AD-A243 705

√

(2)

FASHION INSTITUTE OF TECHNOLOGY

DTIC ELECTE DEC 12 1991 S D D

RESEARCH REPORT

AUTOMATED HANDLING OF GARMENTS FOR PRESSING

Fashion Institute of Technology

FIT AAMTD

91-17670

91 12 11 004

DLA900-87-D-0016/0006 SEPTEMBER 1991

30 Sep 1991 FINAL Nov 1989 to Sep 1991

(U) Automated Handling of Garments for Pressing

Contract
DLA900-87-D-0016/0006

Professor Aaron Schorr

Program Element No.
78011S

Educational Foundation for the Fashion Industries
Fashion Institute of Technology
227 W. 27th Street
New York, NY 10001

FIT Delivery Order 0006

Defense Logistics Agency (DLA-PRM)
Cameron Station
Alexandria, VA 22304-6100

DLA900-87-D-0016/0006

Unclassified/Unlimited

Employing all relevant existing knowledge, the state-of-the-art has been advanced in three areas: modeling of limp materials; sensing of positions and conditions of seams and garment parts; and the automated loading of garment parts on pressing machines. The feasibility and practicality of coordinating all physical activities of loading trousers onto a pressing machine with seams aligned and without wrinkles in a single, continuous, automated operation has been demonstrated.

(U) Apparel, (U) Automation, (U) Pressing, (U) Handling,
(U) Garments

347

Unclassified Unclassified Unclassified Unlimited

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to **stay within the lines to meet optical scanning requirements.**

Block 1. Agency Use Only (Leave Blank)

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	- Contract	**PR**	- Project
G	- Grant	**TA**	- Task
PE	- Program Element	**WU**	- Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Names(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency. Report Number. (If known)

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of ..., To be published in .... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availablity Statement. Denote public availability or limitation. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR)

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE - See authorities
NASA - See Handbook NHB 2200.2.
NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - DOD - Leave blank
DOE - DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports
NASA - NASA - Leave blank
NTIS - NTIS - Leave blank.

Block 13. Abstract. Include a brief (Maximum 200 words) factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code (NTIS only).

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

ADVANCED APPAREL MANUFACTURING TECHNOLOGY

FASHION INSTITUTE OF TECHNOLOGY

DLA900-87-D-0016-0006

AUTOMATED HANDLING OF GARMENTS FOR PRESSING

FINAL TECHNICAL REPORT A008

Aaron Schorr
Project Leader

SEPTEMBER 30, 1991

This project has been sponsored by the
DEFENSE LOGISTICS AGENCY
CAMERON STATION
ALEXANDRIA, VA 22304-6100

Automated Handling of Garments for Pressing

Final Report

The Educational Foundation for the Fashion Industries

and

Center for Manufacturing Productivity and Technology Transfer

September 30, 1991

This report consists of two parts: project accomplishments; and additional information in the areas of Sensing and Integration.

This research report is the result of work performed under sponsorship of the Fashion Institute of Technology and the United States Department of Defense, Defense Logistics Agency (Contract DLA900-87-D-0016). It's contents are the property of the Educational Foundation for the Fashion Industries and mav not be reproduced or used without written permission.

It is hereby submitted to the DLA office (DPMSO), Cameron Station, Alexandria, VA 22304-6100 in accordance with the Contract Data Requirements List, sequence A008.

Henry K. Seesselberg
Director, Advanced Apparel
Manufacturing Technology Programs,
Fashion Institute of Technology

Noah Brenner
Research Coordinator,
Advanced Apparel Manufacturing
Technology Programs,
Fashion Institute of Technology

September 30, 1991

Project Leader's Note

It is with conflicting emotions that I respectfully submit the Final Report for Phase I of Project 006, Automated Handling of Garments for Pressing. On one hand we, the project team, feel pride in what we were able to technologically accomplish in the first phase of this project, and on the other we feel the frustration of not being able to properly complete the task.

This is further bolstered by our knowledge of what is being done elsewhere in the world, particularly in the Pacific Rim and in the European Economic Community by organizations such as CIRCEA, the Clothing Industry Research Centres European Association, one of many groups forging ahead in the study of robotics and automated garment handling. As one of the primary aims of the AAMTD program was to aid in the further development of the American apparel industry in order to maintain its manufacturing capability onshore for military preparedness, it is difficult to understand how this type of research effort can be abandoned.

Military and civilian contractors must be price competitive as they deliver quality products or they will be forced out of the marketplace. How will they be able to effectively compete against foreign apparel manufacturers who are investing in higher levels of manufacturing and information technology that is being developed through government sponsored research centers and/or legalized cartels?

I personnally hope that we do not witness with the apparel industry the debacle that occurred in the automotive industry in recent years. We, on our side, advocate long term development and research but require immediate commercial results. Our foreign competitors have taken to heart the long term approach and have ended up with a very profitable portion of the pie. The military and civilian apparel firms are located throughout America and any additional erosion of their numbers will have significant negative impacts on many suburban and rural areas that have been attempting to stabilize local economies that were subject to seasonal climatic variations because of their historical reliance on agriculture.

The loss of this manufacturing base to overseas firms can have far-reaching economic consequences that are beyond measurement by this research team in an informal way. It also tends to compromise the DLA mission of assuring a responsive manufacturing base in case of future mobilizations such as Desert Storm.

Aaron Schorr
Project Leader

September 30, 1991

Authors

Julianne Boyd, RPI, Graduate Student ECSE
David Breen, RPI, Engineer RDRC
Noah Brenner, FIT, Professor Textile Science
Claudio Caballero, RPI, Programmer CMPTT
Tiehua Cao, RPI, Graduate Student ECSE
Dr. Steven Derby, RPI, Professor MechE
Dr. Leo Hanifin, RPI, Director CMPTT
Dr. Donald House, Williams College, Professor CompSci
Dr. Robert Kelley, RPI, Professor ECSE
Cliff Lansil, RPI, Graduate Student MechE
Laura Mark, FIT, Bachelor Student Production Management
John Peiffer, RPI, Graduate Student MechE
Lawrence Ruff, RPI, Project Manager CMPTT
Dr. Arthur Sanderson, Professor and Chairman ECSE
Aaron Schorr, FIT, Professor Apparel Production Management
Ada Warrach, FIT, Bachelor Student Production Management

TABLE OF CONTENTS

Part 2

List of Figures

List of Tables

1.0 INTRODUCTION

1.1 MISSION - One of the greatest impediments to significant improvements in apparel manufacture is the absence of reliable devices for manipulating garment parts and assemblies into and around the sewing and pressing equipment. Many attempts have been made to solve this problem with very limited and narrow success .

In January, 1990, a team of researchers began a project funded by the Defense Logistics Agency aimed at automating garment handling for the pressing operation. The rationale for this research is straightforward. While many process steps in garment making have derived considerable advantage from the automated movement of material *during* the process, and many "unit handling" systems have automated the movement of material *between* processes, little has been achieved in the automated *loading and unloading* of the various garment making process operations.

The pressing operation was chosen for reasons relating to effective manufacturing and technological challenge. With regard to garment manufacturing, pressing is a taxing task performed in an inhospitable and potentially dangerous environment. Pressing has been an area reserved for workers unable to sew. Pressing has traditionally attracted males, with higher pay used to compensate for poor working conditions. The labor pool has been shrinking in recent years, and with new curbs on immigration it is expected that this will amplify the problem. In addition, press operators, upon gaining seniority, often "bump up" onto other more attractive jobs. This constant rotation of new press operators creates both quality and rate problems, and in certain labor areas, difficulty in even securing the necessary labor. It would therefore be desirable to remove human operators from this operation. Secondly, because of the challenges of seam matching and flattening multiple plies, this has remained a labor intensive operation which demands skillful operators. It therefore would provide an important cost advantage to reduce the labor content in this operation.

These challenges to the human operator also provide significant technological challenges for automation, warranting a substantial research program. In particular, the issues of handling and modeling of "floppy" materials have never been fully solved. To perceive garment conditions (such as seam location and wrinkles), predictively model the behavior of fabric, and grasp/manipulate fabric are indeed challenging to the researcher and practitioner alike. In fact, if the problems for this task are solved, the results will be applicable to a broad array of operations in garment making and other industries, especially composites manufacturing. Finally, it is well known that foreign research organizations have begun to address these issues. (That research is being performed in the closed environment of funded industrial consortia, making the results inaccessible by American manufacturers.) If we are to compete effectively in an international marketplace, we must rise to this technical challenge.

1.2 BACKGROUND - Fabric, although usually thought of as a simple, old-fashioned manufacturing material, has some unique characteristics which make it extremely difficult to handle. While application of robotics in the hard goods industries has been highly refined, it has hardly started in garment assembly. (For additional information on robotics and robots in the apparel industry see Appendix D) The engineering community has shied away from attacking the problem as no one seems to be able to define fabric in engineering terms. "Hand", "drape", " stretchy" . . . typical fabric and garment terminology . . . are not sufficiently precise for scientific analysis and engineering solutions to the problem. Fabric is a material which has few predictable dimensions and very ill-defined characteristics.

Very little has been done until the past three or four years to improve the pressing technology. Most advances have been in the steam heat timing controls, principally in adapting microprocessor techniques for this purpose. Also, some work has been done to speed up the operation by the use of automatic carousels where one garment can be pressed while another is being loaded. Still, the main part of the labor content remains, i.e., the loading of the garment on a buck and the shaping and smoothing.

If apparel manufacturing systems technology is to be improved from start to finish (and great strides have already been made in the design, cutting, and sewing), the pressing technology must be moved forward to true automation. This can only be done by scientific examination of the impediments to this automation and the development or technology to overcome those impediments. This project is establishing a knowledge base from which practical, truly automated pressing systems can be designed and manufactured.

The Fashion Institute of Technology (F.I.T.), acting as the prime contractor, and Rensselaer Polytechnic Institute, its research subcontractor, are well-suited to perform this project. F.I.T.'s broad knowledge and experience in designing and manufacturing requirements for all clothing types provides the qualitative and quantitative information necessary to specify, plan, execute and evaluate the various tasks to be accomplished. Its Advanced Apparel Manufacturing Technology Demonstration (AAMTD) will be a valuable forum for demonstrating results. F.I.T. provides not only the knowledge of the garments and fabrics, but also the bridge between the high technology world and that of the fashion industry. This is due, in part, to the extensive mix of staff in such disciplines as textiles, garments, factory, and equipment engineering.

The Center for Manufacturing Productivity and Technology Transfer (CMP) at the Rensselaer Polytechnic Institute in Troy, NY is the leader in the field of advanced manufacturing technology required to develop mechanisms, software and controls to demonstrate robotic pressing possibilities. Rensselaer has an Intelligent Control Laboratory, a Mechanical Robot Laboratory, an Automated Assembly Laboratory and a Design Research Center, all of which will be particularly appropriate for the modeling, systems design and simulation, and mechanism design, and controls/sensing systems development needed to accomplish this project. Further, Rensselaer has been designated as a Center for Advanced Technology by the State of New York Science and Technology Foundation and has 90 faculty and students involved in new research projects dealing in automation and robotics. In addition, the CMP has recently established, as a result of the designation from the Department of Commerce (National Institute of Standards and Technology), the Northeast Manufacturing Technology Center (NEMTC). New York State has a network of Industrial Innovation Extension Agents through whom NEMTC is disseminating manufacturing technology to small- and medium-sized manufacturing companies. This center has, as its goal, the transformation of manufacturing research results into hardware/software systems specifically suited for small and medium sized companies. . . their initiation in such companies. . . and their commercialization.

1.3 OBJECTIVE - The overall objective of this project is to acquire (by creation or discovery) the knowledge necessary for a totally automated pressing system which loads a complete garment or sub-assembly on a form, smooths it for pressing and unloads the press. This will be achieved in three year-long phases:

- Year 1. Development of enabling knowledge and subsystems for modeling, sensing, grasping and manipulation, and control, with integration architecture.
- Year 2. Creation of a working prototype which integrates the subsystems, and proof-of-concept at the alpha (research) site.
- Year 3. Commercialization through beta site implementation in partnership with equipment vendors.

1.4 FACILITIES, EQUIPMENT AND MATERIALS - Rensselaer Polytechnic Institute's Center for Manufacturing Productivity and Technology Transfer is housed in the $65M facility called the "Center for Industrial Innovation". This facility also houses the robotics laboratories of the Electrical, Computer,and Systems Engineering and Mechanical Engineering Departments, the Rensselaer Design Research Center, the New York State Center for Advanced Technology in Automation and Robotics, and the Northeast Manufacturing Technology Center. Together, these organizations have seven laboratories which focus on robotics and robotic sensing, control, mechanisms and modeling. These laboratories have a broad array of robotic equipment, end-effectors, sensory systems and control for computers necessary for the completion of this effort.

Some of the development of the vision sensing and the associated algorithms have been performed in the Machine Vision and Sensing Laboratory and the Intelligent Robotics Laboratory. These laboratories are interconnected via a local ethernet network. This integrated research facility consists of eight Sun workstations, several VME-bus 68020 micro-processors, a real-time Datacube vision processor and an AIS-3500 vision processor with color vision capability. The microprocessors are dedicated to robot control and sensor interfacing functions. There is a MATROX-based vision system which resides in its own Sun workstation with a TAAC board and hard disk drive. All software has been written in a high-level language to ensure portability.

F.I.T. has supplied all fabric samples, garment parts, and finished garments needed for the project.

1.5 PERSONNEL - The research team consists of personnel from the Fashion Institute of Technology and Rensselaer Polytechnic Institute (RPI). The principal investigators of the contractor have broad knowledge and accomplishments in apparel and textile research and development. They are as follows:

> **Aaron Schorr** - professor of apparel production management, Fashion Institute of Technology, has had a varied career in the apparel field since 1973 as an industrial engineer, consultant, and product and marketing manager for a pressing equipment supplier. He has assisted in developing sewing and cutting room training and procedures manuals, and has been involved with garment costing, equipment layouts, and work methods analysis. He holds an A.A.S. degree from Fashion Institute of Technology and B.B.A. and M.B.A. degrees from University of Arkansas.
>
> **Noah Brenner** - professor of textile science at the Fashion Institute of Technology. He is a graduate of Philadelphia Textile Institute and the University of Chicago. He has spent more than 35 years in the textile and apparel industries with primary experience at the interface between the two industries in both government and private sectors. He has been involved with apparel plant management and textile fiber product management, market planning and development, quality control, and technical marketing. In the fabric area his management experience has included responsibility for new product and market development, and strategic planning for apparel, industrial and home furnishings fabrics. Since 1985 he has been active as an independent consultant to the textile and apparel industries.

These investigators have been complemented by four student assistants with pressing training to assist in demonstrations and knowledge transfer of existing pressing technology.

The subcontractor (RPI) research group is comprised of thirteen technical contributors. The subcontractor's principal investigators includes four faculty members and three research staff members. These individuals are uniquely qualified to address the inter-disciplinary research challenges presented by this problem. They are as follows:

Dr. Leo E. Hanifin - Dr. Hanifin is Director of Rensselaer's Center for Manufacturing Productivity and Technology Transfer. He holds a bachelor's degree in mechanical engineering, a master's degree in solid mechanics, and a doctor of engineering degree (thesis on modeling of manufacturing systems), all from the University of Detroit. Before coming to Rensselaer, he worked in the automotive, aerospace, and computer industries. His responsibilities included development of major ($80M) manufacturing systems and management on the production floor.

Prof. Arthur Sanderson - Dr. Sanderson is Department Head of the Department of Electrical, Computer and Systems Engineering. He holds a bachelor's degree from Brown University and master's and Ph.D. degrees from Carnegie-Mellon University. His current research interests include planning systems for robots and automations systems, sensor-based control, computer vision, and applications of knowledge-based systems. He is President of IEEE's Council on Robotics and Automation. He was a founder and Associate Director of the Robotics Institute at Carnegie-Mellon University.

Prof. Robert Kelley - Dr. Kelley's research interests include robotics and computer vision. He is a professor in the Electrical, Computer, and Systems Engineering Department at Rensselaer. He earned a bachelor's degree from Newark College of Engineering, a master's degree from the University of Southern California, and a Ph.D. from the University of California at Los Angeles. He was previously Director of the Robotic's Research Center at the University of Rhode Island.

Prof. Stephen Derby - Dr. Derby holds bachelor's, master's, and Ph.D. degrees in mechanical engineering from Rensselaer Polytechnic Institute. He has conducted research in the areas of robotics, computer graphics simulation, robotic autoloaders, and robotic die finishing. He created the GRASP robot simulation program, and has taught courses in robotics, mechanisms, kinematic synthesis, and computer graphics.

Lawrence Ruff, Project Manager, CMPTT - Mr. Ruff holds bachelor's and master's degrees in mechanical engineering from Rensselaer Polytechnic Institute. He has worked in the areas of machine tool design, robot system integration, and assembly and packaging equipment design. He has many years of manufacturing experience in all areas of metal cutting.

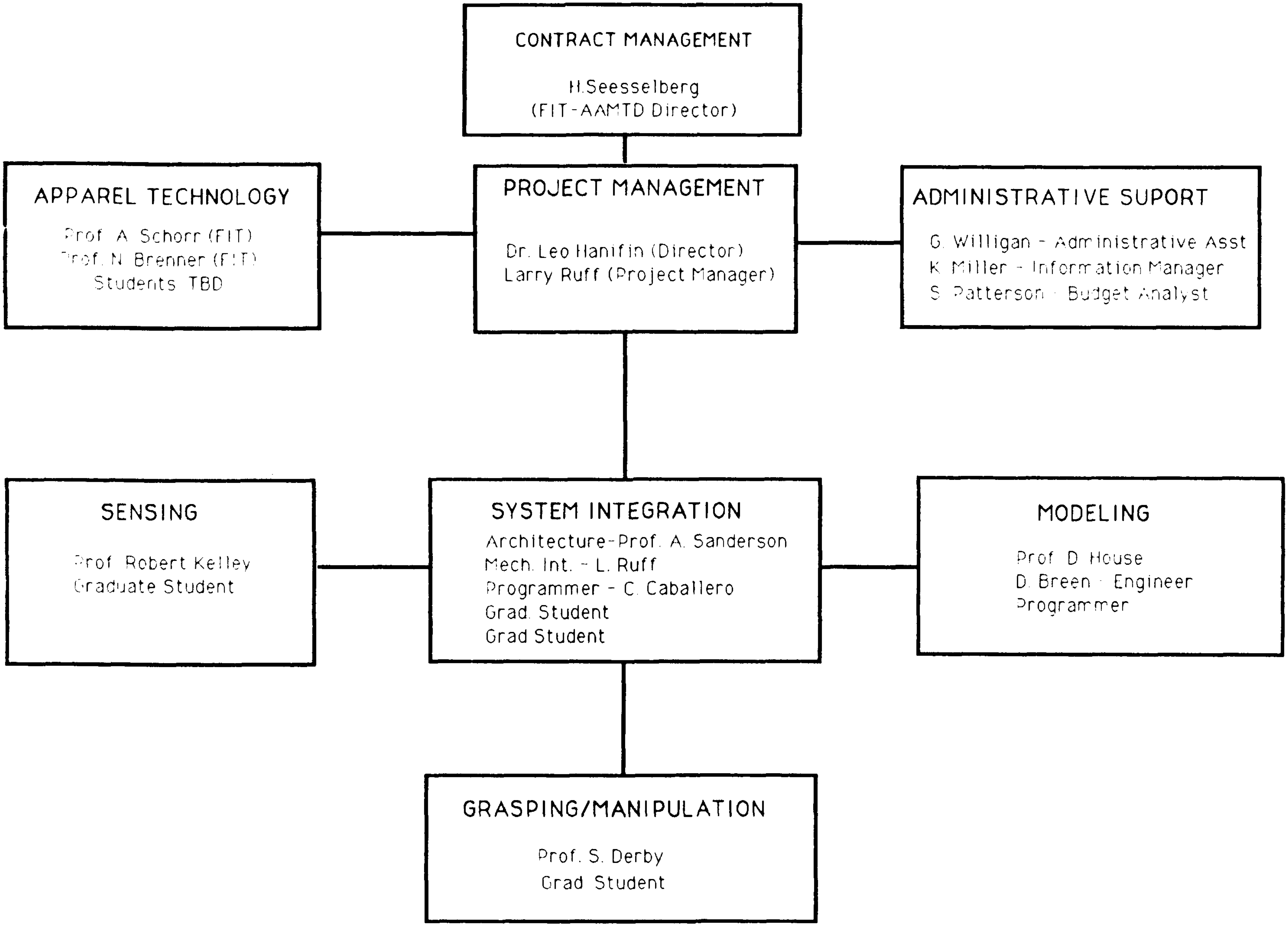
David Breen, Research Engineer, RDRC - David E. Breen is a research engineer at the Rensselaer Design Research Center (formerly the Center for Interactive Computer Graphics). He has been on the full-time staff of the RDRC since1985. From August 1987 to July 1988 he was a visiting research engineer at Zentrum fuer Graphische Datenverarbeitung in Darmstadt, Germany. His research interests include particle-based modeling, dynamic simulation, computer animation, geometric modeling and object-oriented programming. He is a member of ACM SIGGRAPH, and the IEEE Computer Society. Breen received his BA in Physics from Colgate University in 1982. He received his MS in Electrical, Computer and Systems Engineering (ECSE) from Rensselaer Polytechnic Institute in 1985 and is currently pursuing his PhD in ECSE at RPI.

Prof. Donald H. House, Visiting Faculty Member - Dr. House is an Assistant Professor of Computer Science at Williams College. He also maintains an office at

Rensselaer's Design Research Center where he conducts research in simulation and computer animation. He holds a master's degree in electrical engineering from Rensselaer and a Ph.D. in computer and information science from the University of Massachusetts.

These researchers are complemented by four graduate students, each working between fifty and sixty-six percent of their time on this research effort and two programmers. The program's personnel are organized according to the following organizational chart, figure 1.

AUTOMATED HANDLING OF GARMENTS FOR PRESSING RESEARCH TEAM



2.0 FIRST YEAR ACCOMPLISHMENTS

The first year of this project has focused on defining, developing, and demonstrating the technologies required to automatically perform the finish pressing operation on trouser legs. The technologies used to implement the required functions have followed the descriptions in the year 1 proposal. Brainstorming sessions were used to refine the ideas presented in the year 1 proposal and to also incorporate useful technologies learned during a series of university and industry site visits. These sessions defined the primary technologies and devices to be investigated along with secondary ideas to be investigated in parallel. A System Functional Requirements Statement has evolved over the first year of the project and is used to define the system operating parameters and specifications. The functions of seam alignment, trouser transfer and press loading, wrinkle detection and correction, and press unloading are being performed sequentially. The workcell is shown in figure 2.

2.0.1 Coordination with Other Ongoing AAMTD Projects

During the course of the first phase of this project, researchers from F.I.T. and R.P.I. contacted various organizations and other DLA sponsored research facilities to:

1. Develop a knowledge of current research.
2. Establish contacts in relevant areas for future work.
3. Keep other projects informed of the progress being made during this first phase.

We participated in AAMTD group meetings with other researchers and DLA personnel.

2.0.2 Evaluation of Current Knowledge

To avoid duplication of effort the researchers spent the first part of the project meeting with a variety of experts in the industry to establish the state-of-the-art in this technology. This included visits to:

1. Joseph J. Pietrafesa Co. to observe production of trousers and pressing in real time.
2. F.I.T. AAMTD demonstration site by RPI researchers to look at material transfer and feeding mechanisms on advanced apparel equipment and to learn how to press a pair of trousers.
3. TC2 in Raleigh, N.C. for robotics and observation of material transfer and feeding mechanisms.
4. Computer Design Inc. for garment modeling.
5. Draper Labs for garment handling.
6. Jet Sew for the Clupicker and material feeding.
7. DPSC manufacturing facility in Philadelphia to observe military trouser production.
8. The Bobbin Show to meet with equipment vendors.
9. JIAM '90 to observe the latest in technology.
10. Clemson Apparel Research to compare notes on technology and material handling.
11. R.P.I. research facilities by F.I.T. researchers for working sessions, and arranged a demonstration for all of the researchers by Mr. Michael Grogan, pressing consultant, on the proper methods of pressing trousers and quality.

