have more independent artists involving in this field for producing and publishing their works. Therefore, the highly skilled and applied technology has become so important. So, we can combine digital technology and movement analysis together. For example, we can analyze "Life Forms" movement changes and sequences. Thus, there is a more effective application of technology in analyzing dances and in training exercises. Viewers can also communicate with the composers. The chart of this model is shown in Figure 1.

Also, it has the following major effect upon dance innovation. It makes dance instructions even clearer and accurate and provides dance composers with an independent composition method. Through using the analysis from "Life Forms" and Motion Capture, the composer can see the video images. By using virtual interaction, we can be provided with movement analysis data and new movement ideas.



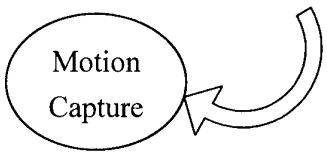


Figure 1 The chart of analysis Life Forms and Motion Capture

IV. Conclusion

Most art performers unfamiliar with this new computer technology worry very much. It is perhaps due to their unfamiliarity with these types of tools or they doubt that computers can help the dancing profession rise to new levels or perhaps because they feel that performances will be replaced by cold computer performances.

Actually, this fear is caused by the lack of understanding of how technology and dancing are integrated. This integration has great potential: 1) outstanding recording ability, 2) the persuasiveness of the interactive CD, and 3) the specialized abilities of both Motion Capture and Life Forms. Nevertheless, digital technology can never replace choreographers; it can only offer resources and be just like a servant. The soul-searching ability and perception of the composer is still the most important aspect and he uses technology only to help reveal the deep meaning of life. As for CDs can never fully replace dancers' live performances. With the era of digitalization coming, we are awakened by the way how the audience to accept digitalization; therefore, as society changes, the art world must move on by following in the direction of technology. If we attempt to draw a large of audience's attention, then we need to continue to develop this aspect of art.

We cannot deny the fact that many innovative ideas need the assistance of the new technology so as to be more effective; thus, new technology should allow art forms to develop new compositions. Now what we need to do is to allow technological transformation to be the service of art itself, and also infuse new strength into choreography.