2.0.3 Work With Industry Review Board

It was initially felt that we would establish an Industry Review Board (IRB), but during the course of the first phase it was determined that industrial partners would be more valuable to the success of the project. There are patent questions that make an IRB potentially unwieldy. As a result we have established partnership relationships with Hoffman/New Yorker Pressing Machinery Company, and General Motors Fanuc. This revision was approved by DLA.

Both companies occupy a position of leadership in their respective fields with GMF the largest manufacturer of robots in the country, and Hoffman/New Yorker the largest manufacturer of pressing equipment in the country. As such, they offer effective resources for equipment and software, guidance in their fields, and commercializaiton of results. GMF is providing to the effort an S-700 Robot with the advanced KAREL control system and vision system. Hoffman/New Yorker is providing both equipment and technical manufacturing support. The equipment is primarily a Hoffman Universal Press Model #UAL135 pressing system. In addition, Hoffman has agreed to provide design and manufacturing support services to modify and/or extend their system.

It is anticipated that the industrial partnerships will continue through the next two phases of the project and may include other firms. Inclusion of other firms will depend on additional technical expertise that they might be able to offer the project to advance the technology in a positive direction.

2.0.4 Location of Work Cell

During the first phase of this project the work cell has been established at R.P.I. Phase 1 of this project involves the initial development of the various possible components of the system. The work cell therefore needed to be located at R.P.I..

2.0.5 Information Transfer

The R.P.I. team met weekly during Phase I. There was irregular participation at these meetings by F.I.T. personnel. There were additional meetings at F.I.T., R.P.I., and other locations previously discussed as needed. Formal quarterly reviews were held and presentations were made at F.I.T. monthly project meetings, and the Annual Contract Briefing. One of the areas of original research we have identified-garment modeling-may have other possible uses at F.I.T.. During the early part of Phase II we will be attempting to run the software developed at R.P.I. on computers at F.I.T.. If compatible then we plan to hold a seminar for interested instructors and students during Phase II of this project to investigate other possible applications for this modeling technique. It has been determined that due to the sensitivity of the material currently under investigation, that public dissemination of the results will be withheld at the present time. Information will be considered classified at this time and will be distributed according to the guidelines established under the contract DLA900-87-D-0016.

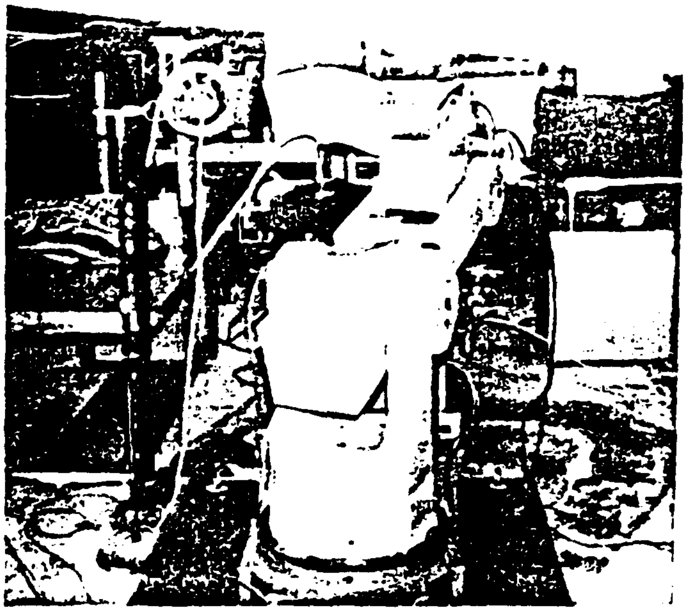
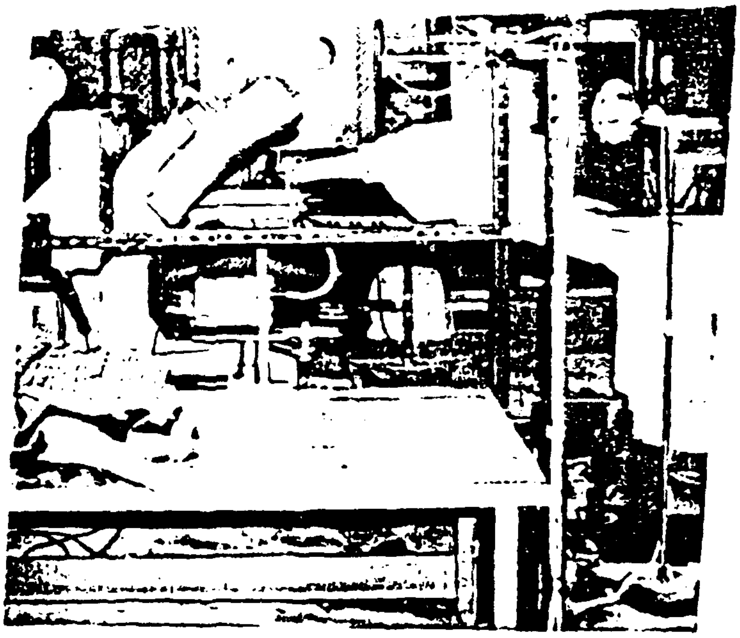
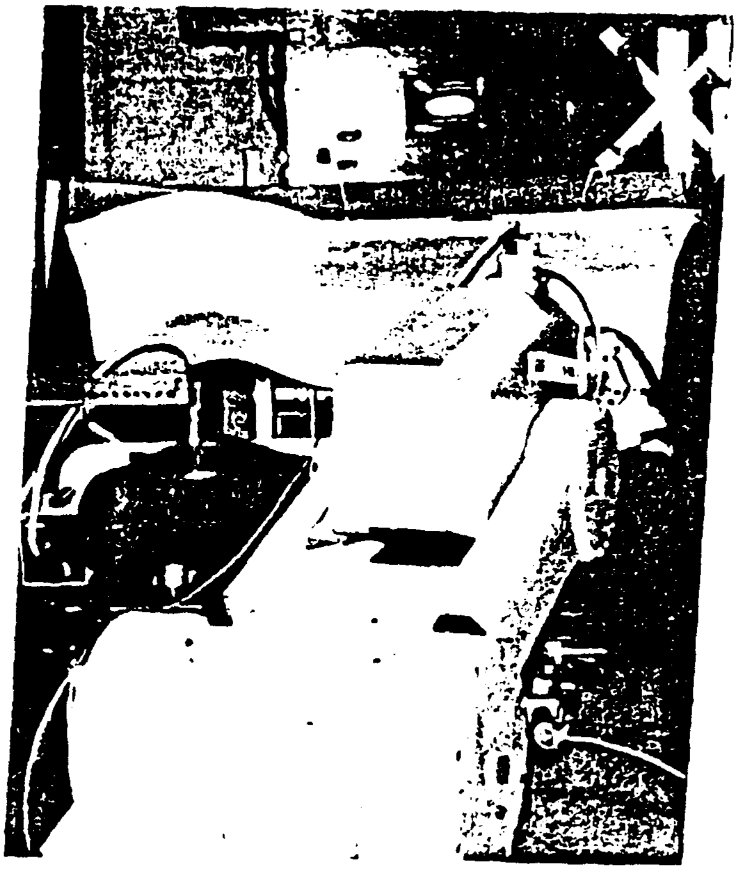


Figure 2 Pressing Workcell





2.1 MODELING

2.1.1 Introduction

The modeling group has the responsibility for developing a working model of the mechanical behavior of cloth. When this model is fully developed it will be able to be used in designing manipulators for handling cloth, and could ultimately be used in conjunction with sensing data as part of a real-time control system for the planning of such complex procedures as smoothing of cloth.

During the first year of the project, work in modeling took a two-pronged approach, simultaneously laying the theoretical foundations, and developing tools and working models. On the theoretical side we spent some time developing a level of expertise on the nature of woven fabric and understanding previous models of cloth. Some of the special problems with cloth modeling were recognized right from the start, so we devoted a good deal of effort to laying theoretical foundations and developing working methods for a new modeling methodology that we call "particle-based modeling". Our tool-building effort focused on extending Rensselaer's computer visualization environment, to make it into a suitable computer-based testbed for experimentation with and visualization of the evolving cloth model. As tool building and theoretical investigation progressed, we began constructing a preliminary model and testing it. This work consumed much of the middle part of the year, and provided experience that we applied to further refining the tools, theory, and ultimately the model.

A final goal for this year's modeling work was the production of a computer animated film showing a cloth being draped over a surface. The purpose of the film was to push all of our technology to its limits, providing at once a demonstration of the capabilities of the computer testbed environment for visualization and experimentation, and a realistic test case for evaluation of the model. *In fact we found computer animation to be extremely important to the* progress of our work. It provided excellent feedback on the quality of the model, helping us to uncover several important shortcomings which we have been able to correct. Thus, instead of producing one animation, we have produced several of varying degrees of quality along with a series of still pictures that surpass the animated film in level of detail and image quality.

The following section summarizes the results of this years modeling efforts. It provides a fairly complete description of the model as it stands to date, and projects work that still needs to be done to perfect it. This report does not precisely follow the effort as outlined in the project Gantt Chart, but the following cross-reference should make the connections fairly clear. Introductory material including the overview of the model and the survey of previous models summarizes the work on *Theoretical Foundations* and *Fabric Properties Studies*, although the description of the model also includes much that was discovered during these phases of the work. The section on evaluation of the model also describes methods that were developed as part of the *Theoretical Foundations* work. The description of the model focuses mainly on the particle-based nature of the model and describes in detail the *Fabric Particle Interactions* that were incorporated into the model. Finally, the section on visualization describes the use of the *Computer-based Testbed*.

2.1.2 Overview of the Model

In order for modeling and visualization to become standard engineering design tools it must become possible to treat the full range of materials commonly encountered in construction, manufacturing, and fabrication. The materials that have been successfully handled so far exhibit a mechanical behavior that, over a wide range of stresses and strains, is adequately characterized by a nearly-linear continuum representation. Such "engineering materials" as steel, glass, and many plastics fit this criterion. Their behavior can be simulated very successfully using finite-element or finite-difference methods, and a variety of readily available visualization tools can be used to view results of these simulations. Unfortunately, many other commonly used materials do not admit to this form of analysis, and thus lie outside of the range of current design automation technology. Such materials include plastic foams, certain composites, and process materials that undergo phase changes. One of the most commonly used materials that has evaded precise engineering analysis is woven fabric. Perhaps it is because of the lack of a good engineering model of cloth that most of the design work for materials-handling in the garment manufacturing industry has been largely a "seat-of-the-trousers" art.

Over the past year we have been developing a model of woven fabric that ultimately is to be capable of predicting the dynamic draping behavior of woven cloth. Our project goal can be summarized by a question. Given a piece of fabric and a geometric environment with which it interacts, how will the fabric fall on the geometry, and what will its final equilibrium configuration be? Besides this immediate pragmatic goal, we were also intrigued by the potential for developing a physically-based model of cloth useful for computer graphics and animation. This technology would have possible application in apparel design, where currently available Computer Aided Design (CAD) tools allow the designer to work only in two-dimensions.

Our model of cloth is based on the fundamental idea that the macroscopic behavior of complex materials arises out of the interactions between its microscopic structures. In some materials it is sufficient to statistically aggregate these microscopic interactions and then to treat the aggregate as a continuum that can be described by partial differential equations. In the case of cloth, however, the microstructure consists of threads and yarns interlaced in a particular weave pattern. Much of cloth's unique character comes from this underlying structure, with its various highly nonlinear geometric constraints, frictional interactions, and anisotropy. Therefore, modeling the basic structure of threads and their interactions is essential to simulating cloth's true macroscopic behavior. Modeling the full detail of the underlying micro-structure of woven fabric is clearly beyond current computational capacities, but there is an intermediate position that can capture the most important interactions.

A piece of woven cloth consists of two sets of parallel threads, called the warp and the weft, running perpendicularly to each other. If as the weft threads travel through the weave, they alternately cross over, then under the warp threads as diagrammed in figure 3a, they form a

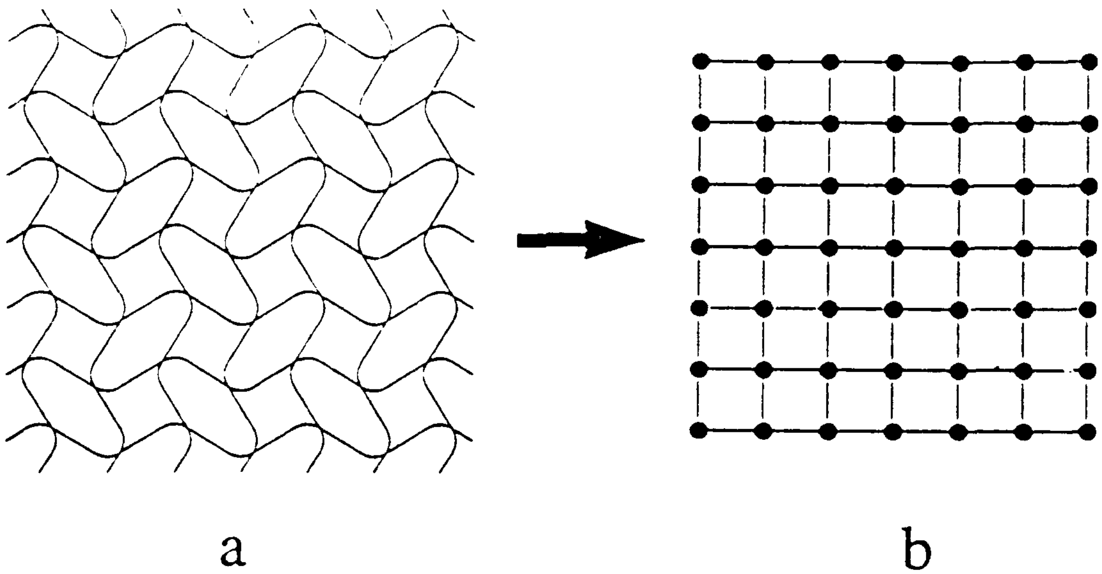


Figure 3 Mapping a Plain Weave to a Particle Grid

plain weave. Many of the important interactions that determine the behavior of plain woven fabric occur at the point where the warp and weft threads cross. For example, usually the tension is so great at the thread crossing points that the threads are effectively pinned together providing an axis around which bending can occur in the plane of the cloth. The pinned thread crossings also hold the threads in place when they are pulled along the direction of the thread.

Given that thread crossings play such an important role in influencing the local behavior of cloth, our model treats the thread crossings as the fundamental modeling unit. We call each such unit a "particle", and it is at the level of these particles that we maintain constraints, in the form of potential energy functions, about the relationships between the threads. Figure 3b shows the topologically two-dimensional particle network we use to represent a plain weave in our model. The thread-relationship constraints maintained in the particle grid account for four basic mechanical interactions occurring in cloth at the thread level. They are thread collision, thread stretching, thread bending, and thread trellising, with trellising being in-plane bending of a thread around a crossing point. If these simple interactions are properly accounted for in the grid, local interactions acting on the microscopic level aggregate to produce a macroscopic behavior that is convincingly close to cloth.

Simulations of draping cloth from our particle grid are obtained using a two step process. At each discrete time step in a simulation, we first remove the inter-particle constraints and the particles are allowed to free-fall. Collisions with and inelastic reflections from solid models are calculated at this point. After this, the inter-particle constraints and relationships are reenforced by applying an energy minimization process to the inter-particle energy functions. This pulls the particle grid together and produces the complex buckling and folding that is characteristic of draped fabric.

2.1.3 Previous Work

The initial pioneering work in fabric modeling was conducted by Peirce over 50 years ago [26]. He developed and analyzed a basic modeling cell of fabric geometry. The modeling cell detailed the geometric relationships between threads at a thread crossing. The model consists of two thread cross-sections constrained by a third thread segment running perpendicularly to the cross-sections. Using simplifying assumptions, Peirce derived a set of equations that define the relationships between the geometric parameters of the modeling cell. Over the years the Peirce model has been extended and enhanced [25][20]. Graphical methods for evaluating the equations defining the parameters have also been developed [14].

Another significant area of research in fabric modeling has been to apply the theory of elasticity, continuum mechanics and finite element techniques to modeling the mechanical properties of cloth [4][5][19][21][23][27][28]. This research has modeled threads as elastic rods and fabrics as continuous plates and shells. These approaches have produced limited successes. They have been used to predict in-plane deformation of fabric and the associated stress-strain relationships. Recent work has begun to look at buckling out of the plane. This approach unfortunately has severe limitations. Firstly, applying conventional analysis techniques to in-plane deformations is of little practical value since cloth quickly buckles out of the plane when stressed. Methods for keeping cloth in a plane during measurement and analysis simply produce mechanical data for cloth in an unnatural state with questionable value. Secondly, by assuming that fabrics are continuous sheets of material, these methods ignore the very basic fact that fabric is a complex mechanism. They are limited in that they usually assume and are only applicable for small deformations and linear behavior and have difficulty in capturing the inherent anisotropy of woven fabrics.

Another approach to modeling the behavior of cloth involves describing and minimizing the strain energy defined for basic modeling cells [1][7][8][9][15][22][29]. In this approach energy functions are based on the structure and deformation of a single modeling cell. One or more partial differential equations are derived and must be minimized in an iterative fashion. The approach is promising, but in most cases energy methods have been used only to calculate a few of the conventional mechanical parameters of woven cloth and some of the geometric parameters of just the modeling cell.

A survey of the textile research literature shows that most of the fabric modeling work of the textile community, with a few exceptions [1][22], have focused on relating the behavior of materials to traditional mechanical parameters, such as Young's modulus and Poisson's ratio. The work predominantly calculates stress-strain curves, load-extension relationships, the relationship between geometric parameters, and the dependence of bending moment on curvature. Very little of the work has actually used this mechanical information to predict the overall shape of a piece of fabric due to draping and buckling.

Surprisingly, it is mainly in the the computer graphics community where the problem of simulating the complex shapes and deformations of fabric in three dimensions has been tackled. Weil defined a geometrical approach that approximates the folds in a constrained piece of square cloth [31]. Terzopoulos and Fleischer, and Aono have developed constitutive equations for cloth and have applied finite element methods to create 3-D cloth-like structures that bend, fold, wrinkle, interact with solid geometry, and even tear [30][2]. Haumann and Parent present a method similar to ours in their Behavioral Test-Bed [13]. They define simple behavioral actors and hook them together to produce complex flexible objects. Even though their actors may be tied together in order to simulate a flag waving, the primitive actors they propose are not sufficient for the modeling of complex drape and buckling in fabric. Their hinged triangular meshes do not accurately represent the true structure of cloth. A recently published paper by Bassett and Postle in the textile literature proposes a method similar to Haumann and Parent's that models fabric as a collection of simple geometric elements [3]. This work, once again, does not address the issue of overall fabric shape, but rather the stress-strain relationships at thread crossings for a fabric stretched over a sphere. Feynman describes a technique for modeling the appearance of cloth [10]. His computational framework is the same as ours, minimizing energy functions defined over a grid of points. His and our approaches differ in the basic assumptions used to derive the energy functions. Feynman derives his functions from the theory of elasticity and the assumption that cloth is a flexible shell, whereas the interactions in our model are derived from our view of cloth as a complex mechanism, rather than a continuous material.

2.1.4 Description of the Model

Our model represents a piece of woven cloth as a topologically 2-D network of 3-D points arranged in a rectangular grid, as shown in figure 3, where each point represents a thread crossing in a plain weave fabric. We map all of the interactions occurring between threads into a set of energy functions whose independent variables are the geometric interrelationships between points in the 2-D grid. Each point is effectively connected to and influenced by a small set of neighboring points through the definition of four energy functions. These four energy functions embody local mechanical relationships and geometric constraints that exist at the thread crossings. The four thread-level features of woven fabrics that we are attempting to capture in these functions are non-interpenetration, thread stretching, thread bending, and thread trellising. Each is represented by a term in the energy equation,

$$U_{total} = U_{repel} + U_{stretch} + U_{bend} + U_{trellis}. \tag{1}$$

U_{total} is the total energy of a particle. U_{repel} is the energy due to repulsion. Every particle effectively repels every other particle, preventing interpenetration. U_{bend} is the energy due to threads bending over one another out of the plane. $U_{trellis}$ is the energy due to bending around a pinned thread crossing in the plane of the cloth.

Our approach to modeling cloth is an application of the ideas presented by Witkin et al. [32]. They have described a technique for enforcing geometric relationships on parameterized models. They define geometric constraints in terms of an "energy" function. The variables of the function are the geometric parameters of the objects being constrained. The function is formulated in such a way that the constraints are met when the ensemble of the parameter values minimizes the function. Finding the parameter values which minimize the "energy" function imposes the geometric constraint. In our approach, we are attempting to capture the geometric and mechanical relationships occurring at thread crossings. The process of developing our energy functions begins with first analyzing the relationships that we wished to enforce. This determines the general shape of the energy function. Next a function that matches the general desired shape must be derived.

In the case of the repelling potential, U_{repel}, we needed a function that would not allow one particle to pass through another. An energy function that rises from zero at a prescribed distance and goes to infinity at zero would meet these needs. The positive portion of the Lennard-Jones potential, which is used in molecular modeling, is one such energy function. The complete function is given in equation 2. The function U_{repel} effectively provides collision avoidance between particles and helps to prevent self-intersection of the fabric grid as a whole. In order for connected particles to maintain a standard distance between each other, they must sit in a steep energy well. The well was created by simply evaluating the Lennard-Jones potential, reflecting these values about the zero point, and fitting a polynomial through them using regression techniques. Equation 4 is the resulting polynomial. Figure 4b presents the general shape of the combined functions. Both U_{repel} and $U_{stretch}$ were defined to produce an energy curve of this shape. The well is produced by directly connecting each particle with the stretching potential to four neighboring particles, except along the edges of the grid. The combined repelling and stretching functions enforce the distance constraint between neighboring particles in the grid. The combined functions are

$$U_i = 4\varepsilon[(\sigma/r_i)^{12} - (\sigma/r_i)^6 + 1/4], \ r_i \le 2^{1/6}\,\sigma \tag{2}$$

$$U_{repel} = \sum_{i=1}^{n} U_i \tag{3}$$

$$U_j = 34464\varepsilon\,[(r_j - 2^{1/6}\,\sigma)\,/\,\sigma\,]^5, \ r_j > 2^{1/6}\,\sigma \tag{4}$$

$$U_{stretch} = \sum_{i=1}^{4} U_j, \tag{5}$$

where r is the distance between two particles as seen in figure 4a. The equations that we present here are our first pass at representing the true energies acting at the thread level. They most certainly will evolve as we continue our studies, refine our model, and make them more accurate and realistic. At this point it is their general shape and the role they play in maintaining geometric constraints that is most important.

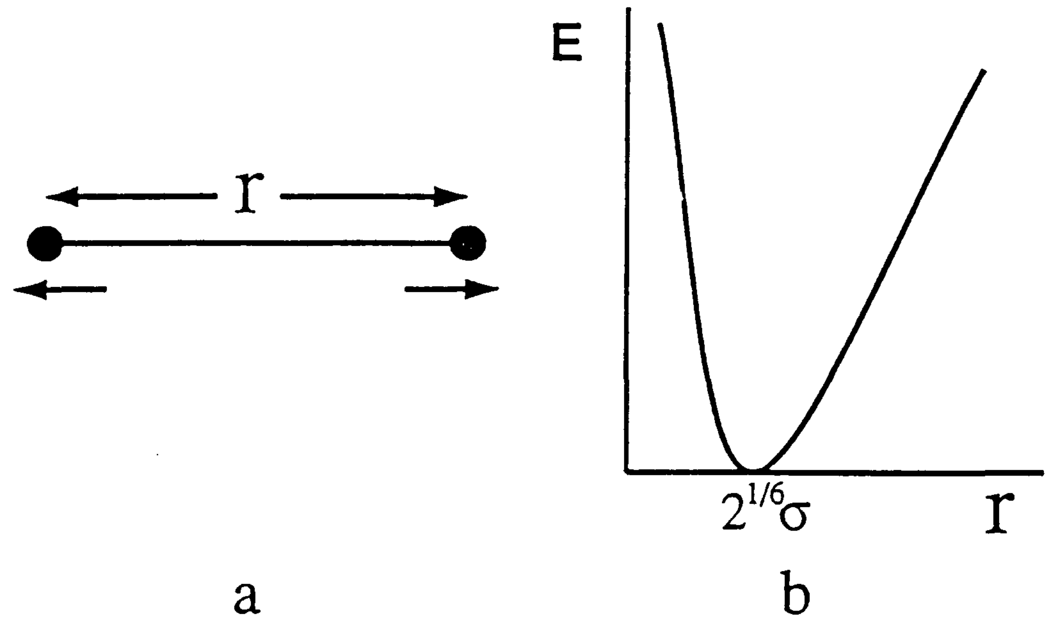


Figure 4 Combined Repelling and Stretching Function

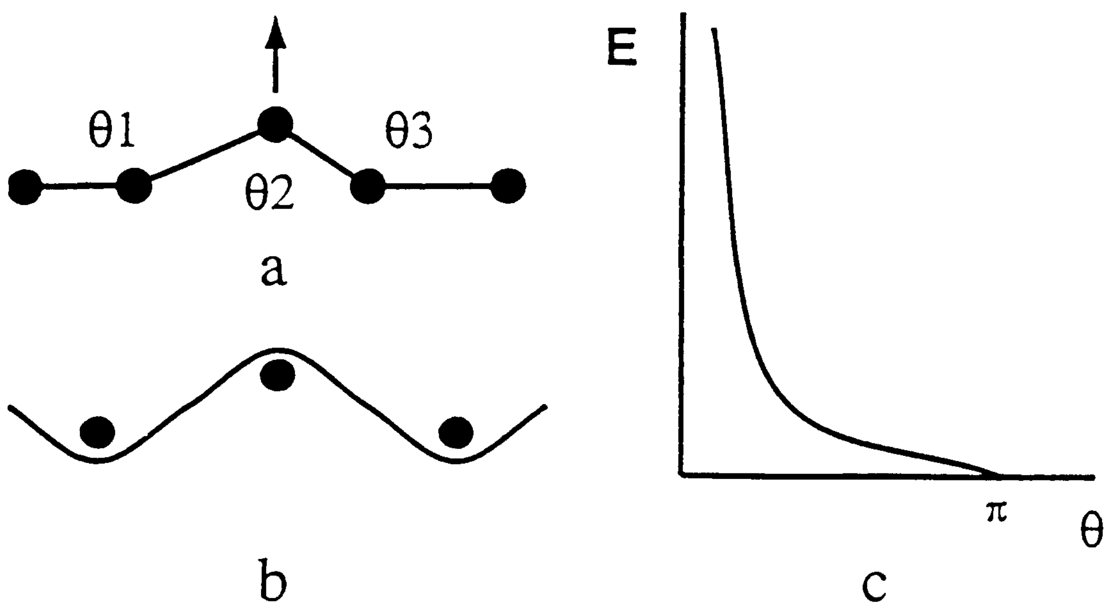


Figure 5 The Bending Potential

As seen in figure 5b a single thread will bend around crossing threads out of the plane. We have represented this phenomenon by modeling the angle formed between each set of three particles. Figure 5a shows that moving one particle, which represents a thread crossing, changes three bending angles along a "thread" line. The energy associated with this angle should be zero when the angle is π and should go to infinity as the angle goes to 0. This defines the equilibrium state of the thread as flat, and does not allow the thread to bend into itself. $Tan(\pi - \theta)$ has the general shape necessary to enforce this constraint and is shown in figure 5c. The function that represents the energy of thread bending is based on the positions of eight of a particle's neighbors, two in each of the horizontal and vertical directions. These eight particles define six bending angles. We define the energy of bending as a function of the angle formed by three particles along a horizontal or vertical "thread line". The complete bending energy function for each of these angles is

$$U_k = C_0 \tan[(\pi - \theta_k) / 2] \tag{6}$$

$$U_{bend} = \sum_{k=1}^{6} U_k. \tag{7}$$

Figure 6b presents the phenomenon that we call trellising. It occurs when threads are held fast at their crossings and bend to create an "S-curve" in the local plane of the cloth. The overall behavior produced by this kind of deformation is related to shearing in a continuous sheet of material. Since our model treats cloth as an interwoven grid of threads rather than a continuous sheet, trellising is a more accurate descriptive term than shearing. We have mapped this phenomenon into the trellising modeling cell defined in figure 6a. In the cell, two line segments are formed by connecting the four neighboring particles surrounding the central particle. The two segments cross to define an angle. An equilibrium angle for the cell is predefined. Currently, we assume that the equilibrium of the cell is at $\pi/2$. This may change over the course of a simulation as we model friction related slippage. The trellis angle is then defined as the angle formed as one of the line segments moves away from the equilibrium configuration, as shown in figure 6a. In order to enforce the trellising constraint, i.e. keeping the crossed segments at $\pi/2$, a function is needed which is zero at the equilibrium angle $\pi/2$, but that goes to infinity as the angle between the crossed segments approaches zero. This prevents the two crossed threads from crossing over each other. The general shape of the curve is given in figure 6c. The complete function for our energy of trellising is

$$U_l = C_l \tan(\theta_l) \tag{8}$$

$$U_{trellis} = \sum_{k=1}^{4} U_l. \tag{9}$$

Again, the tangent function does not necessarily accurately represent the true strain energy due to trellising. It does, however, give the correct characteristic shape.

Figure 7 summarizes the dependencies that exist between a particle and its neighboring particles. As a particle moves, it changes the energy of the system as a whole. The energy change is based on that particle's position relative to its neighbors' positions. Each component of the total energy function depends on a different set of neighboring particles. These sets are presented in figure 7. The particle being moved is represented by the large black dot. It repels every other particle in the grid, here represented by the small black dots. The stretching potential is defined between the current particle and its four immediate neighbors, signified by

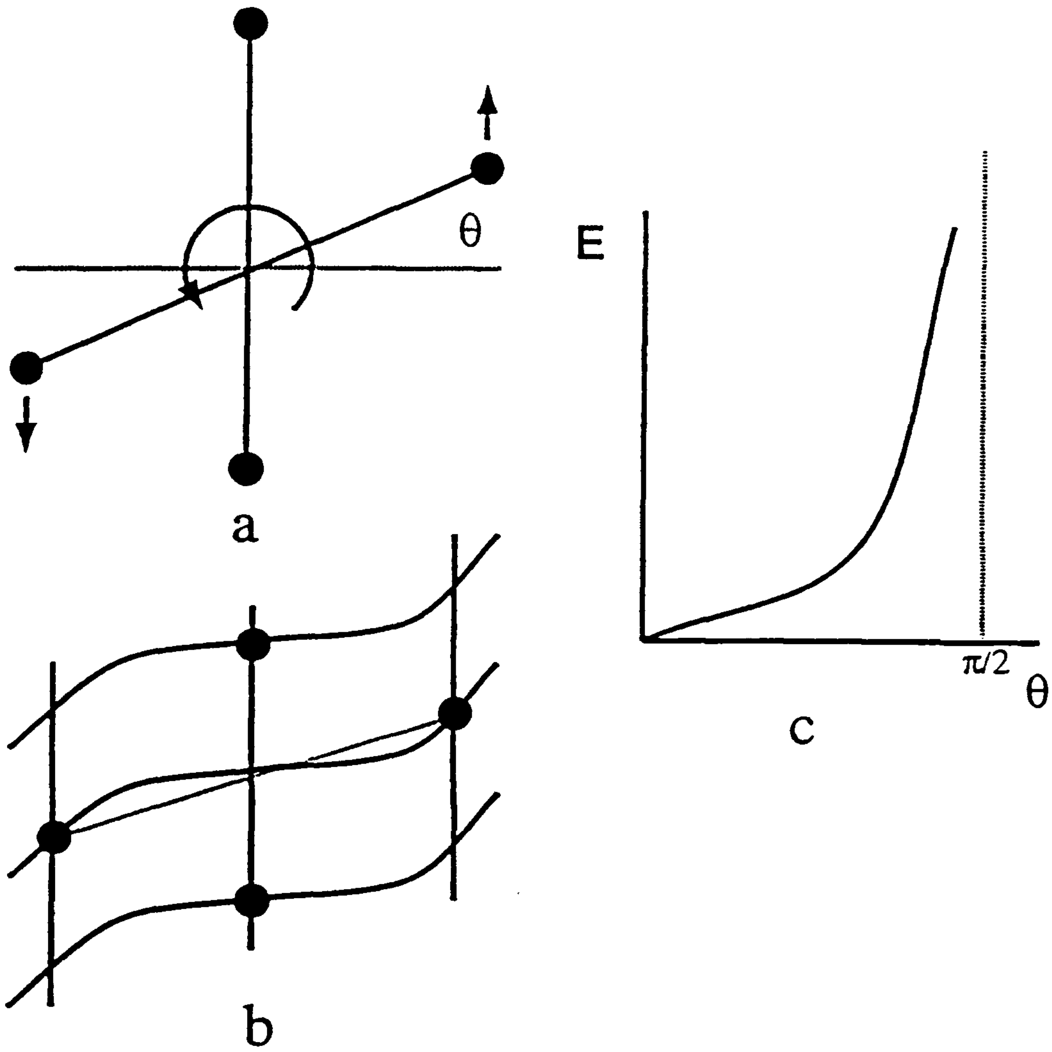
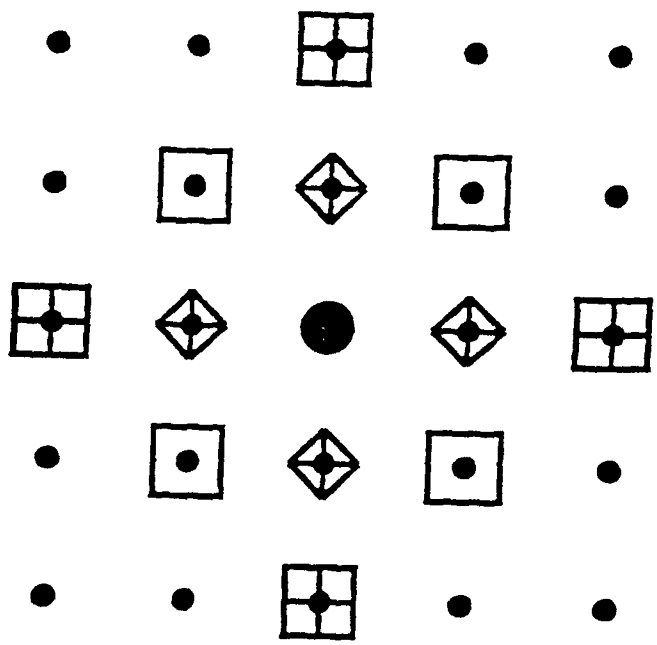


Figure 6 The Trellising Potential



Particles needed to calculate energy change. ◇ - stretching, + - bending, □ - trellising, • - repelling.

Figure 7 Particles Needed to Calculate Energy Change a diamond shape. To calculate the change in the energy of bending after a particle is moved, requires the positions of the eight nearest neighbors along the "thread" lines. These eight neighbors are marked with crosses. Every particle affects the trellising energy of four neighboring trellis modeling cells. The particles of those cells are marked with boxes. The positions of all of these particles are needed when calculating the change of energy of the particle system as a whole, when one particle is moved.

2.1.5 Evaluation Method

The four inter-particle energy relationships defined in our model embody the mechanical and geometric relationships between threads at their crossing points in a woven fabric. Structuring these thread particles in a 2-D grid, placing them into a simulated gravitational field, and allowing the grid of particles to interact with solid models, produces a simulation of buckling and draping cloth.

We have implemented our cloth simulation as a two-phase process operating over a series of small discrete time steps. The first phase of the simulation for a single time step models the effect of gravity, and accounts for collisions between the cloth model and a geometric model defining the objects with which the cloth is interacting. The second phase enforces interparticle constraints and moves the configuration into a local energy minimum before continuing with the next time step. See figure 8.

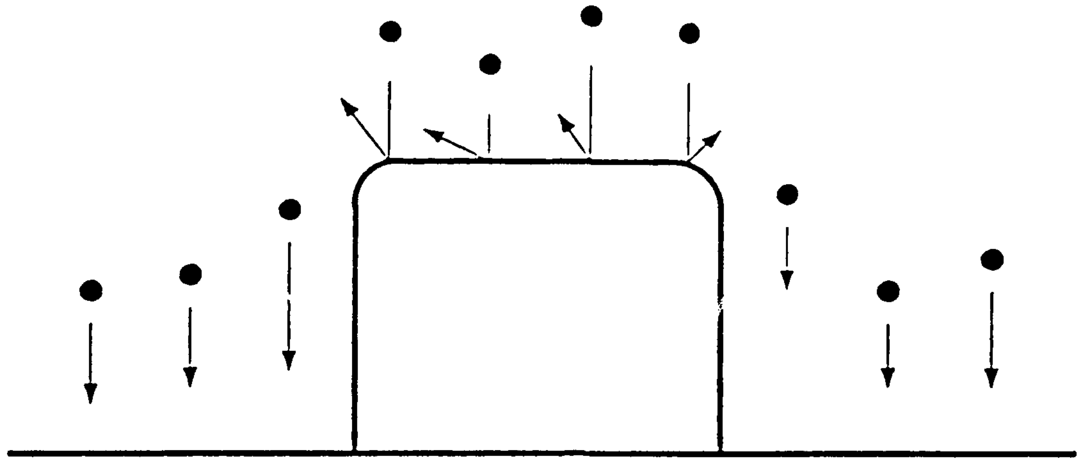
Phase one is simply an expedient way to produce large-scale motions of the grid in the direction of gravity, thus quickly placing the cloth in an approximately-correct draping configuration. During this phase, the inter-particle constraints are removed and each particle is allowed to drop as if in free fall. This fall is modeled by the differential equation

$$M\,a + C\,v = M\,g, \tag{10}$$

that takes into account the first two time derivatives of the particle's distance (acceleration a and velocity v), mass M, gravity g, and viscous drag resistance due to air C. Our simulations use the analytical solution to equation 10, but there is no reason that a more complex equation could not be used and solved numerically.

During this phase, we pass the ray-segment formed from the starting and final positions of each particle to the intersection routine of our ray tracer [11]. If the ray-segment intersects any of the geometric objects, the particle most probably collides with that object during the time step. However, as shown in figure 9, the ray intersection point is only an approximation to the actual curved path that the particle might have followed during the time step. We assure an accurate intersection point by an iterative process. First, we reduce the time step to the time of collision estimated by the distance along the ray-segment at which intersection occurs, and repeat the free-fall calculation for the particle in question. This process is repeated until we have an intersection point that lies exactly at the end of the ray-segment, or we have determined that no collision actually occurred. When an accurate collision point is determined, a partially inelastic reflection of the particle is calculated and used to adjust the particle's velocity, and the "bouncing" particle is allowed to continue its fall for the remainder of the time step. If another collision is detected, the process is repeated until the time step is exhausted.

Once the first phase is completed, and every particle has been moved one time step according to the equation 10, the inter-particle constraints, as defined by our energy functions, are enforced by subjecting the entire particle system to an energy minimization process, in which each particle's position is adjusted so as to achieve a local energy minimum across the model. When the entire process is complete, particle velocities are adjusted to account for each


a

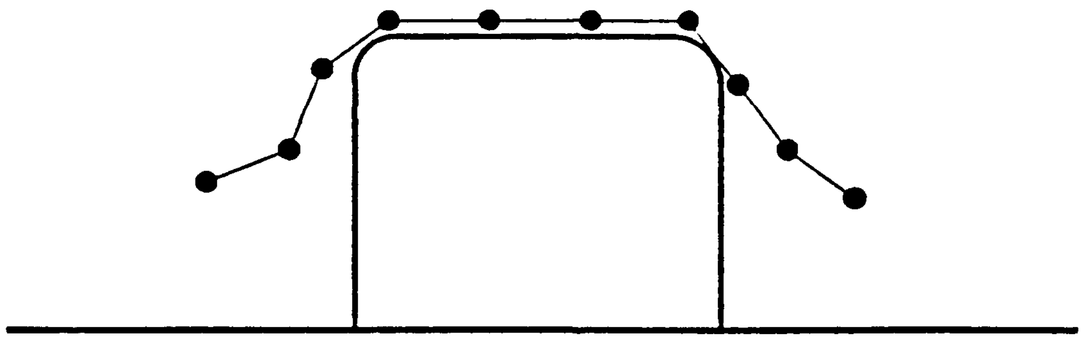

b

Figure 8 Two Phase Evaluation Method

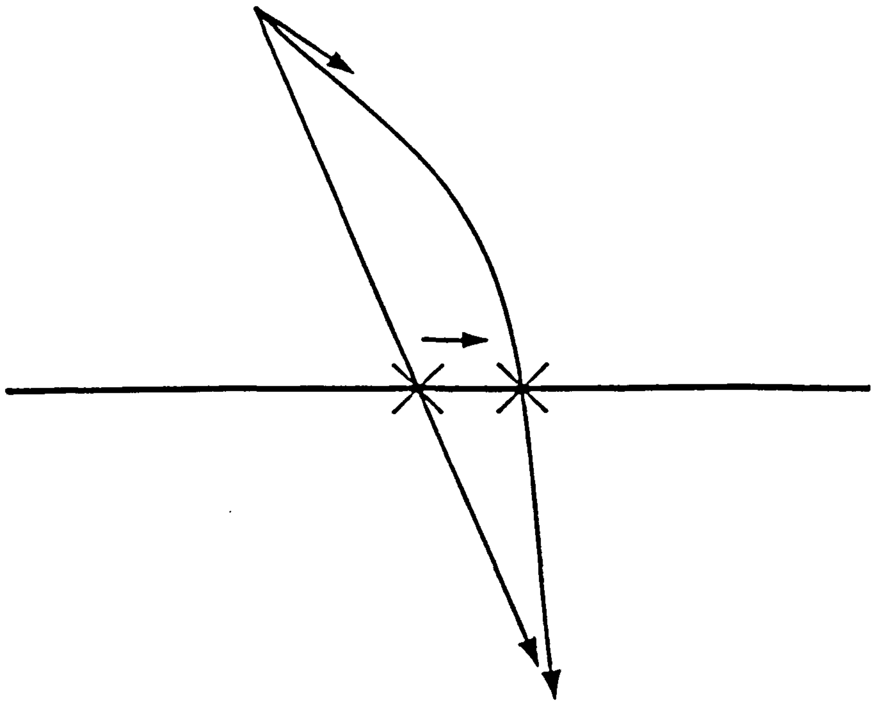


Figure 9 Intersection of Particle Path with Geometry

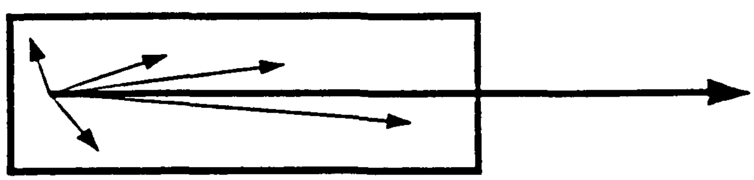


Figure 10 New Position Selection in SGD

particle's position change. This minimization process pulls the grid together and maintains the geometric constraints and relationships [32]. The first phase quickly moves the grid through space and into contact with the objects in the environment, giving the grid its general drapingshape. The second, energy minimization, phase fine-tunes the model producing its more complex and detailed folding structures.

We have implemented a stochastic technique for the minimization step in our simulations, that approximates the process modeled by the Metropolis algorithm [24]. We call this technique stochastic gradient descent (SGD). The Metropolis algorithm randomly perturbs each state in a system. For each perturbation, the change of energy in the system is calculated. If the energy change is negative, the new perturbed point is accepted, but if the energy change is positive, the point is probabilistically accepted based on the Boltzmann factor associated with that energy change. Our experiments with this method showed that it was much too inefficient for our purposes, requiring very large numbers of iterations to bring the particle grid into a satisfactory minimum energy configuration.

SGD has proven to be more effective. In this method, the gradient of the energy function is calculated numerically and is used to construct a box around the region of space that approximates the volume into which new positions that would be accepted by the Metropolis algorithm would fall. This construction is shown in figure 10. Once this box is constructed, we use a uniform distribution to stochastically select a new particle position in this region of probable minimum energy configuration. This technique will generate position changes that are drawn from a probability distribution quite similar to the Boltzman distribution that the Metropolis algorithm generates.

Our intention for developing SGD was to produce an algorithm with the same characteristics as the Metropolis algorithm, but with improved efficiency. The Metropolis algorithm produces a distribution of energy states by accepting all lower energy states and a percentage of the higher energy states based on the Boltzmann distribution. It randomly generates new potential states with no regard for the underlying energy functions in a system and then picks appropriate states and throws away the rest. SGD stochastically generates new states in a region determined by the negative gradient, increasing the likelihood that these states will lower the energy of the system, and accepts all of them. By adjusting the parameters of the algorithms, both can produce the same ratio of higher energy states versus lower energy states. Our experiments have shown that SGD is at least 10 times more efficient than Metropolis for minimizing our energy functions, but still gives a natural looking stochastic variation to the resulting configurations.

During the energy minimization phase it is necessary to again prohibit particles from moving through the surface of geometric objects. We do this in two ways. First, when estimating the energy gradient about a particle we associate a very high energy penalty with movement into a geometric object. This makes it unlikely that we will generate a new particle position inside a geometric object. Second, we do another ray-segment intersection test on each potential new particle position and simply throw away any position change that crosses the boundary of an object.

2.1.6 The Role of Visualization

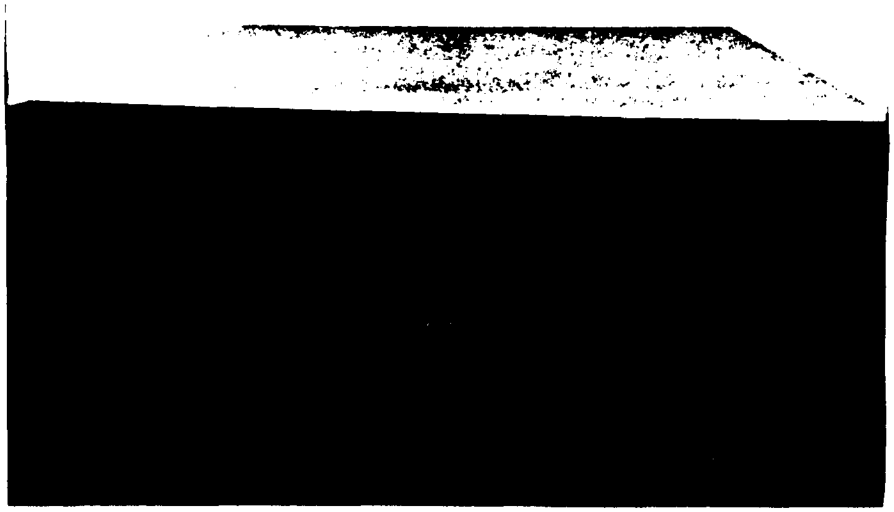
Visualization is an essential part of the experimentation with and evaluation of our cloth model. Our model has been implemented in a testbed visualization and animation system, The Clockworks [12]. By conducting our simulations within this system, we have access to a variety of visualization tools. Since the particles in our model are arranged in a grid, it is easy to construct a polygonal surface for visualization. The Clockworks provides interactive wireframe and Gouraud shaded polygon viewing capabilities, allowing us to quickly inspect the details of the particle grid. Our models are fairly large and we are currently unable to follow the progression of a simulation interactively on our Silicon Graphics Iris 4D/60T. Therefore the system has been interfaced to a single-frame videotape system. This allows us to record animations of our simulations progressing through time directly from the SGI. These animations have given us insight into the workings of the model and how it interacts with solid objects. Finally, in order to fully inspect the detailed behavior of our model we generate high quality ray traced images from the simulation data. These images give us useful information concerning the shape, continuity and curvature of the folds generated by our model.

Our experiments with the model relied heavily on visualization for evaluation. Visualization offered us some insight into the importance of the various components of the particle energy functions. The first attempts at minimizing the energy of the particle grid with the Metropolis algorithm produced an amusing "mozzarella cheese" model where inter-particle distances were not maintained. As particles struck a solid cubic object, they stopped, while those not striking the object continued to move toward the floor, stretching out the model and making it look like melting cheese over a block. Once the Metropolis algorithm parameters were tuned and the stretching function adjusted, we could create a grid that maintained one equilibrium distance between all connected particles. At this point the importance of trellising became apparent. At the corner of the cube where one would expect folds and buckles, we found high trellis angles which in essence were decreasing the surface area at that point. Instead of the cloth buckling and folding to adjust to the excess surface area collecting at the corners, it was "trellising" out to produce a tongue of cloth that simply drooped to the floor. Once stochastic gradient descent was implemented and applied to the same simulations, we found that the trellising constraint was enforced.

Two successful simulations were calculated and visualized. In the first simulation we drape a 50x50 square particle grid over a cube. In the second simulation, the 50x50 grid is trimmed to produce a grid with a circular outline and then draped over a cylinder. Figure 11 presents four instances in time from the first simulation and figure 12 presents four instances from the second simulation. All of these images are ray-traced. Animations demonstrating the complete simulation for both cases have been recorded. For the circular case, the animation shows the stiffness of the cloth in the direction of the yarns. The grid remains fairly flat during most of the simulation in those outer regions where the yarns run approximately radially out from the center. On the contrary, the cloth folds quite easily at the 45 degree angles from the yarn directions. The animation shows that once the cloth comes to rest the folds becomes fairly evenly distributed.

We are also developing other visualization tools to assist us in our research. One will generate a pseudo-color texture map on the polygons that approximate our cloth model. The coloring will display how the energy functions vary over the grid surface. This capability will allow us to isolate each component of our energy functions and to determine the relative importance of each of them at different locations on the grid. Another texture map feature is being developed to allow us to map an arbitrary surface texture on the polygons defined by the grid in order to enhance visual realism.

The importance of all of these visualization tools cannot be underestimated. Currently, we are evaluating the correctness of our model by examining the previously described visualizations. In the future we intend to recreate experiments conducted in the 1960's on draping cloth [6][14]. We will need to "see" our results and visually compare them to the results obtained in these earlier experiments.



[illegible]

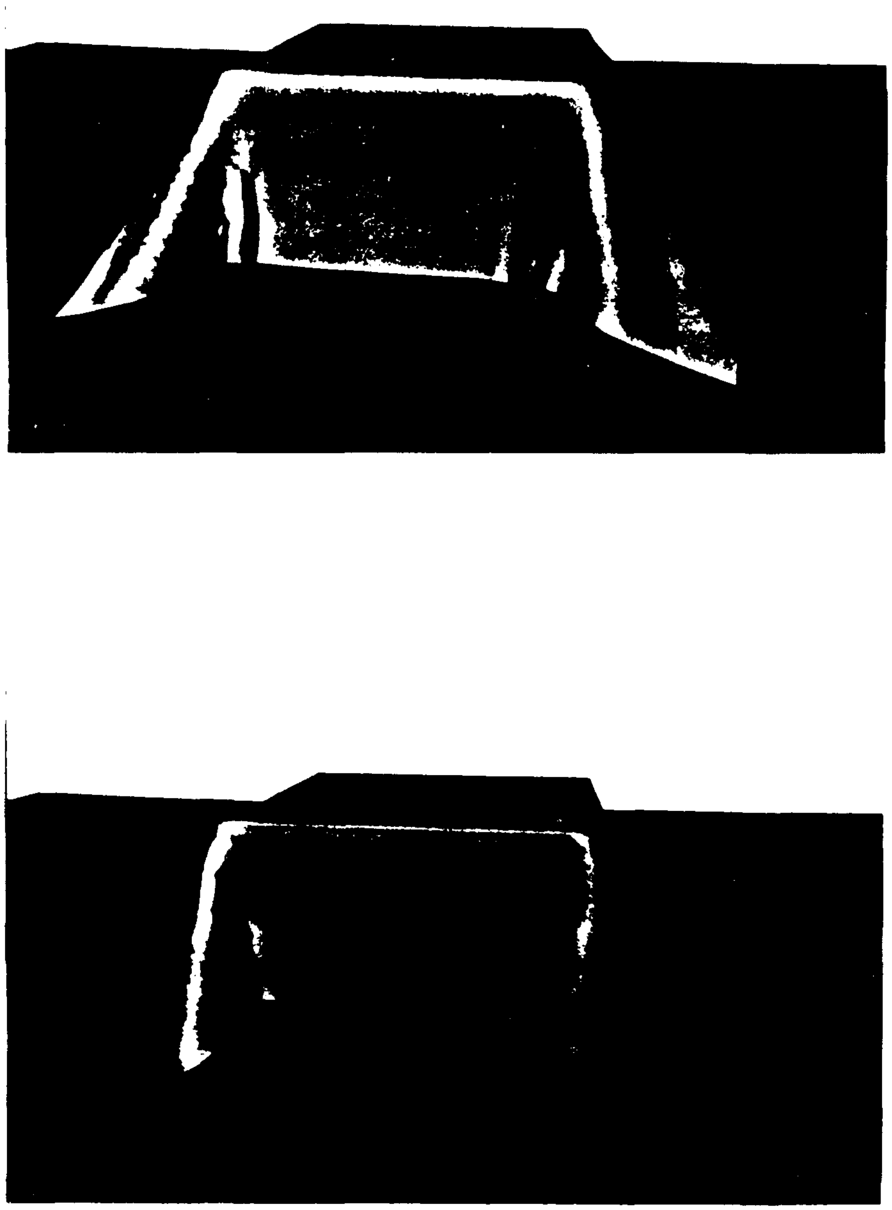
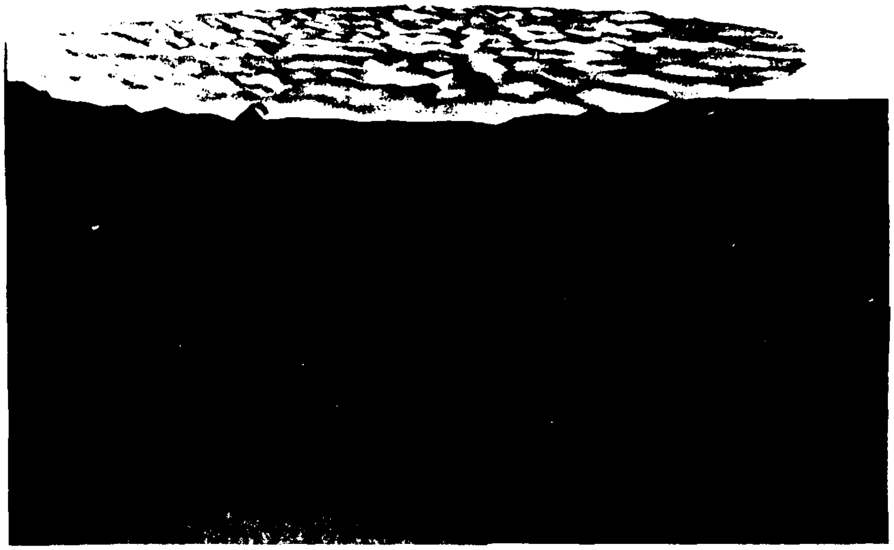


Figure 11 Simulation of a Square Cloth Draping



[illegible] of a Circular Cloth Draping

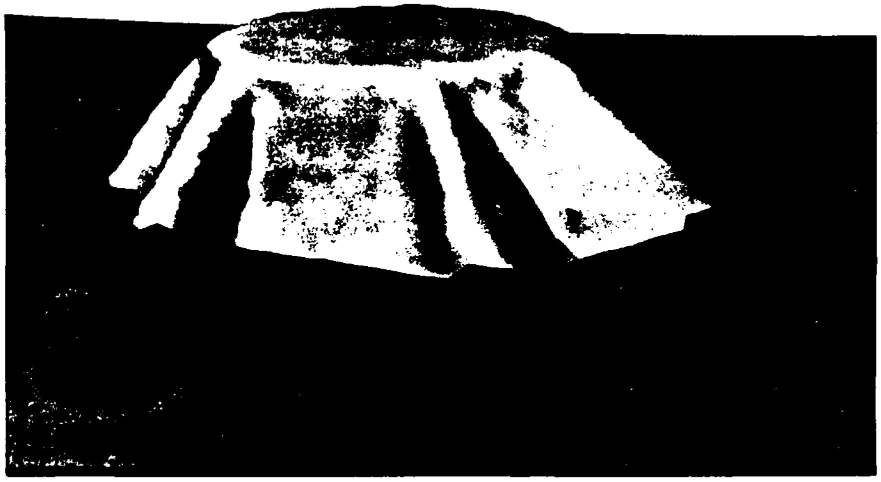


Figure 12   Simulation of a Circular Cloth Draping

2.1.7 Discussion

We have described the interactions between threads with energy functions defined over a 2-D grid of 3-D points, minimizing these energy functions produces the draping and buckling we are attempting to model. At this point in our work, we are not claiming that the interaction functions that we have defined accurately model the energy of threads. Certainly our energy functions have reasonable boundary values. What we are currently doing is testing our basic assumption that by representing a material's microstructure, and by applying qualitatively correct geometric constraints, characteristic macroscopic behavior will emerge. Our visual results show that our assumptions are correct and encourage us to continue our research in this area.

Once the model is refined, extensive simulations will be conducted in order to recreate studies by Cusick [6][14]. His numerous experiments on the draping behavior of cloth produced quantitative characterizations of draping behavior. We will consider our models correct once they give the same results that Cusick produced through experimentation. Given that the functional relationships of our model have stabilized, it will be necessary to tune the parameters of the energy functions, so that the generic models may mimic specific types of cloth.

Of previous studies of cloth, the model that is most similar to ours is by Feynman [10]. He presents a method for modeling the appearance of cloth that uses the same computational framework as our technique. He simulates some of the mechanical properties of cloth by defining a set of energy functions over a 2-D grid of 3-D points. He minimizes the energy of the grid with a stochastic technique somewhat related to ours, and a multigrid method. This is where the similarities end. Feynman assumes that cloth is a continuous flexible material and derives his energy functions from the theory of elastic shells. We do not believe that this is the correct starting point for modeling cloth. Feynman's energy equations also do not take into consideration a phenomenon that we see as essential for modeling the mechanical behavior of cloth, the trellising of threads. By modeling cloth's thread structure and its trellising, our model naturally buckles, removing the need for a special and questionable energy of buckling defined by Feynman. Two additional issues presented in our work that are not present in Feynman's are self-intersection and interaction with constructive solid geometric objects.

One significant component missing from our approach is the modeling of frictional interactions between threads. Our model already incorporates parameters for representing the slippage and sticking of crossed threads due to trellising. It needs to be extended to do the same for thread length. Once frictional slippage is implemented, the model will be able to represent the ability of cloth to be shaped and formed.

An important issue that has not been addressed so far is the issue of computation time. Both the square and circular simulations required on the order of one CPU-week of computation on a DECstation 3100 class workstation. Clearly these are not computation times conducive to design automation. Simulating the mechanical behavior of complex materials requires enormous numbers of computations. However, if we assign each particle to a processor, our particle-based approach maps naturally onto the architecture of a massively-parallel computer. It was this fact that motivated us to explore the concept of particle-based modeling in earlier work [17], where we investigated the task of modeling human cartilage for a surgical simulator. The future implementation of particle-based models on a massively-parallel machine holds out the hope for real-time simulations of complex materials. We have begun to explore this idea by implementing a simplified version of our model on the Connection Machine [16]. So far, our investigations have revolved around experimenting with different implementations of the Metropolis algorithm on the CM-2, and have produced encouraging results [18].

2.1.8 Conclusion

We have proposed a model that simulates the draping behavior of woven cloth. The model attempts to capture the thread structure of cloth, and assumes that cloth is a mechanism, not a continuous flexible material. The model consists of particles (points) placed in a 2-D grid, where each particle represents a thread crossing. Energy functions are defined that embody the interactions of the threads. By minimizing these functions and allowing the particles to interact with solid objects, the draping and buckling behavior of cloth may be simulated.

Our current results are extremely encouraging and provide a convincing demonstration of the correctness of the assumptions upon which our model is based. There is still much work to be done. The most obvious addition that needs to be made to the model is frictional slippage. Second, we need to take advantage of the considerable literature on the experimentally determined mechanical properties of threads and yarns to refine our energy functions. Finally, in order to achieve reasonable simulation speed, and to take full advantage of the architecture of the model, we need to implement a complete version of the model on a massively parallel computer.

2.2 SENSING

The objective of the first year of sensing research is to determine the sensing techniques that are appropriate and adequate to the pressing of trousers on a "legger" press. The sensing techniques must enable the acquisition, manipulation, positioning, and smoothing of the trousers during the loading and pressing procedures. Specifically, sensing is used to locate and align in-seams relative to the out-seams prior to placing the trousers on the pressing form. After the trousers are placed on the form and prior to being pressed, the trousers must be verified to be free of significant wrinkles, puckers, and the like. Each required sensing activity is to be accomplished separately.

To facilitate proper sensing testing we solicited fabric from a variety of sources and then had the fabric cut and sewn into completed trousers. The fabric was received with full documented specifications and a swatch book was developed for R.P.I. and F.I.T. by student researchers.

2.2.1 Definition of Sensing Requirements

Evaluation of the automated garment handling unit resulted in the proposal of the following sensing requirements: seam sensing, wrinkle sensing, and force sensing.

Sensing Requirements

The seam sensor is utilized for locating the inner and outer seam of each trouser leg for proper alignment prior to pressing the garment. The seam sensing and alignment take place prior to transfer of the trousers onto the press buck, and is performed on each leg with the trousers in a hanging orientation external to the press. There are at least two places where the seam alignment needs to be accomplished, at the hem (or cuff) end and above the knee (in some instances, to within 5 cm of the crotch area).

The detection of wrinkles and other departures from the desired surface contours is performed when the trousers are on the pressing form. If the placement of the trousers on the form is sufficiently consistent, only a surface smoothness validation check is required. If the placement produces wrinkles, then the detected wrinkles need to be described in terms of location, orientation, height and length.

2.2.2 Background

The two major areas of research during the past year have been the seam sensor and the wrinkle sensor. Several methods were initially proposed and pursued for each of these sensing requirements. Investigation of the different methods led to a single method for the wrinkle sensing and a standard method for the seam sensing with a backup method to handle a unique and infrequently encountered fabric.

Sensors

It is desirable to employ the most simple and robust sensing techniques to accomplish the sensing required. Three fundamental sensor types have been identified: a reflected light sensor for seam location, a transmitted light sensor for both seam and selvage location, and a reflected light sensor for wrinkle detection. It is clear that reflected light sensors are easier to apply than transmitted light sensors. Reflected light sensors can be applied entirely from the outside of the trousers; whereas, transmitted light sensors need to be introduced inside the trousers.

Thus it would appear that reflected light sensing is the sensor type of choice. However, the results to date do not support the notion that reflected light sensing alone can locate the seams for all kinds of fabrics which are under consideration. In particular there is some difficulty with the reliable detection of the seam location for some samples which have pronounced visual or geometric textures. Therefore, to be prudent, transmitted light sensors continue to be considered as a back-up to the reflected light sensors. Further, transmitted light sensors can serve a redundant capability, providing robust performance in the event of unforeseen difficulties.

Reflected Light Sensor for Seam Location.

The seam can be located by analyzing the reflected light in the vicinity of the seam. This is done for both the in- and out-seams at both the hem end and above the knee. This location information is passed to the seam alignment mechanism to bring the seams into alignment. Locating the seam above the knee is especially important. This is because the sewn fabric is not usually distributed in a symmetric fashion relative to the seam. Hence, the seam location is not guaranteed to be predictably related to the mid-line of the seam allowance boundary; it must be located by more direct means. The reflected light sensor is the preferred method for seam location.

Transmitted Light Sensor for Seam Location.

The sensor detects the amount of light that passes through the fabric. For most fabrics, the amount of light that shines through a single ply is reliably more than when there are two or more plies. This sensor can locate the boundaries of the double-ply region of the seam allowance portion of "busted" seams.

This sensor can also be used to verify the location of the seam relative to the seam allowance boundaries. This is done by detecting the light that shines through the gaps in the sewn seam when the trousers leg is put under spreading tension, as in the case when the leg undergoes manipulation for seam alignment. However, in a small number of cases, the seam thread tension is too tight and a detectable amount of light does not shine through the seam gaps.

Reflected Light Sensor for Wrinkle Detection.

The detection of wrinkles and other departures from the desired surface contours is performed after the trousers are placed on the pressing form. Wrinkle detection currently employs a plane-of-light stripe projector. The light stripe sensor is scanned along the trouser leg to verify its smoothness. An analysis of the projected light patterns allows the presence of departures from the expected smooth surface contour to be detected and described. The location, height and orientation of puckers, creases, wrinkles and the like is passed to the handling module.

Force Sensing.

Force sensing has not been pursued at this stage of the project since it is used in conjunction with grasping and transfer mechanisms. The need for force sensing will be pursued upon the completion of the concepts for the grasping and transfer mechanisms.

2.2.3 Seam Sensing

Seam sensing requires accurate location (+/- 0.125 inches) of the seams on both sides of the trouser leg for alignment. Several methods have been proposed for seam sensing. Most of these proposed methods can be eliminated due to measurement inaccuracy, inconsistent performance or implementation complexity. The proposed methods include mechanical location of the seam, mechanical detection of multiple plies , transmitted light through seam detection, one ply versus two plies detection, transmitted light detection, laser light technique, and exterior seam-shadow detection.

Mechanical Location of the Seam.

The location of the seam can be determined by determining where the two pieces of fabric meet for a seam. Mechanically this can be done by sliding a knife-edge along the surface of the fabric until the resistance of the folded seam is sensed. This same action could be done to locate the seam from the opposite direction as a confidence check. The opposite seam would be located in the same manner. Some experiments have been performed that indicate this technique could be used for heavier weight fabric which have well-formed seams. Such a device might be incorporated as an integral part of the rotating seam alignment mechanism which is described in the Grasping and Manipulation section.

Mechanical Detection of Multiple Plies.

The detection of the differential thickness of multiple plies formed by the seam might be detected by mechanical means. An instrumented roller could be used to measure the thickness of the trouser leg material as it is being rotated by the seam alignment mechanism. Some experiments have been performed that indicate this technique could be used for the heavier weight materials to locate the multiple ply selvage areas. However, as discussed next, the detection of one ply versus two plies does not really solve the problem of locating the seam. Such a device might be incorporated as an integral part of the rotating seam alignment mechanism.

One Ply versus Two Plies Method.

The one ply versus two plies method utilizes an interior light in the trouser leg.The light is a high intensity gas lamp with an aluminum housing and two plastic diffusion plates to permit simultaneous illumination of both the in- and out-seams with the same light. The overall dimension of the light source is approximately 3 x 4 x 0.75 inches for easy placement within the interior of the trouser leg. The light intensity is great enough for detection by an exterior camera. The light passes easily through a single ply of fabric, but is more attenuated by the two plies of fabric located on either side of the trouser seam. The camera image of the transmitted light then contains a dark central strip, indicating the two non-transparent plies, with lighter areas to either side of the strip, indicating the one transparent ply. The seam is estimated to be at the mid-point of the two-ply vertical strip. Testing of this method on a variety of trousers produced very inaccurate seam location results, due to improperly busted seams and an uneven amount of excess fabric on either side of the seam in the majority of trousers tested.

Transmitted Light Detection.

This method uses the same light as the one ply versus two plies method just discussed. The light is placed on the inside of the trouser leg to transmit light through the seam gaps for location of the seam. The application of an appropriate tension force across the seam of the trouser leg will cause the seam to separate slightly and create gaps between the threads in the seam stitching.

The light from high intensity gas lamp on the inside of the leg shows through the seam gaps. In the image, these gaps appear as a broken line of light in the darker two ply area. Appropriate algorithms are applied to extract the broken line of light from the image. This method has not been fully tested. It is more complicated than the seam shadow technique discussed below. Transmitted light sensing requires the insertion of the light into the trouser leg and more extensive computer processing of the image data.

Laser Light Method.

The laser technique utilizes a single laser line scanned across the seam. The line of light is distorted at the seam due to the change in the trouser surface at the seam, producing an extremely accurate seam location value. This method was tested on many pairs of trousers. It was observed that, in some instances, the surface change produced by the presence of the seam is not significant enough to produce a detectable distortion in the light. This is most probably due to either the light weight of the fabric or the high sewing tension of the thread binding the seam.

Seam-Shadow Detection.

The seam shadow technique utilizes an exterior light at an angle to the trouser leg to cast a shadow along the length of the trouser seam. An infrared filter is added to the camera lens to aid in distinguishing the cast shadow from any stripes or other fabric pattern markings that are parallel to the seam direction. The infrared filter blocks light below 850 nanometers, which includes the majority of color variations in materials. The external light used to create the shadow contains enough light energy above 850 nanometers (light not blocked by the filter) to be effective in illuminating the trousers. Testing of this method proves it is fast, simple and reliable in all but one situation. The exception case is caused by the presence of vertical red threads in trousers. The infrared filter passes red light above 850 nm through to the camera; the red threads reflect enough energy above 850 nm to appear on the image as vertical lines parallel to the seam shadow.

Despite this one anomaly, the seam shadow technique appears to be the most accurate and reliable of all techniques proposed. Thus the seam shadow technique has been the chosen technique for further research. The problem of handling fabric with vertical red stripes has been handled as an anomaly for now. It will most likely be solved using other light source-filter combinations or it could be solved by utilizing the transmitted light through seam technique as a backup method when trousers with red vertical threads are encountered. Algorithms have been developed for the seam shadow technique; but algorithms have yet to be developed for the transmitted light through seam technique.

2.2.4 Wrinkle Sensing

The sensing of wrinkles requires recovering three-dimensional information from a two-dimensional image. The problem is a significant one, but can be simplified through the use of structured lighting and stereo vision, both well-documented methods for recovering three-dimensional surface information.

Wrinkles in a flexible material can be viewed as surface height changes across the material, whereas a flat material will display no surface height changes. The projection of a high-intensity light onto a non-flat surface will result in geometric distortions of the light pattern at points of surface height change of the material. By locating the camera and light projector at different points (commonly referred to as triangulation stereo vision), the light pattern distortions on the surface of the object are detectable in the two-dimensional image acquired by the camera. The light pattern distortions seen by the camera are proportional to the actual surface height changes of the material. The three-dimensional surface height data is recovered through processing of the two-dimensional image.

Construction of Reflected Light Sensor

A laser scanner was originally proposed for use as the reflected light sensor for projecting the light pattern onto the fabric. The benefits inherent in using a laser are the ability to reprogram the scanner to produce a variety of light patterns, the ability to produce complex patterns, and the high intensity of the light. The created laser light patterns are designed to cover wrinkles at all angles and of all sizes. A computer program was successfully written to produce spiral, Lissajous (figure-eight-like) and snake-like patterns. The patterns are fairly complex and require a large minimum number of points to be correctly displayed. This results in a significant amount of time that is required to display a single pattern. This time requirement results in a significant amount of flicker in the viewed pattern. Such flicker is caused by the entire light pattern not being fully captured by the camera in a single image. The problem is not resolvable without going to more expensive scanning equipment; so a second means to produce light patterns is proposed. The second approach for the light source is a high intensity light source and lens which produces a high intensity line of light. When projected across a trouser leg, the ten inch wide line of light will be displaced where there are changes in the fabric height. Processing of the image produces information about the height and width of the wrinkles at each image sampling point. By scanning the light line down the length of the trouser leg, wrinkle information may be extracted at the appropriate sampling frequency. Interpolation of the data between the sampling points will provide wrinkle orientation information.

2.2.5 Algorithms for Sensors

The majority of time has been spent writing and testing algorithms for wrinkle and seam sensing. The algorithms may be grouped into three operations: seam detection, wrinkle detection, and modeling of the buck (lower pressing surface of the trouser press) and wrinkles. The modeling algorithms have been developed to prove the reliability of the methods for acquiring surface information for wrinkle detection.

The seam sensing and wrinkle sensing algorithms were originally developed using the GMF Karel and Insight system for convenience. For processing speed, the ability to handle a wider range of operations, the ability to store a larger number of images, and future compatibility, the algorithms were re-written in C language and compiled and processed on the IBM host computer. The original Karel algorithms proved the processing techniques operated properly and produced appropriate results. In conversion to C, several algorithms were modified and many algorithms enhanced.

Seam Sensing: Karel Algorithms

The Karel algorithms for seam sensing were produced to prove that the seam line of the trouser leg could be detected using the seam shadow technique. The Karel algorithms do not attempt to locate or model the seam but do perform image processing to extract the seam from the image.

The original image is first filtered to remove noise and enhance the vertical edges in the image (in this case, the vertical edge is the seam). The optimum filter kernel size for an image with a field of view of approximately 8 inches was determined to be 3 by 3 pixels. Larger kernel sizes (5 by 5 pixels, 7 by 7 pixels, etc.) did not produce as sharp seam contours as did the 3 by 3 kernel. The filtered image has a gray background with a bright contour for the seam.

Then the histogram of the filtered image is determined. The histogram produces the frequency of occurrence of each gray level intensity in the image. The histogram is used to determine where to set the threshold level in the third step of processing. The goal is to produce a binary image with the seam contour at gray level 255 on a background of gray level 0. The seam contour contains the pixels with the brightest intensity levels, and thus are found in the upper end of the histogram. Starting from the upper end of the histogram, the number of pixels per gray level is summed until a given number of pixels has been encountered. This number is determined as the approximate number of pixels that may be found in the seam contour. A given line across the image will contain approximately 256 pixels; thus the sum is determined to be complete when 200 pixels have been summed. The gray-level at which the 200th pixel is found is set as the threshold.

The third step in the processing entails thresholding the image with the gray-level found in step two. Thresholding produces a binary image of the bright seam contour against a dark background.

All images processed using the above algorithms produced images with well-defined seam contours on dark backgrounds. Some noise was evident in the images, but the noise was not significant enough to hamper extraction of the contours.

Seam Sensing: C Algorithms

The C algorithms for seam sensing are similar to the Karel algorithms. A fourth step was added to allow extraction of the seam contour location.

To locate the seam contour following the filtering, histogramming and thresholding of the image, a Hough transform is performed on the image. The Hough transform determines the best line fit for pixels of high intensity, in this case, pixels of intensity 255. The transform works by scanning through possible slope and intercept combinations for each bright pixel. The best fit line is determined by the most common slope-intercept pair. The seam is thus modeled by the best fit line, and the location of that line is known.

Many images were successfully processed utilizing the above algorithms. The following figures are of images at each stage of processing. Figure 13 is the original seam shadow image. Figure 14 is the filtered image. Figure 15 is the thresholded image. Figure 16 is the best-fit line approximation to the curve found in figure 15. From comparison of figures 13 and 16 it is evident that the Hough transform produces lines which accurately model the original seam contours.

Figure 13 Seam Shadow Image

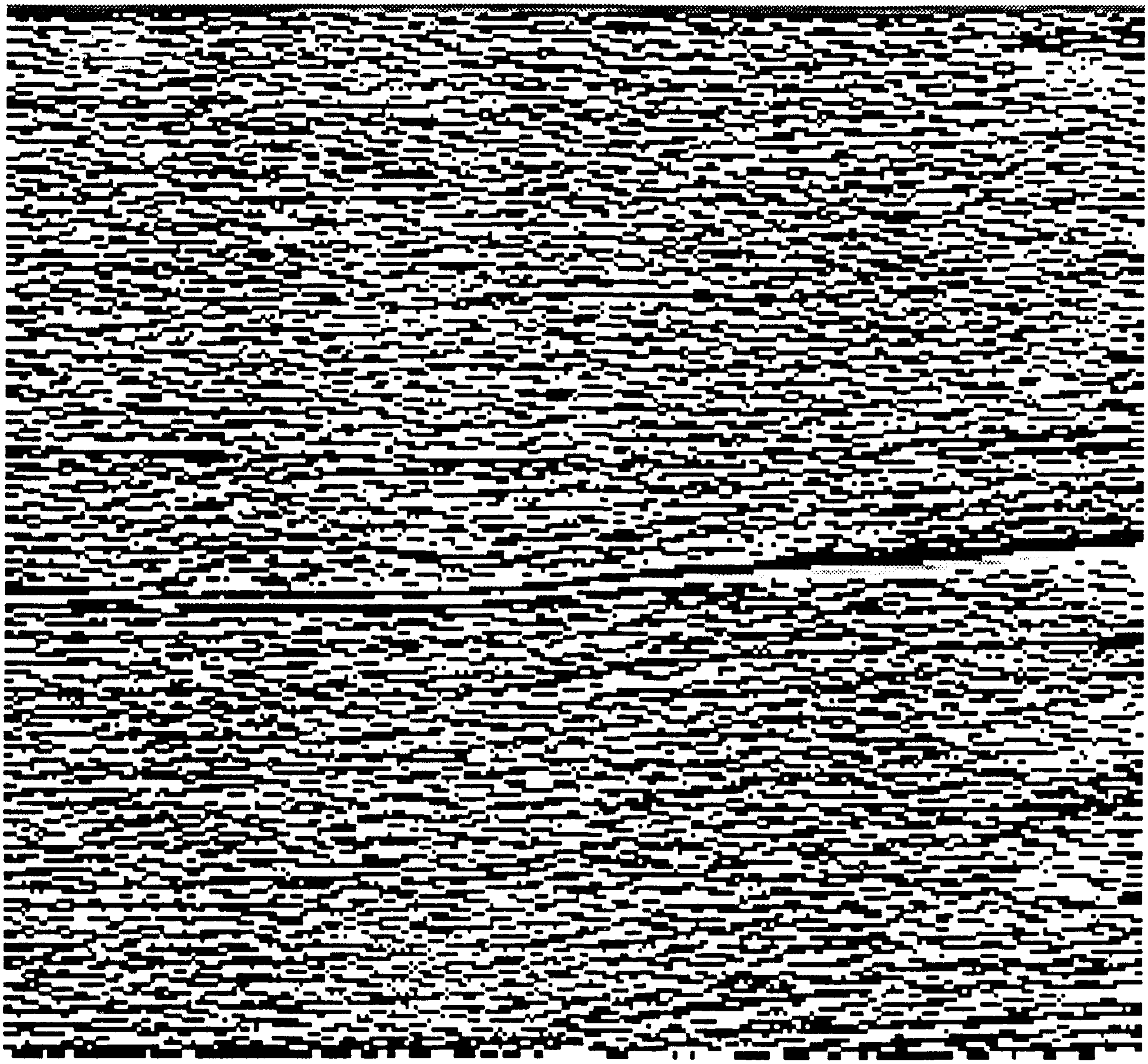


Figure 14 Filtered Seam Shadow Image

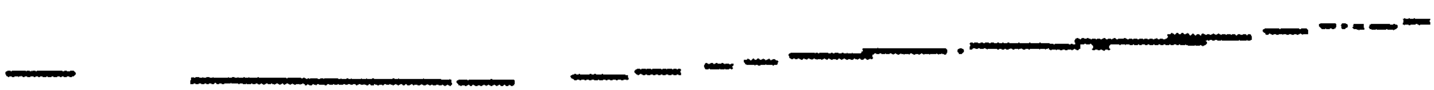


Figure 15 Thresholded Seam Shadow Image

Figure 16 Best-Fit Approximation to Seam Contour

Wrinkle Sensing: Karel Algorithms

The Karel algorithms for wrinkle sensing were produced to verify that the light stripe technique was appropriate for wrinkle detection. Due to the varying curvature of the buck, wrinkle detection requires determining the difference image of the image of the empty buck contour at a particular location and the image of the garment contour on the buck at the same location. The difference image thus contains the same wrinkle height information as would be obtained if the garment had been laid on a flat surface.

The light stripe and camera are initially scanned along the length of the empty buck such that the light stripe falls across the width of the buck.Reference images are obtained at particular locations (the appropriate lengthwise sampling frequency has yet to be determined) and the buck curvature information is extracted at these locations. The buck curvature is seen as a bright contour across the width of the image against a dark background.

The images are filtered to remove noise using a 3 by 3 smoothing kernel whose weights are all equal to 1/9. The buck contour is then extracted from the image by scanning each vertical column for the brightest pixel, the light stripe location for that particular column. The search for the brightest pixel guarantees the center location of the light stripe due to the Gaussian shaped cross-section of the light intensity. The light is brightest at the center of the line and falls off to each side of the center, as observed in the numerous images analyzed.

The above process of scanning the buck, filtering the image and locating the brightest pixel in each column is repeated with a garment on the buck. The images are acquired at the same locations utilized for the reference images.

The difference information between the images with trousers and the reference images is then computed. This yields the wrinkle height information, as displayed in figure 21. A flat reference value is produced which gauges the smallest difference between the empty buck and trousers on buck at any column.

The Karel algorithms then utilize moments to determine points of change along the difference curvature. The image is divided into approximately 42 windows (6 pixels wide by the height of the image) and moments are calculated for each window. The major axis orientation for each window determines the relative slope of the curve segment in the window. Major changes in the major axis orientation indicate changes in the curvature of the curve. Such changes indicate points where the wrinkles begin, the local minima and maxima of wrinkles, the peaks of wrinkles, and the points where wrinkles end.

The algorithm works fairly well. However, the sensitivity of the moment calculations are highly dependent upon the width of the window across which the moments are determined. In addition, the moment calculations are fairly time consuming.

Wrinkle Sensing: C Algorithms

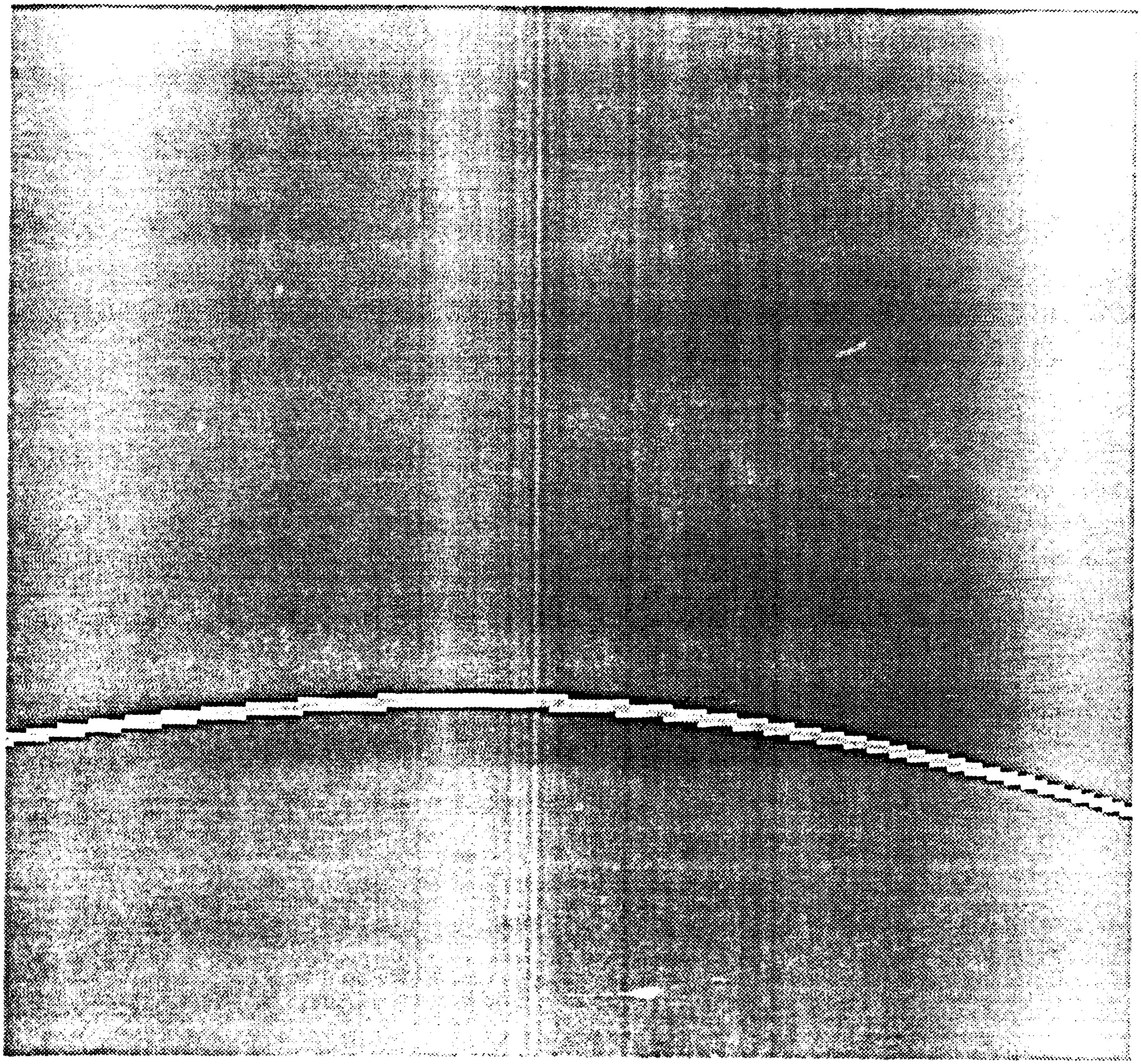
The C algorithms for wrinkle sensing are similar to the Karel algorithms and utilize the same light stripe and camera scanning methods as the Karel algorithms. The steps for obtaining the difference images are identical, but the method for extracting the wrinkle information uses a different approach.

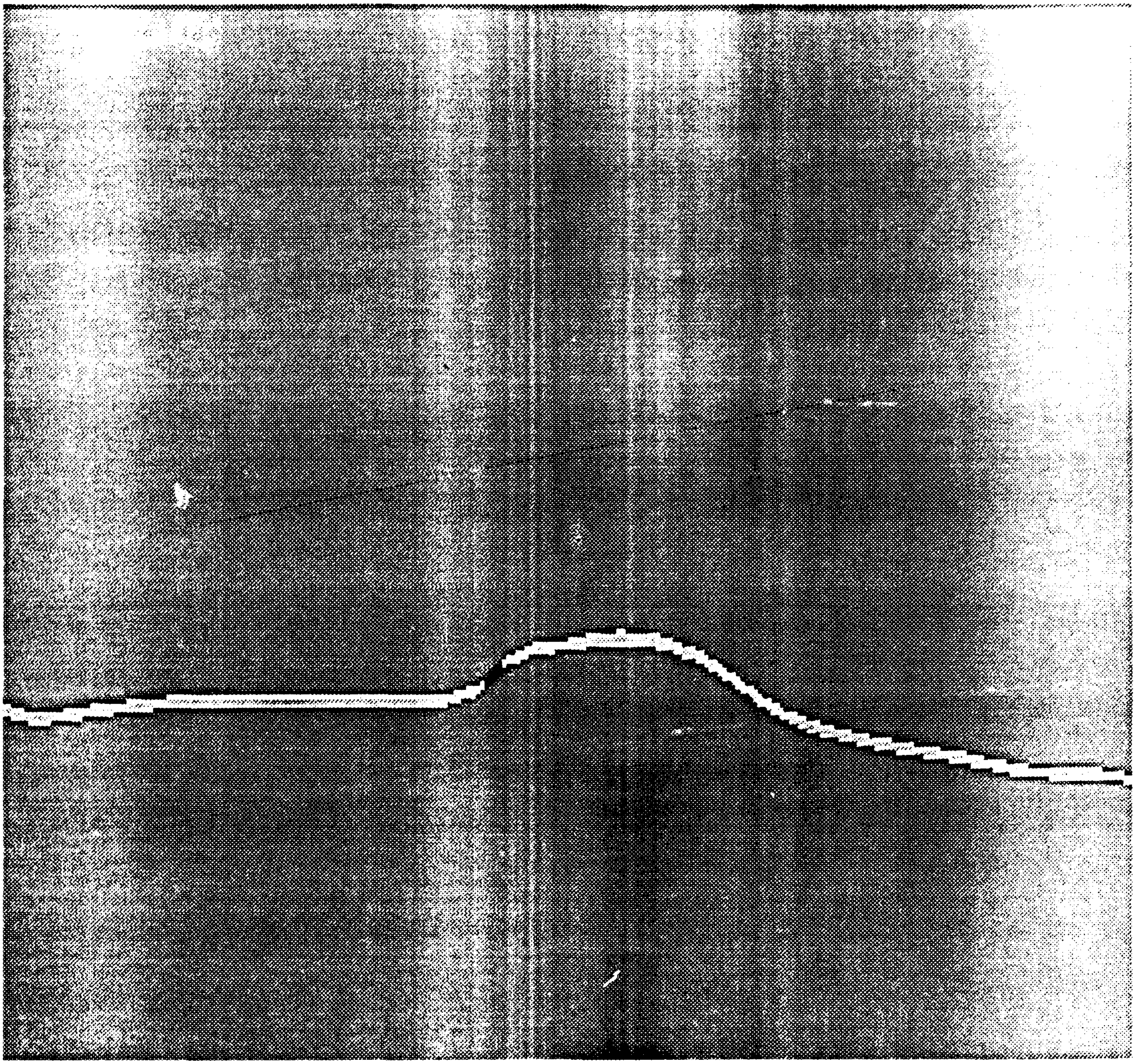
The C algorithms for extracting the wrinkle information uses an analytic approximation to the curvature data. Inflection points are computed from these analytic approximations. Inflection points are points at which the direction of the slope of a curve changes sign and are used to determined points of change in the difference curve. The inflection points will thus indicate points where wrinkles begin and end. The analytic approximation also provides the location of local minima and maxima of wrinkles, and peaks of wrinkles.

Inflection points are obtained by determining the derivatives of the cubic functions which approximate the curves. B-splines are used to approximate the curve data with cubics and each cubic is evaluated to determine where the inflection points are located, if any exist.

The height, width and characteristics of wrinkles can be determined from the inflection data, as well as the number of wrinkles present.

The following figures are images of the light stripe projection and the final images obtained from processing. Figures 17 and 18 are images of the light stripe projected on the empty buck and onto wrinkled trousers on the buck, respectively. Figures 19 and 20 are the curves obtained from processing the images of figures 17 and 18; the curves are extremely accurate estimations of the projected light stripes. Figure 21 is the difference image of the curves in figures 19 and 20, which is equivalent to the wrinkle occurring on a flat surface.





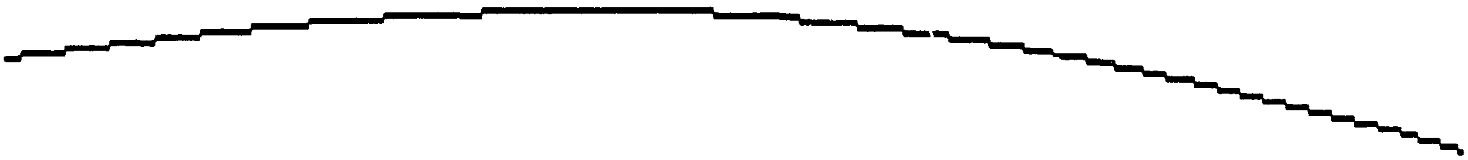


Figure 19 Curve Extracted from Light Stripe Projected onto Empty Buck

Figure 20 Curve Extracted from Light Stripe Projected onto Wrinkled Garment on Buck

Modeling of the Buck and Seams: C Algorithms

The modeling algorithms are developed as part of the wrinkle sensing algorithms to ensure the reliability of wrinkle sensing algorithms. The modeling approximates a curvature in an image and evaluates the approximating cubic B-splines using interpolation. The modeled curvature that is obtained is nearly identical to the curvature in the original image.

The modeling algorithms utilize several of the modules developed for wrinkle detection. The image with the curvature to be modeled is filtered to remove noise. The curvature is then approximated with cubic B-splines. The B-splines are cubic functions which, when evaluated at a point, produce the appropriate curve for that point. A function is obtained for every 4 neighboring points to ensure that only accurate local information is utilized in determining the appropriate function. Through interpolation, the B-splines are evaluated to produce an approximation, or model, of the original curvature.

The approximations are very accurate models of the original curvatures, thus verifying the validity of the B-spline algorithm that was written. The successful modeling, utilizing the same B-spline algorithm utilized in wrinkle detection, suggests that the B-splines are accurate approximations, and thereby,so are the inflection points obtained from the B-splines.

The modeling has been tested utilizing a series of images of the empty buck and images with a garment on the buck. The following figures are of images created using the modeling algorithms. Figure 22a is an interpolation of the buck curve in figure 19. Figure 22b is an interpolation of the wrinkle curve in figure 20. Figure 22c is an interpolation of the difference image in figure 21. Figure 23 is an interpolation using six images to model a wrinkled garment on the buck.

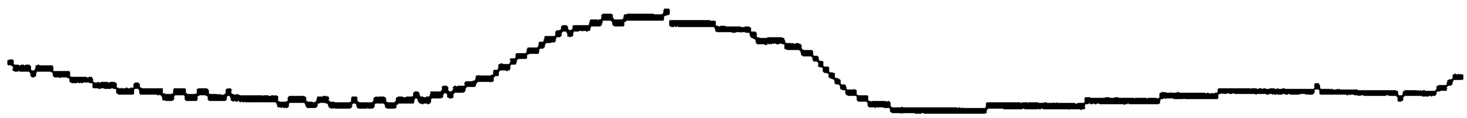


Figure 21 Difference of Curves from Figures 19 and 20

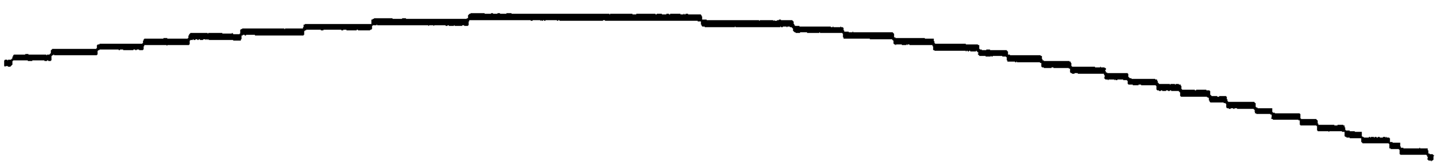


Figure 22a Interpolation of Light Stripe Projected onto Empty Buck

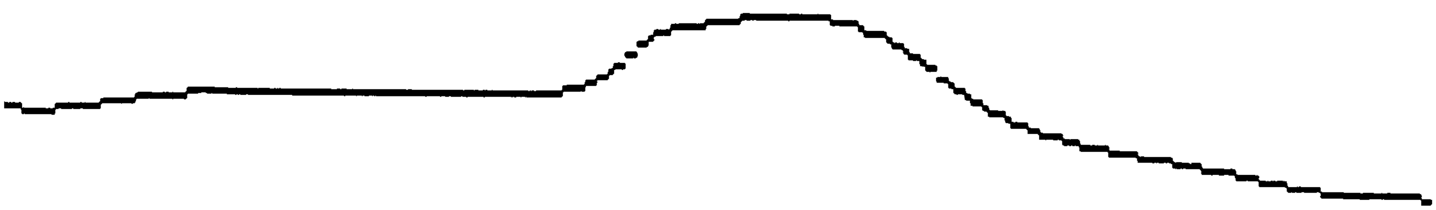


Figure 22b Interpolation of Light Stripe Projected onto Wrinkled Garment on Buck

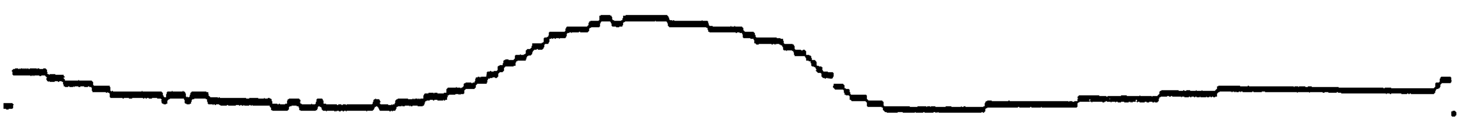


Figure 22c Interpolation of Figure 21's Difference Image

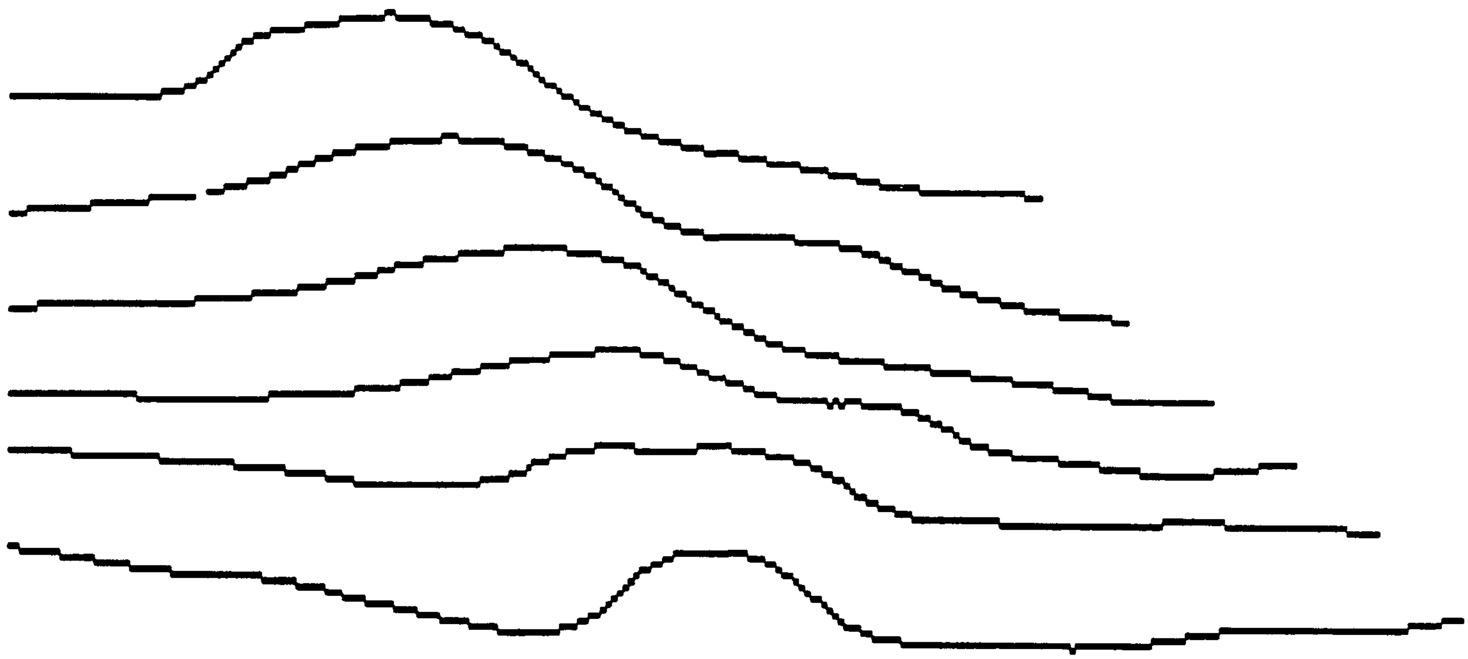


Figure 23 Model of Wrinkled Garment on Buck

2.3 GRASPING AND MANIPULATION

2.3.1 Introduction

The efforts of the Grasping & Manipulation team started by trying to ascertain and understand the current level of technology in the apparel industry. This was done by doing a literature search and touring apparel manufacturing facilities and R&D labs. Next, the knowledge gained was used to conceptualize and develop prototype grasping and manipulation devices that are relevant to this project's goals. Four prototype devices were designed, built, and tested. In general, the result of these tests were promising and they are used to indicate directions for future effort.

2.3.2 Grasping Technology in Industry and Novel Grasping Techniques

The grasping and manipulation team conducted a state of the art review of existing grasping and manipulation technology by doing a literature search and touring garment manufacturing facilities or laboratories.

The places visited included:

1. Pietrafesa (Syracuse, NY) - manufacturing facility for men's garments.
2. Cluett & Peabody (Troy, NY) - Research and Development Lab.
3. Fashion Institute of Technology (New York, NY) - Automated Apparel Production Lab.
4. Jetsew (Utica, NY) - maker of automated apparel manufacturing equipment.
5. Draper Laboratories (Cambridge, MA) - Research arm of (TC)2.
6. Bobbin Show (Atlanta, GA) - trade show of the apparel industry.

During our state of the art review we observed five types of grippers that can be used to grasp and manipulate fabric:

1. Friction - development of a frictional force between the gripper surface and fabric is required.
2. Puncture - the gripper penetrates the fabric to develop holding forces.
3. Adhesive - chemical bonds create the attractive force.
4. Vacuum - atmospheric pressure is used to create a frictional force between fabric and gripper and develops buoyancy.
5. Electrostatic - electrical potential creates an attractive force.

The state of the art review revealed that the adhesive and electrostatic grippers are rarely used, except in closely controlled environments, because of their susceptibility to failure. As far as the implementation of the five gripping methods, there are only a few commercially available fabric grippers. A study conducted by the American Apparel Manufacturers Association showed that, of 47 different limp fabric handling devices, only 3 are currently in wide-spread commercial use. These devices are:

1. Clupicker - a rotating wheel frictionally pulls the fabric ply in between the wheel and a finger.
2. Walton picker - saw toothed edges partially puncture the fabric and then close and clamp.
3. Polytex or Littlewood Needle device - punctures the fabric with needles at different angles so the fabric can't slip off.

These three devices are only for picking up a single ply of fabric. In most applications they are used to grasp and manipulate freshly cut parts from a stack. They were not designed for handling finished garments. However, an application may exist for them where it may be necessary to separate two adjacent plies of fabric.

Draper Laboratories and Jetsew are conducting the work that is most closely related to the work at Rensselaer in that they are developing automated equipment that manipulates finished garments. However, the operations conducted are assembly and sewing and therefore are somewhat unlike handling garments for pressing.

There are many problems that must be overcome to successfully grasp and manipulate trousers in the pressing operation. Perhaps the most challenging problems arise from fabric's handling characteristics. The work at Draper Laboratories, in developing the automated suit-coat sleeve sewing machine, revealed many of these problems.

1. The manipulation of fabric parts requires the use of multiple gripping points.
2. Fabric buckles under its own weight.
3. Excessive manipulation can cause wrinkles.
4. The low rigidity of fabric practically eliminates the application of force feedback or contact sensors.
5. The mechanical properties of fabric, like its stiffness and coefficient of friction, vary from part to part and may even vary across a single part.
6. Mechanical properties are dependent upon environmental conditions like humidity and are time-dependent.
7. Precise alignment can be spoiled by residual stresses in the part.
8. Usually it is inefficient to attempt to duplicate human manipulation motions, although much can be learned from them.

The steps that were taken to counteract these problems were:

1. Gated vacuum tables were used to hold sections of the part from moving while the remainder of the part was being manipulated.
2. A flexible gripper was needed to accommodate various part shapes.
3. Air jets were used for non-contact manipulation.
4. Creasing bars and auxiliary manipulation devices were used when standard gripping is not sufficient in itself.
5. Non-contact sensing was used when the part could not be disturbed.
6. Sensors were used to determine if grasping and manipulation occurred correctly. If not, then the gripper and manipulation path would be adjusted and the sequence tried again.
7. Use a unique trajectory for manipulating parts made of different fabrics. This helps to account for the mechanical property variations from fabric to fabric.

2.3.3 Prototype Conceptualization and Design

An effective method for the structured design of a complex process or a component is called the functional analysis method. This method and brainstorming was used by the research team to generate design alternatives. In Tables 1 and 2 the results of some of the brainstorming sessions are given. Some of these ideas were immediately chosen over others. The important point is that any idea, no matter how ridiculous, should be considered.

The primary manipulation device currently in the workcell is a GMF six-jointed anthropomorphic arm. The flexibility in motion and the re-programmability of the robot make it ideal for conducting this research effort. This allows different paths and manipulation sequences to be tried before any commitment is made to dedicated hardware. A similar, but less costly, robot may be used in the final prototype system, or the necessary manipulations in the final prototype may be performed by dedicated automation devices.

TASK	PERFORMANCE CRITERIA	PROCESS OPTIONS	DESIGN ALTERNATIVES
PRESENTATION TO AUTOMATED SYSTEM	Adapted well for automated handling, simplicity, minimize the number of manipulations required to move to seam alignment station	Unit Presentation >waist-end down >waist-end up >horizontal	>Pants initially loaded by operator onto seam alignment device >Pants loaded by operator onto a simple hanger that is compatible with the alignment device
SEAM DETECTION	High repeatability, Accuracy within + or - 1/16 inch	>Selvage Detection & Calculation	>Vision -light transmissibility >Mechanically -2 ply thickness detection
		>Direct Seam Detection	>Vision -ultraviolet reflective thread -light transmission through seam -seam shadow -use pre-drilled holes or notches as markers >Mechanically -seam detection mech. using selvage fold back & triple thickness detection >Other -proximity sensor and metal thread
SEAM ALIGNMENT	High Repeatability, No damage to fabric	>Cuff-end	>Rollers inserted into end of leg
		>Waist-end	>Rollers inserted into waist end of leg >Long rollers thru leg >Slide plies relative each other other externally

Table 1: Functional Analysis Worksheet (page 1)

TASK	PERFORMANCE CRITERIA	PROCESS OPTIONS	DESIGN ALTERNATIVES
TRANSFER TO PRESS	Adapted well for automated handling, simple path, must not cause wrinkles, must maintain seam alignment	>Transfer pants to horizontal press >Transfer pants to vertical press	>Grab roller alignment devices at each end of leg and remove at press >Multiple-handed gripper >Temporarily tack seam and use a simple gripper
WRINKLE DETECTION AND ELIMINATION	Reliable, accurate, must not disturb fabric	>Non-contact	>Air-jet smoothing >Light-line distortion det. >Wrinkle shadow detection
		>Contact	>Rotary brush >Multi-fingered hand >Tactile probe sensor
		>Use a highly reliable transfer device that presents pants to press with acceptable wrinkles	>Multiple handed gripper with tensioning and some smoothing capabilities

Table 2: Functional Analysis Worksheet (page 2)

A description of each of the prototypes follows.

Internal Seam Alignment Device

The internal seam alignment device was devised as a solution to automating the alignment of the seams prior to pressing. This alignment should be seam-on-seam at the cuff end of the leg as well as at some specified point above the knee (above this specified point the seams begin to diverge). The exact point of divergence is determined by the style and design of the trousers. The tolerance on the alignment is one eighth inch.

This device uses two cylindrical rollers that are inserted into the cuff end of a trouser leg (see figure 24 and Appendix A). One of the rollers is actuated by a stepper motor, the other is a follower (or idler). The idling roller is supported by a bearing block that includes a set of linear bearings mounted perpendicular to the roller shaft. The linear bearings allow the idling roller to move with respect to the stationary roller while maintaining the same orientation. This linear degree of freedom allows the device to accommodate trousers with cuff-end widths of up to 12 inches.

The slide is actuated by a pneumatic cylinder. The air supply to the cylinder must be controlled to avoid damaging the fabric or seam threads. This is accomplished by a pressure regulator and flow controls. An electric solenoid valve in the air line is interfaced with a digital output port in the robot. This valve can be switched by a command from the teach pendant or from a robot program.

The stepper motor is controlled by software and a control card installed in the host computer. The resolution of the motor is 200 steps per revolution. Since the roller diameter is approximately 1.025 inches, and the motor directly drives the roller shaft, the linear resolution is .016 inch. Therefore, the accuracy of seam alignment is limited to one sixty-fourth of an inch with the current set-up.

External Seam Alignment Device

A drawing of the external seam alignment device is shown in Appendix B. This device was prototyped based on the discovery that two adjacent fabric plies can be made to slide relative to each other by using a frictional gripper on the outside of each ply (see figure 25). The most important criterion that must be met is that the coefficient of friction between the grippers and the fabric be greater than the coefficient of friction between the two fabric plies. An efficient method of moving the fabric plies was found to be rubber coated rollers. The rollers have to be driven in the same rotational direction and at the same speed if no net motion of the fabric is to be experienced. The rollers should be driven with a single actuator so that the rotational speed and the starting and stopping of the rollers will be synchronized. If each roller was driven independently a control system would have to be developed to ensure synchronization. A motor with a double ended shaft is ideal as a single actuator, since no gearboxes are necessary, to split the rotary motion and drive each roller.

This device could be used in conjunction with the internal seam alignment device. The internal seam alignment device works well at the cuff end of the trouser leg, but there are two reasons why it would not work well at the waist end. First, the rollers would be difficult to insert into the waist end and past pockets without human aid. Second, the shape of the trouser leg is diverging from the cuff end to the waist end. This could cause the trouser leg to walk off of the rollers at the waist end of the leg. These problems can be avoided by moving the seams relative to each other using a method which is external to the trouser leg. There would have to be one of these external seam alignment devices on each side of the trouser leg to move the seams and maintain the stability of the trouser leg (see figure 26).

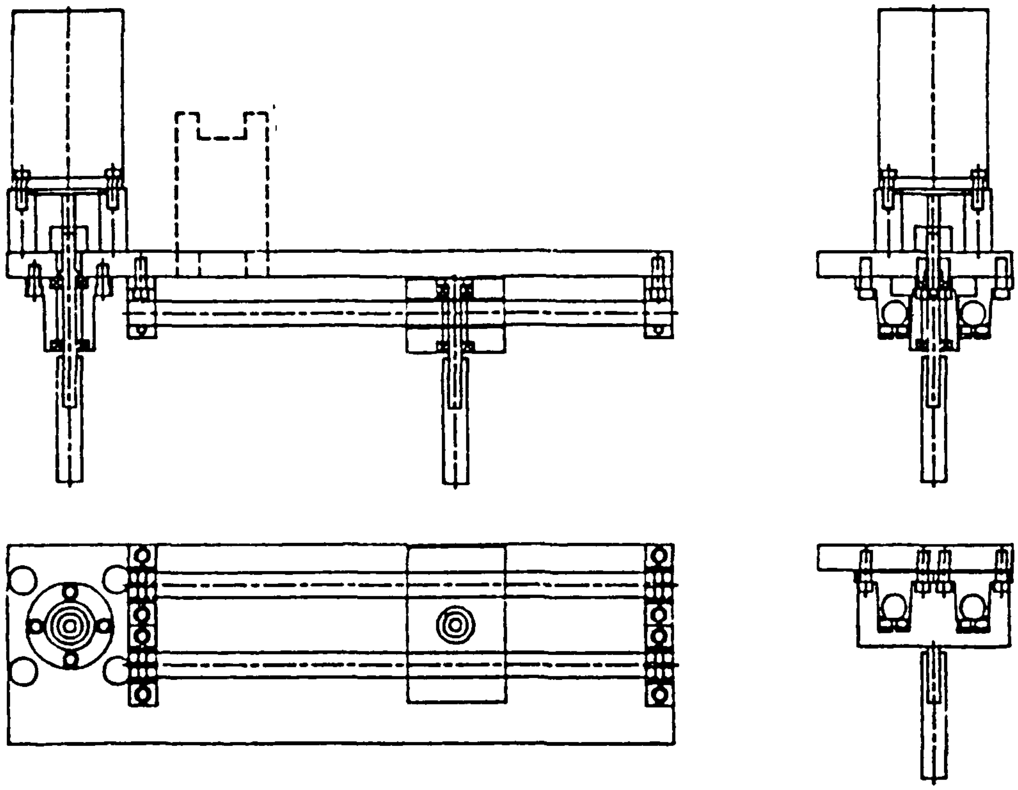


Figure 24 Internal Seam Alignment Device

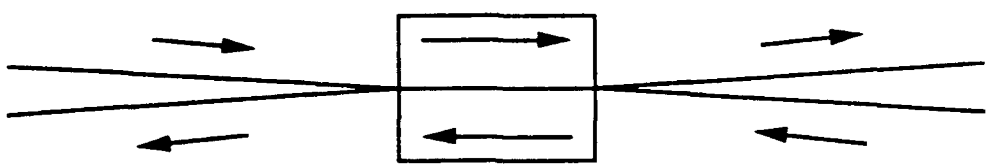


Figure 25: Relative Sliding of Adjacent Fabric Plies

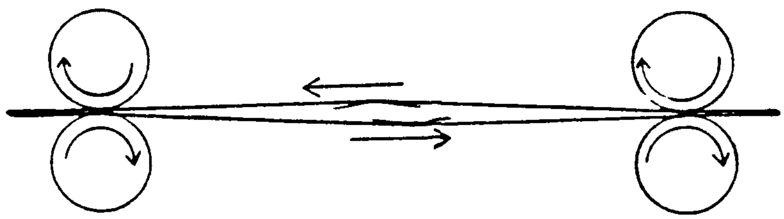


Figure 26 External Seam Alignment Device Concept

The external seam alignment device can also be used as a mechanism with which to tension the trouser leg laterally (perpendicular to the seam orientation). If the rollers are restricted from rotating then a linear force can be transferred through them to the trouser leg. As will be explained later, a lateral tensioning force is required near the waist end of the leg in order for the transfer gripper to grasp the trousers without inducing wrinkles.

Mechanical Seam Detection Mechanism

The mechanical seam detection mechanism is shown in figure 27 and in Appendix C. This prototype was developed as a low cost and reliable method of locating a seam. The change of the number of fabric plies at the seam is used with a thickness measuring scheme to identify the location of the seam.

The cuff end of the leg is loaded into the device and then is moved perpendicular to the seam orientation. When the seam approaches the detection apparatus the selvage is caught by one of the two selvage catches, depending on the direction of motion (see figure 28). The caught selvage is flipped on top of the selvage on the other side of the seam (see figure 29). A momentary contact switch is activated by the three fabric plies as they pass between the base and the switch roller.

An earlier version of this device did not have selvage catches on the base. In this case, the double thickness of fabric at the seam was going to be measured and the position of the seam was going to be found indirectly be assuming that the seam was one-half of the distance between the selvage beginning and ending points. Tests showed that this method was inconsistent since the selvage would randomly catch and be flipped by the base. The selvage catches were employed to consistently catch and flip the selvage and the measurement scheme was changed. Instead of measuring where the selvage begins and ends, the microswitch is activated exactly where the seam is located (see figure 30). This measurement scheme is direct because it requires no additional calculations to locate the seam.

Theoretically, the seam detection mechanism would have to be adjusted for different thicknesses of fabric. However, in most cases adjustment is not needed. Since three fabric plies are used to activate the switch, instead of two, the device is less sensitive to changes in fabric thickness. This was demonstrated by experiment and is presented in the experimental results section.

Transfer Gripper

The transfer gripper is a two-handed single degree of freedom end-effector that mounts on the tool face plate of the robot (see figure 31).

The left hand gripper is a pneumatically actuated four bar linkage (see figure 32). The linkage allows the gripper to occupy less space when it is open compared to other kinematic linkages. The motion of the gripper fingers resembles parallelogram motion, since the links are close in dimension.

The right hand gripper is a scissors type of gripper, since it has a single revolute joint (see figure 33). This gripper is also pneumatically actuated. A set of linear bearings allows this gripper to move relative to the left hand gripper. The single degree of freedom provided by the slide serves two purposes. First, the trousers can be tensioned in the longitudinal direction (parallel to the seam orientation). The tension keeps wrinkles from being introduced into the trouser leg during the transfer process. Second, the slide is long enough so that the right hand gripper can be entirely removed from the press buck. Thus, while the pressing cycle is running the left hand gripper can remain grasping the waist end of the trouser leg near the crotch while

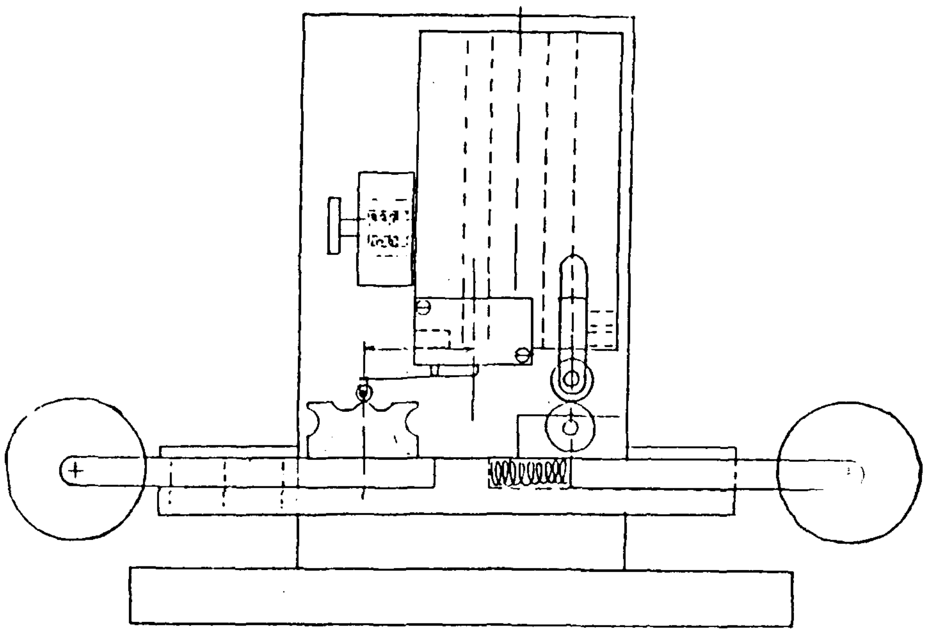


Figure 27 Mechanical Seam Detection Mechanism

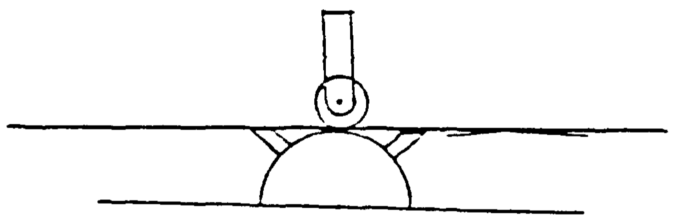


Figure 28 Seam Approaching Selvage Catch

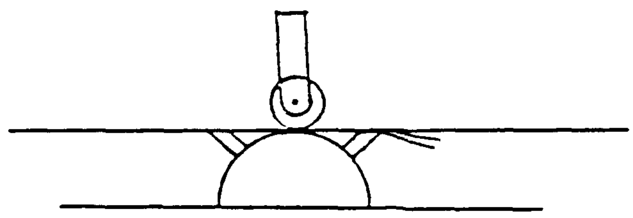


Figure 29 Seam Folded Back by Selvage Catch

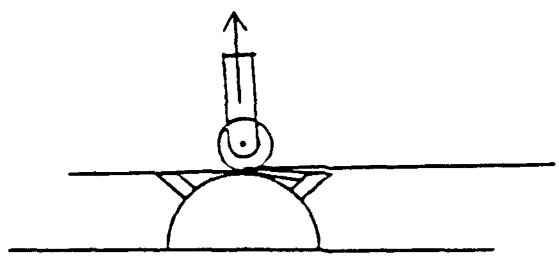


Figure 30 Switch Activated at Seam Location

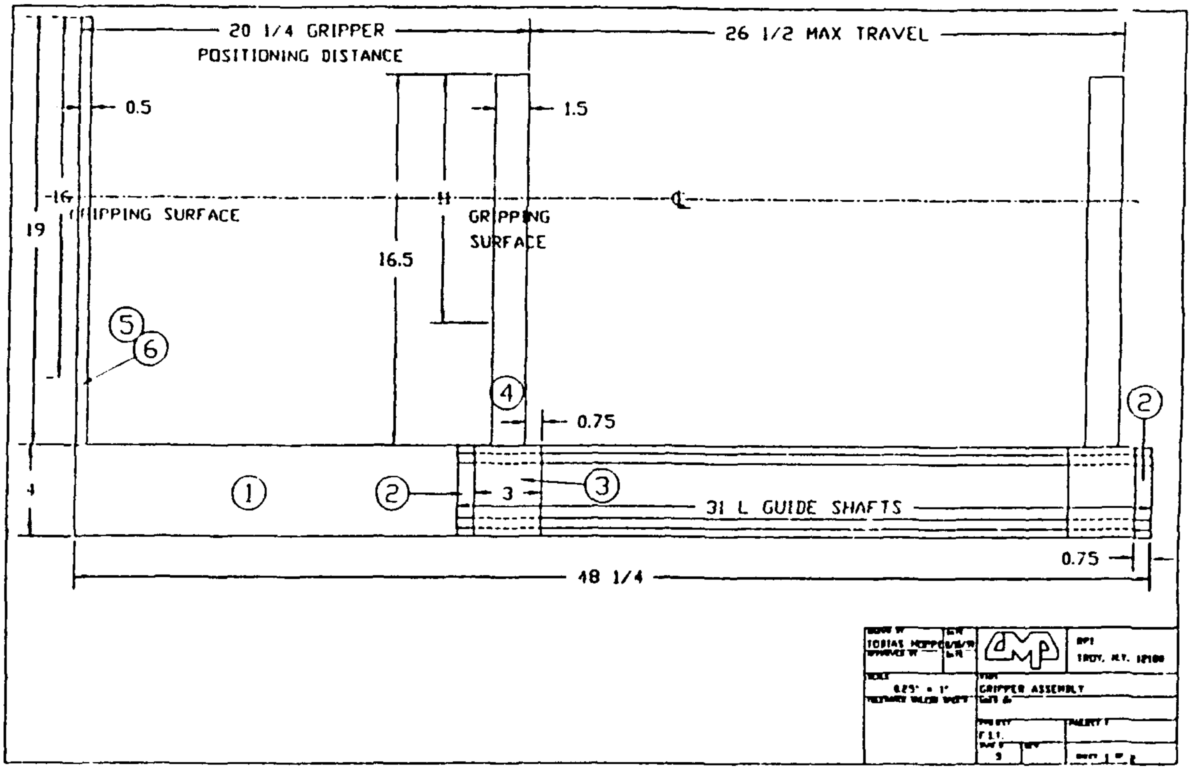


Figure 31 Transfer Gripper

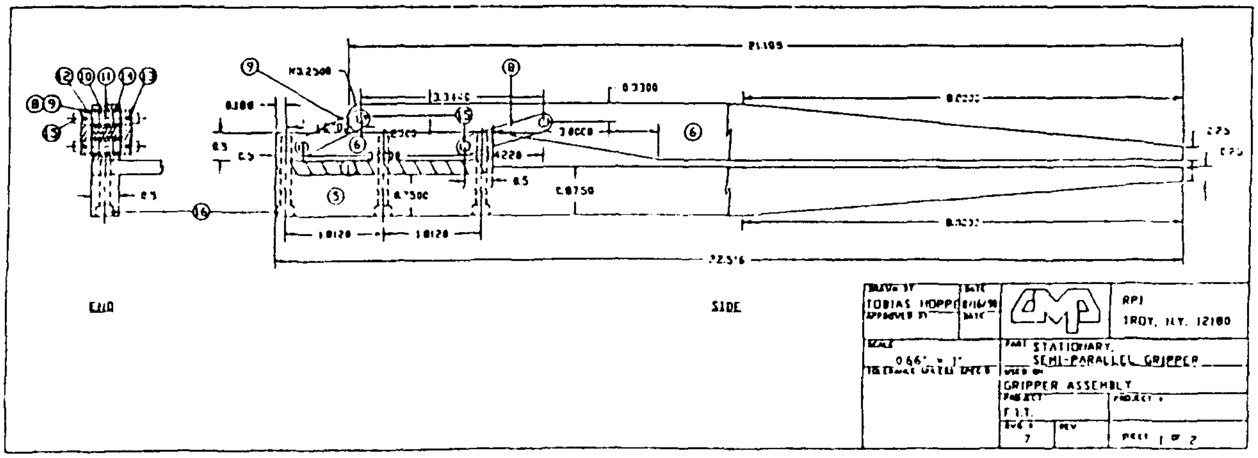


Figure 32 Left Hand of Transfer Gripper

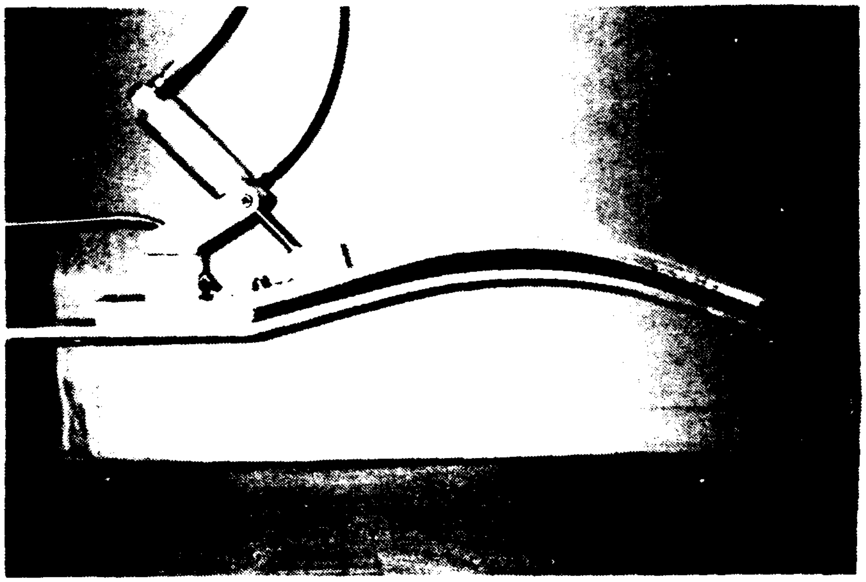


Figure 33: Right Hand of Transfer Gripper

the right hand gripper is removed from the press buck. Both grippers have adhesive-backed foam rubber tape at the interface between the gripper and the fabric.

The sliding motion is actuated by a DC servomotor and a ball screw assembly. A servomotor control card and software installed in a PC allow the slide to be accurately positioned. Limit switches mounted on the gripper base indicate the end of motion limits of the slide. A third limit switch provides a home position signal to the motor control card. This is needed because a relative encoder is used on the servomotor.

The air flow to the grippers is controlled by two electric solenoid valves interfaced with the digital output ports of the robot controller. In addition, the air flow rate to the left hand gripper is restricted by port mounted flow controls on the pneumatic cylinder. The air flow rate to the left hand gripper must be restricted to avoid damaging inertial forces while opening the fingers.

2.3.4 Discussion of Testing and Results

The experiments were conducted using three trousers of different fabrics, waist sizes, and styles. The fabric and garment specifications for the selected trousers are given in Table 3. A variety of fabric types and garment sizes were chosen for several reasons. First, it shows how a particular device's accuracy and repeatability does or does not vary between different fabrics and garment sizes. This will help to define the useful working range of the device. Second, it will indicate the adjustments and changes that may have to be made to the device in order for it to work successfully over a sufficient range of materials.

For the tests that required distance measurements, a machinists scale with one thirty-second of an inch graduations was used. This allows for a theoretical measurement resolution of one sixty-fourth of an inch on either side of the measured value.

Internal Seam Alignment Device

The internal seam alignment device was tested for accuracy, repeatability, and trouser leg walking. The accuracy test was used to determine if the seams moved the amount they were expected to move based on knowing the number of steps the motor rotated.

The repeatability test determined whether the seam would return to its starting position if the motor was commanded to rotate clockwise 300 steps and then counter-clockwise 300 steps. The repeatability is perhaps the most important performance parameter of this device. In most mechanical devices, if the accuracy is less than required it will most likely be an amount that remains unchanged or slowly degrades as the device wears. Thus, this constant amount can be accounted for in the design of other components in the device or in the control system. However, if the accuracy of the device varies randomly (its repeatability is low) then the accuracy offset cannot be corrected since its value is not known.

The walking of the trousers as they are suspended from the internal seam alignment device was also investigated. The trousers have a tendency to move down the rollers as the rollers are rotating due to the effect of gravity. The phenomena of walking should be differentiated from slippage.

Consider the situation shown in figure 34. If the rollers are in a vertical orientation the material will experience a gravitational force proportional to its weight in the downward direction. If the material is rigid it experiences little deformation for a given load and the only way the material can move relative to the rollers is by slipping. If the material is flexible, as is

FABRIC NUMBER	I.D. #	FABRIC DESCRIPTION		GARMENT DESCRIPTION			CUFF-END WIDTH (inches)	FABRIC THICKNESS (inches)
		GENERAL	CONTENT	WAIST SIZE	INSEAM LENGTH	CUFF		
1	3	Texfi Blends Style: 53326 Zenith Linear Yield: 10-10.5 oz.	15/1 blend of 65% Fortrel, 35% Rayon	42	37	NO	9 1/2	.016
2	11	Biltmore Textile Style: 27498 Weight: 6 oz./ square yard	100% Cotton	50	37	NO	11 1/2	.018
3	8	Windsor Textile Style: Brown Twill Weight: 11.5 - 12 oz.	100% Wool	28	37	NO	9	.020

Table 3: Fabric and Garment Specifications

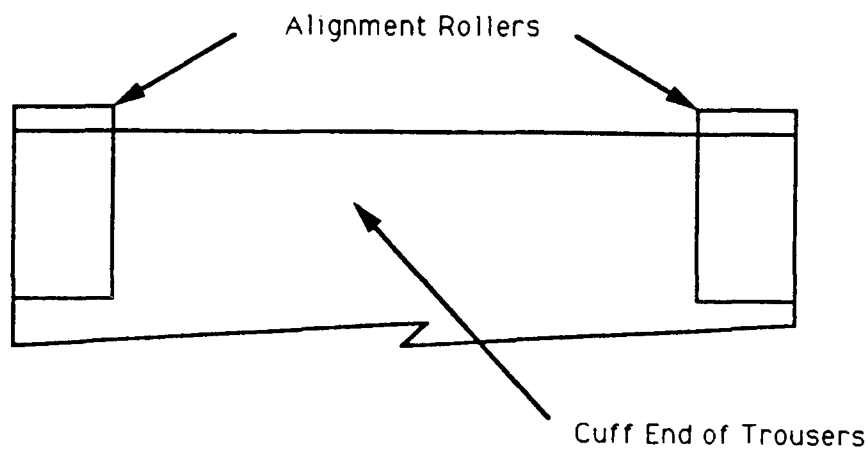


Figure 34: A Rigid Material on Vertical Rollers

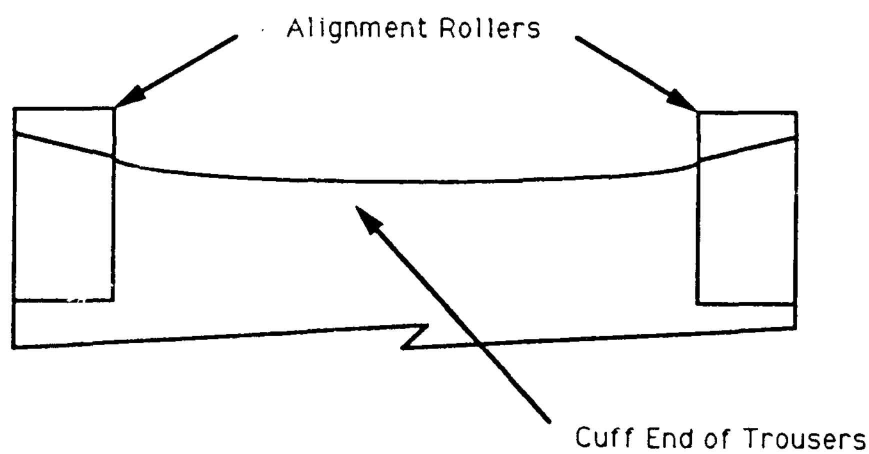


Figure 35: A Flexible Material on Vertical Rollers

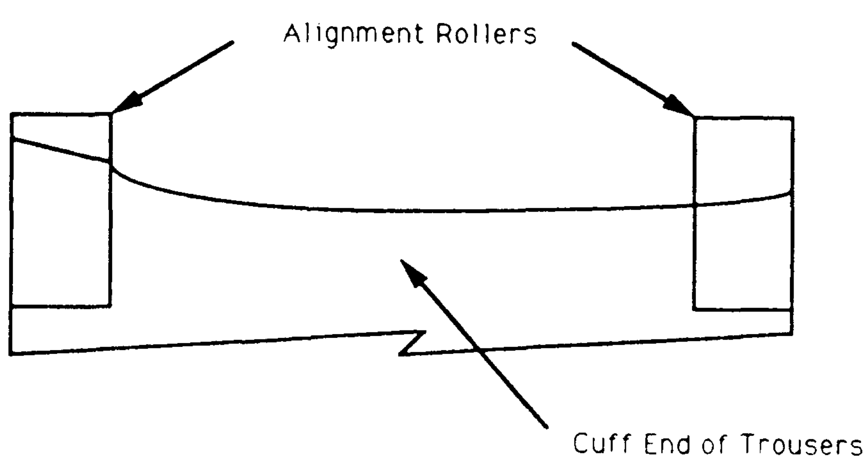


Figure 36: Flexible Material Walking Phenomena

fabric, then the vertical force can deflect the material in between the rollers (figure 35). When this deflected portion reaches the roller it is simply maintained at its same vertical position (figure 36). As rotation continues the material proceeds to walk down the roller and will eventually leave the roller.

The results of the accuracy, repeatability, and walking tests for the internal seam alignment device are given in Tables 5, 6, and 7, respectively. The results show that this is an acceptable solution to the seam alignment problem. Even though the accuracy is not a good as expected, the repeatability is very high. The accuracy of the movement seemed to be dependent on the amount of walking of the leg (see Tables 5 and 7). The walking of the trouser leg can be controlled by a finger oriented in such a way as to force the fabric upward as the trousers move.

External Seam Alignment Device

No formal testing was conducted with the external seam alignment device. However, the prototype was tested informally by observing the response of several fabrics as the rubber rollers were manually actuated. Several useful observations were made as a result of this testing.

The external seam alignment prototype was tested on a variety of fabrics with both polyurethane rollers and rubber rollers. Both of these interfacial materials provided excellent frictional gripping of the fabrics. The clamping force that holds the rollers against the fabric was provided by a spring and was very small (about a pound). Even with this small normal force the frictional force was great enough so that no fabric slippage was experienced between the rollers and the fabric.

The informal testing done on the external seam alignment device has shown promising results. The prototypes should be further developed to allow pneumatic opening and closing and sliding of the two device for lateral tensioning of the leg. Pneumatic cylinders would provide a cost effective and simple method of closely controlling the clamping force of the rollers. It is necessary to control this force, so a variety of fabrics and their associated frictional characteristics can be accommodated.

Mechanical Seam Detection Mechanism

The mechanical seam detection prototype was tested with the four different fabrics given in Table 4. The trousers were manually moved through the mechanism.

The results of the mechanical seam detection experiments are given in Table 8. The seam detection mechanism has a 94% success rate in reliably detecting the seam. The probable sources of most of the failures were identified during testing. First, as with the transfer gripper, it is important that proper lateral tension be utilized. If the lateral tension in the fabric was not sufficient the selvage would ride over the selvage catch and not be flipped. This caused the switch to be activated before the seam reached it. If the detection mechanism was an integral part of the internal seam alignment device then the tension would be consistent and controllable. Second, the quality of the switch should be improved. It was noted in several cases that the switch got bound in its "on" position by small movements of the fabric parallel to the switches roller axis.

Transfer Gripper

The transfer gripper was tested with three different trousers since it was expected that the performance of this device would be the most affected by changes in the fabric or garment

FABRIC	THICKNESS (inches)	CUFF	SELVAGE WIDTH (inches)	BLEND
A	.012	YES	7/8	POLY/WOOL
B	.012	NO	15/16	POLY/WOOL
C	.014	NO	15/16	POLY/WOOL
D	.021	NO	1	100% WOOL

Table 4: Fabr[illegible] Specifications for the Mechanical Seam Det[illegible]ion Experiments

FABRIC NUMBER	TEST RUN	BEFORE MOVEMENT Measurement to datum (in.)	AFTER MOVEMENT Measurement to datum (in.)	Actual distance moved (inches)	Difference between actual and theoretical
	Run 1	10.72	5.75	4.97	.08
FABRIC 1	Run 2	10.72	5.78	4.94	.05
	Run 3	10.72	5.75	4.97	.08
	Run 1	9.88	4.94	4.94	.02
FABRIC 2	Run 2	9.88	4.94	4.94	.02
	Run 3	9.88	4.91	4.97	.05

Table 5: Results of Internal Seam Alignment Device Accuracy Tests

FABRIC NUMBER	TEST RUN	BEFORE MOVEMENT	AFTER MOVEMENT	Difference between measurements (inches)
		Measurement to datum (in.)	Measurement to datum (in.)	
FABRIC 1	Run 1	10.50	10.47	.03
	Run 2	10.47	10.47	.00
	Run 3	10.47	10.44	.03
FABRIC 2	Run 1	9.88	9.88	.00
	Run 2	9.88	9.88	.00
	Run 3	9.88	9.88	.00

Table 6: Results of Internal Seam Alignment Device Repeatability Tests

FABRIC	BEFORE MOVEMENT	AFTER MOVEMENT	NUMBER OF STEPS TRAVELED/ INCHES TRAVELED	DISTANCE WALKED (inches)
	MEASUREMENT FROM MARKER TO DATUM(in.)	MEASUREMENT FROM MARKER TO DATUM(in)		
1	1.00	1.78	1270/ 20.50	.78
2	1.25	1.75	1450/ 23.35	.50

Table 7: Results of Internal Seam Alignment Device Walking Tests

FABRIC	NUMBER OF TEST RUNS	NUMBER OF SUCCESSFUL DETECTIONS	NUMBER OF FAILURES
A	40	36	4
B	45	42	3
C	56	55	1
D	43	40	3
TOTALS	184	173	11

Table 8: Results of Mechanical Seam Detection Mechanism Tests

FABRIC NUMBER	AVERAGE SEAM MOVEMENT	
	CUFF END (INCHES)	WAIST END (INCHES)
1	.07	.03
2	.12	.04
3	.12	.04

Table 9 : Average Seam Movement during Pants Transfer

specifications. A stationary reference, or datum, was used to determine the relative locations ofthe seams both before and after the transfer of the trousers to the press. In some cases the seams were not perfectly aligned before transfer. This is inconsequential since the objective of the experimental was to determine if the seam alignment changed during the grasping and transfer process.

A small amount of tension was applied to the trousers immediately after they were grasped and before transfer to the press started. The tension was applied by moving the right-hand gripper an additional .30 inch after the grippers were closed. This kept wrinkles from forming in the trouser leg during transfer.

The average amount of seam movement that occurred for each different fabric is given in Table 9. It should be noted that none of the trousers used in testing had previously been pressed. The tests on the transfer gripper also showed some additional development work is needed. The seam alignment is not maintained to a sufficient degree of accuracy during the transfer process. It should be noted that the largest change in seam alignment occurs at the cuff end of the trouser leg (see Table 9). Observations during testing indicate that the source of the problem may be in the design of the gripper mechanism. The distortion in the leg as the gripper is closed seems to be caused by the fact that the side of the leg closest to the gripper is experiencing clamping forces before the side of the leg farthest from the gripper. This is due to the nature of the scissors type clamping geometry. A semi-parallel gripper like is used at the waist area may be a solution.

It was also noticed during the course of testing that the lateral tension (perpendicular to the seam orientation) on the leg was a critical factor in determining if the leg could be laid on the press relatively wrinkle free. If this tension was not sufficient or incorrectly oriented the fullness in the leg near the waist area would bunch up as the gripper closed.

The typical transfer process starts with grasping the trouser leg on the seam alignment device as shown in figure 37. Then longitudinal tension is applied with the servomotor and the cuff end of the leg is released from the seam alignment device. The robot then guides the gripper to the press. Note that in figure 39, the extreme cuff end of the leg is free to droop over the right hand gripper. The path of the robot has been programmed to aid in reducing wrinkles caused by this dangling end. This is done by positioning the gripper at a position farther to the right of its releasing point and then moving the leg to the left as it descends to the buck (see figure 40). The typical cotton trouser leg after transfer is shown in figure 43. A sequence of the transfer is shown in figures 37 through 42.

During the testing procedures no single wrinkle could be identified as being too large that it would be pressed into the leg. All of the wrinkles characterized were located in the top ply of the leg. Wrinkles in the bottom ply are almost certain to be pressed into the leg. Further research should be conducted to determine what sizes and number of wrinkles are acceptable in various fabrics, before pressing is initiated, and not be pressed in.

Based on the quality of the transfer process it is believed that more than two grippers are not needed to grasp and manipulate the leg. The longitudinal tension applied to the leg kept the fabric from drooping in between the grippers. However, it may be advantageous to provide some means of support to the extreme cuff end of the leg.

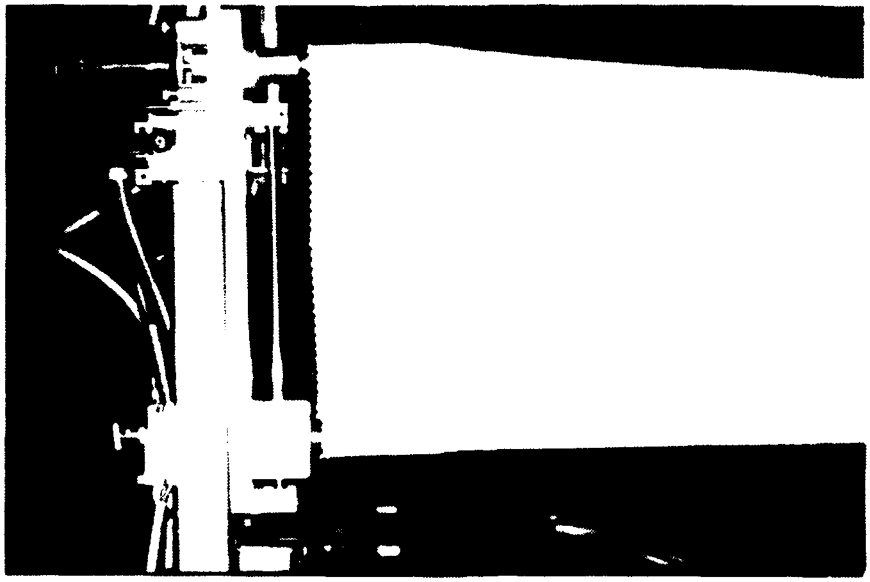


Figure 37: Grasping the Pant Leg at the Seam Alignment Station

Figure 38: Pants Beginning the Transfer Process

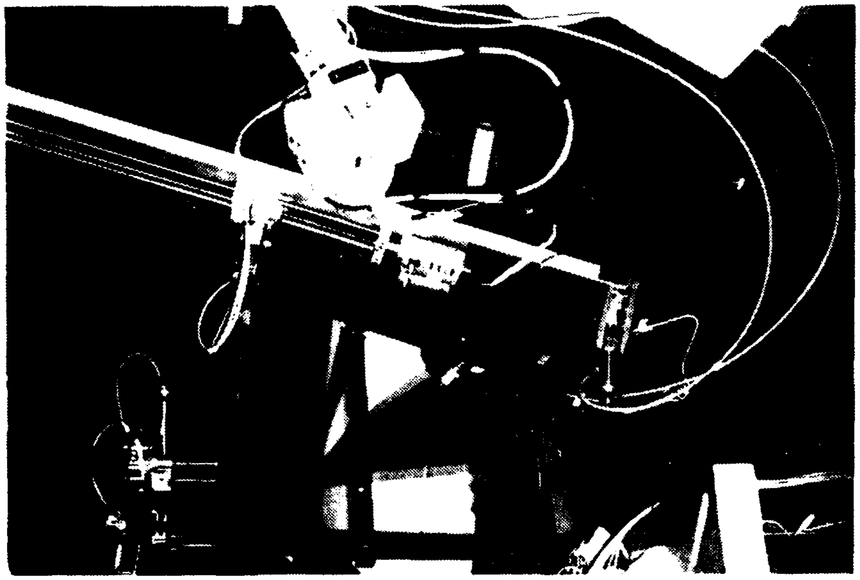


Figure 39: Pants Approaching Press

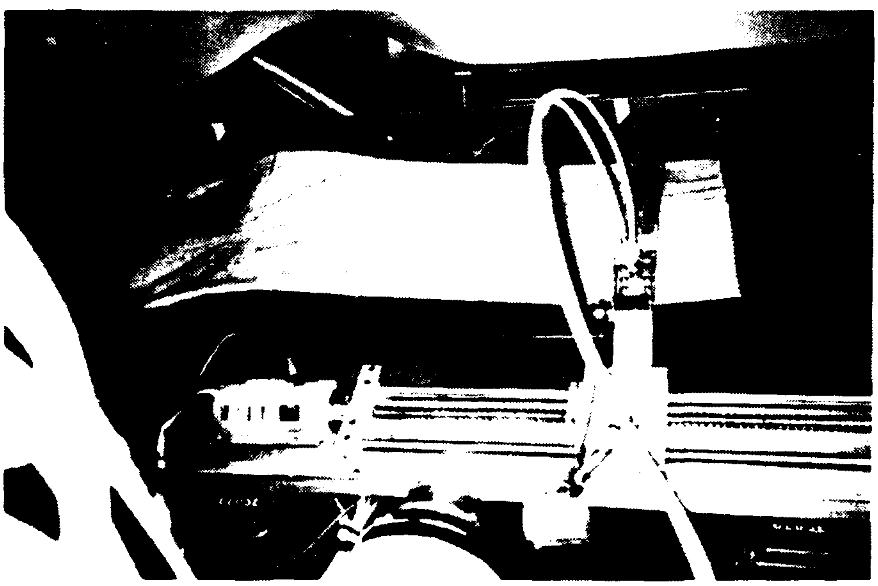


Figure 40: Pants Laid on Press Buck

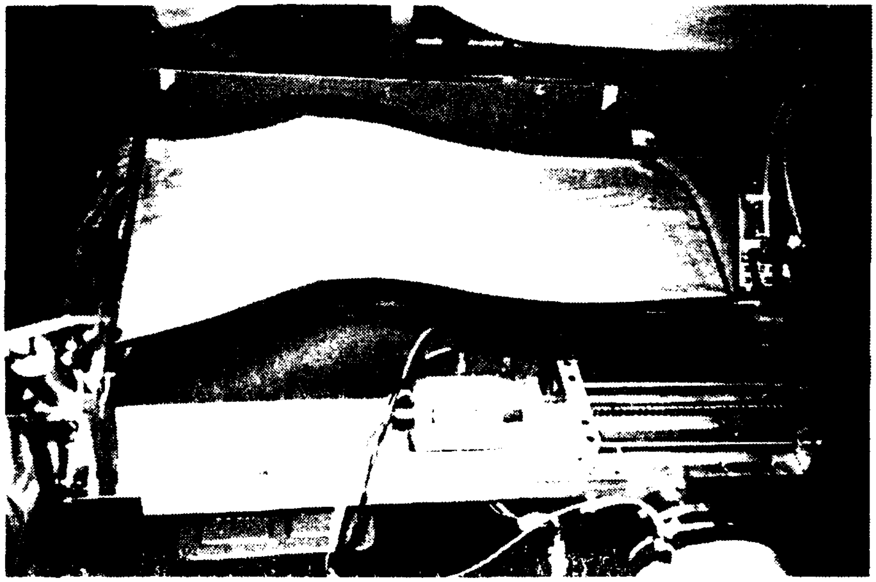


Figure 41: Pant on Buck with the Tensioning Bar Engaged

Figure [illegible]: Pants on Buck with the Right Hand Gripper Removed

This Page [illegible]

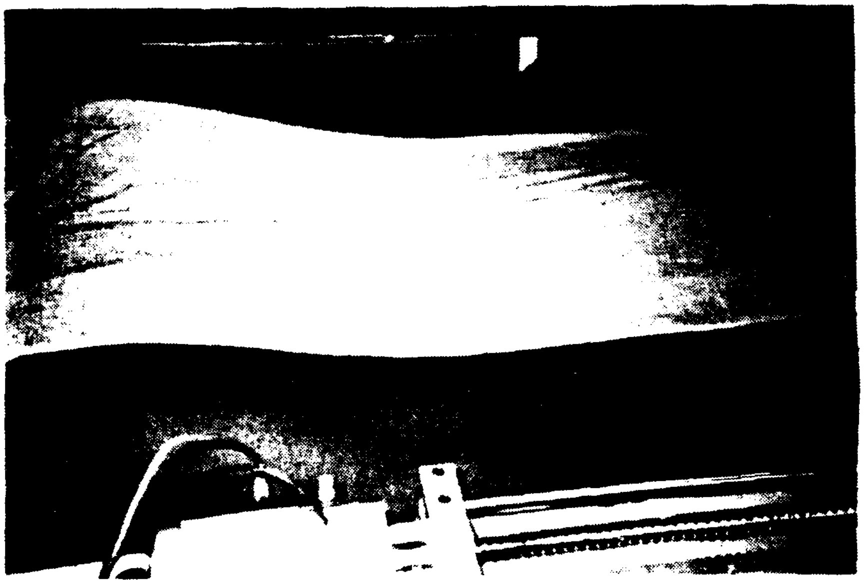


Figure 43: Typical Cotton Pants Lay-up

2.4 INTEGRATION

2.4.1 Task Planning and High Level Control Software Development

2.4.1.1 Petri Net Task Planning

This section presents the ideas of task sequence planning for a garment handling workcell. The search of the sequences is based on a Petri net model of the system. Given the AND-OR net which describes the geometric relationships among all objects in the system, we could map the AND-OR net into a Petri net. Then we search the Petri net using some efficient algorithms. All possible sequences will be extracted when the reachability tree of the Petri net has been developed to a certain depth. The sequences will only depend on the initial states and final states and possibly some intermediate states. A simulation example of the sequences has been proposed and analyzed.

Introduction

This report discusses an automated garment handling system that performs the task of finish pressing of a garment, specifically trousers, automatically under the control of a host computer. The original motivation to start this research was due to the heavy labor and high cost of the finish pressing phase in garment manufacturing. Pressing needs much patience as well as some specific techniques. When the pressed garment is considered an unsatisfactory product, e.g., the seam alignment which is generated during pressing is not correct, additional work will have to be done. Therefore, if pressing could be automated and the garment handling system could be implemented, the number of errors and the amount of rework will be reduced significantly.

Presently, the finish pressing phase of garment manufacturing requires a press, possibly with some auxiliary mechanisms, which is run by an operator. The operator picks up one pair of trousers from the waiting pile, aligns the seams of one leg, holds the trousers in the appropriate orientation, loads the press, smooths out any wrinkles while at the same time checking the seam alignment, and then starts the pressing cycle. This procedure is then repeated for the other leg. Note that in many of the discussions that follow, the pressing of one trouser leg is addressed while in the actual implementation both legs would have to be pressed. Obviously, the pressing operation is done by a human. We would like to introduce a robot to accomplish the same task. The robot will be controlled by a host computer. The seam alignment would be performed by a seam alignment device. The garment waiting to enter the system will be on a garment handler. Therefore, the main objects in the system will include a robot or other mechanical handling device, garment, garment handler, seam alignment device, press, sensor and host computer. The objects in the automated system could be presented as shown in figure 44.

There is a fixed knowledge base existing in the system. Before the robot starts to move, the computer obtains the initial states and the final states. It will generate all possible sequences of operations. Then a certain sequence will be chosen based on some selected parameters. Each operation which is included in this specific sequence will be considered as a sub-goal at that time. For each sub-goal, the lower level planning work will be done. A lower level command sequence corresponding to each high level operation will be generated, each lower level command sequence will be fired and each command could directly be used to control some mechanisms such as the robot, handler, press, sensor and so on. During the working of the system, the new information might be fed back to the computer and the control commands may be updated to make the whole system coordinate.

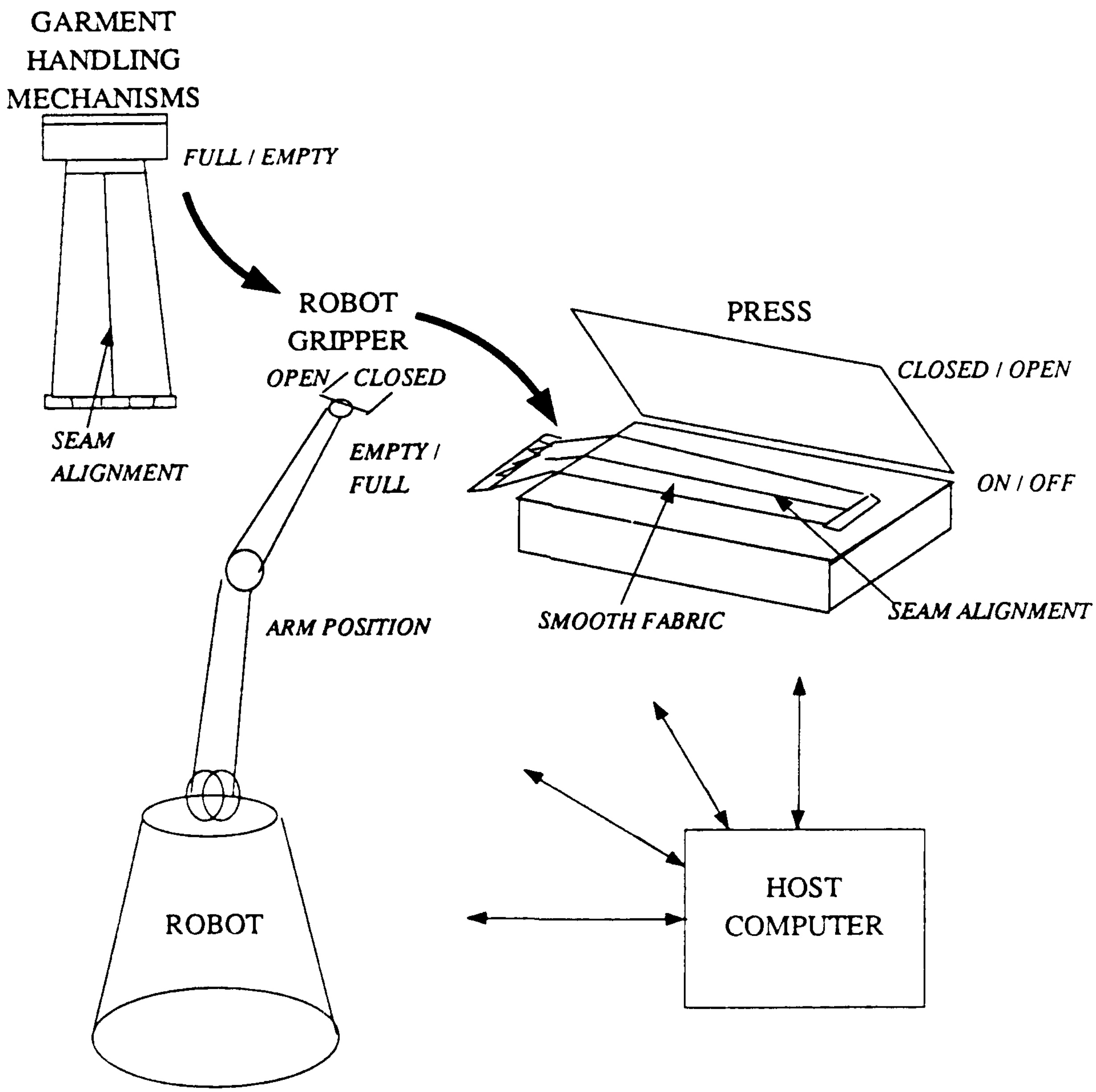


Figure 44

AUTOMATED GARMENT HANDLING
SYSTEM STATES

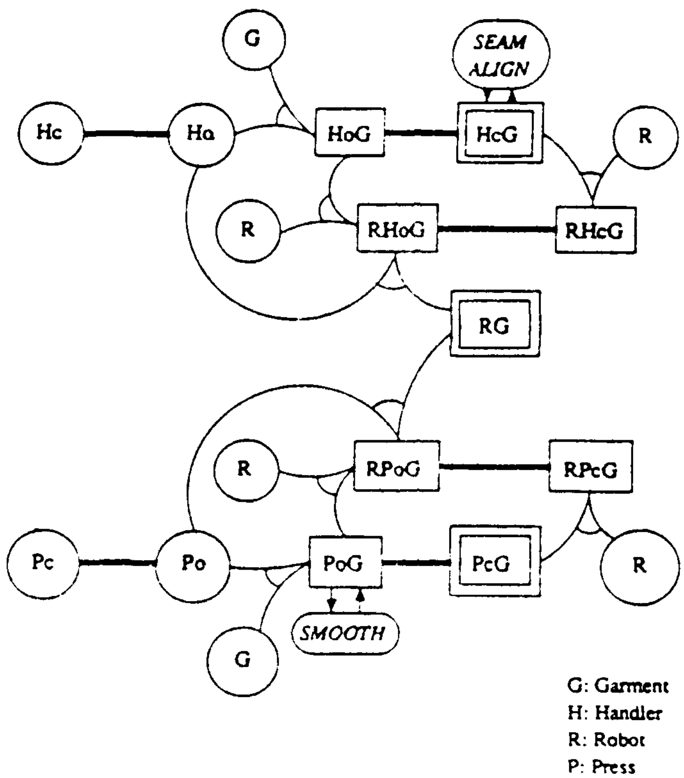


Figure 45 AND-OR Net State Diagram

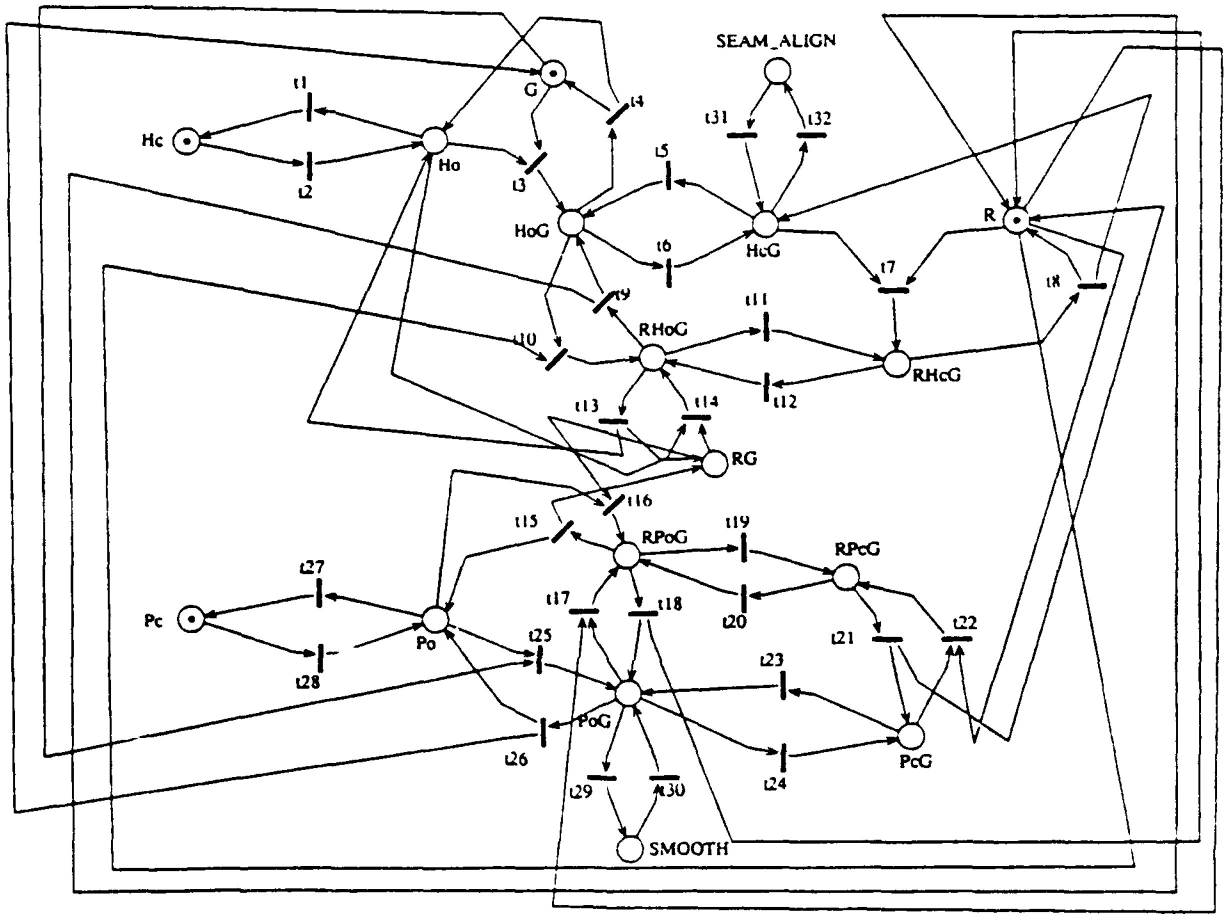


Figure 46 The Petri Net Representation for Garment Handling Sys.

Background

The whole robotic system as well as its environment is in the initial state before the system starts working. The intelligent robot should have the ability to analyze the information it has from the outside world such as the descriptions of the current or initial states, the knowledge base that represents the set of conditional statements which defines the knowledge in the problem domain, and also the final states the system is expected to reach, and then reason so that it could automatically obtain all possible operation procedures to reach the final states.

Task level planning[33] is the most important aspect and the highest level planning in the whole robotic planning system. Many planning techniques and systems have been developed[35,36,37]. Most of these planning methods use search techniques developed in the artificial intelligence area. The aim of task level planning is to efficiently generate all possible task sequences and to be able to distinguish the different efficiencies among them. Some sequences may contain the fewest number of steps of operations. Actually, each different kind of operation may cost different amounts of time or different amounts of energy. Some sequences may also contain parallel operations to save total time needed to fulfill the task. Therefore, for the most economic consideration for the generation of task sequences, we should find the minimum-cost sequence.

Petri net representation of the system

The Petri net has been widely used in modeling manufacturing systems, expert systems, robot assembly planning systems as well as other kinds of engineering applications[38-44]. This is an efficient abstract and formal information flow model. The Petri net is characterized by its flexibility and efficiency in modeling and analysis of complex discrete-event systems. The definition and some relating terminologies are shown as follows:

Definition 1 *A Petri net structure, N:* is a four-tuple, $N = (P, T, \alpha, \beta)$. $P=\{p_1, p_2, \ldots, p_n\}$ is a finite set of places, $n \geq 0$. $T=\{t_1, t_2, \ldots, t_m\}$ is a finite set of transitions, $m \geq 0$. $P \cap T = \emptyset$. $\alpha \subseteq \{P^*T\}$ and $\beta \subseteq \{T^*P\}$ are sets of directed arcs.

Definition 2 *Marking μ:* Marking μ of Petri net N is a mapping from set P to set $\Lambda = \{0, 1, 2, \ldots, L\}$ which is a finite set, i.e.,

$$\mu : P \rightarrow \Lambda,$$

where μ sets tokens to every place in N, $\mu_i = \mu(p_i) \in \Lambda$ indicates the number of tokens in place p_i. μ can be in the form:

$$\mu = (\mu_1, \mu_2, \ldots \mu_n)^T; \qquad \mu_i = \mu(p_i),\ p_i \in P$$

Definition 3 *Marked Petri net M:* A Petri net structure N containing a marking μ is a *marked Petri net* which is the following five-tuple,

$$M = (P, T, \alpha, \beta, \mu)$$

Sometimes, for the sake of simplicity, we refer to a *marked Petri net* as a *Petri net* as shown later in this paper.

Definition 4 *Petri net graph:* The Petri net graph consists of directed arcs and two kinds of nodes. In the graph, circle node and bar node represent place and transition respectively. A

Directed arc, which links a circle node and a bar node, indicates the relation between place and transition. Marking μ is shaped by solid dots in circle node.

One important characteristic of a Petri net is that it could represent serial and concurrent events and resource constraints. For this system, we use Generalized Stochastic Petri Nets(GSPN) software to represent the system and do some simulation work as well as verifying the task sequences.

Initial AND-OR Net Representation of the System

The brief overlook of the garment handling system has been presented in figure 44. The AND/OR net state diagram and the corresponding Petri net representation has been proposed in figure 45 and figure 46, respectively. In this section, we briefly describe how we get the AND/OR net representation. First, we propose the definition of AND/OR net as follows:

Definition 5 *Pair-match set Γ:* Given two finite sets $\Sigma_1 = \{ a_1, a_2, \ldots, a_m\}$, $\Sigma_2 = \{ b_1, b_2, \ldots, b_n\}$,

$$\Gamma(\Sigma_1, \Sigma_2) = \cup_{i=1 \text{ to } m} \cup_{j=1 \text{ to } n}\{\{ a_i, b_j\}\}$$

Definition 6 *An AND/OR net:* is a three-tuple (S, A, N) where S is a finite set of states ($s_1, s_2, \ldots, s_t$), $A \subseteq \Gamma(S, (\pi(S)-\emptyset))$, $N \subseteq \Gamma(S, S)$, and $A \cap N = \emptyset$.

The elements of S are called *nodes*, the elements of A are called *and-arcs*, and the elements of N are called *non-and-arcs*. The and-arc $\{ \lambda, \Psi\}$ is said to connect node λ to the nodes in ψ, $\psi \subseteq S$. The non-and-arc $\{ \lambda_1, \lambda_2\}$ is said to connect node λ_1 to the node λ_2.

The AND-OR net representation is derived from the original system states graph as shown in figure 1. We propose the definition of system states graph as follows:

Definition 7 *System states graph:* is a graph that presents all main objects in the real system, all possible visible states corresponding to each object, and the possible interactions among the objects.

From the system states graph, we could obtain all main objects. Each object could have an internal geometric state, like open or closed. Each object could also have a geometric relationship with other objects which may relate to it. For example, if the robot moves to the place of the handler to get the garment, it should relate to both garment and handler. Of course, the garment and handler should also relate to the robot at the same time. After the robot gripper grasps the garment, it will leave the handler and still keep its relationship with the garment until the time it ultimately leaves the garment on the press. For the robot itself, it will change its own geometric state when it contacts the outside world, i.e., the gripper will change from open to closed or closed to open from time to time. Using this way of considering all possible geometric states each object will possess and possible relationships among objects, we could obtain the AND/OR net state diagram. In this diagram, each single object is enclosed in the circle and each group of objects which are relating to each other are described as block. It is clearly that the seam align and smooth operations are done at the state HcG and PoG, respectively.

The network is indirect, i.e., the transition could happen in either direction. Therefore, the regular methods for extracting all possible sequences from AND/OR graph[45] or AND/OR tree representation of the system will not work in this case. A more efficient method should be explored to fulfill the task plans.

Mapping AND-OR Net to Petri Net

The Petri Net can represent events, conditions and the relations between them in a system. The transitions, the places and the arcs in a Petri net can respectively model events, conditions and relations.

As we mentioned in the last section, all transitions in the system AND/OR network could move in either direction. Based on this idea, we can split each transition into two directed transitions in the Petri net. All states which are either in the block shape or circle shape or ellipse shape in the state network now become places. One of the big differences between the AND-OR net and Petri net is that the same state in the AND-OR net could appear more than one time. For example, there are four robot circle state copies in the AND-OR net. However, there is only one place to represent the robot in the Petri net.

The algorithm for converting AND/OR net to Petri net is shown as follows:

Algorithm 1 *Mapping from AND/OR net to Petri net.*
Input: AND-OR net $N_A = (S, A, N)$
Output: Petri net $N_P = (P, T, \alpha, \beta)$

1. initialize $P = T = \alpha = \beta = \emptyset$, $n_P = n_T = 0$
2. for each set $n_i \in N$, $n_i = \{ n_{i1}, n_{i2}\}$
   begin
   - add 2 transitions t_{np+1}, t_{np+2}
     - $T = T \cup \{ t_{np+1}, t_{np+2}\}$;
   - $n_p = n_p + 2$
   - check whether n_{i1}, n_{i2} is in P,
     - if no, $P = P \cup \{ n_{i1}, n_{i2}\}$;
   - $\alpha = \alpha \cup \{(n_{i1}, t_{np+1}), (n_{i2}, t_{np+2})\}$;
   - $\beta = \beta \cup \{(t_{np+1}, n_{i2}), (t_{np+2}, n_{i1})\}$;

   end
3. for each set $a_i \in A$, $a_i = \{ a_\lambda, \psi_a\}$
   begin
   - add 2 transitions t_{np+1}, t_{np+2},
     - $T = T \cup \{ t_{np+1}, t_{np+2}\}$;
   - $n_p = n_p + 2$
   - for every $e_j \in \{\{ a_\lambda\} \cup \psi_a\}$ and $\{ e_j\} \cap P = \emptyset$
     - $P = P \cup \{ e_j\}$
   - $\alpha = \alpha \cup \cup_j\{(a_\lambda, t_{np+1}), (e_j, t_{np+2})\}$, for all e_j;
   - $\beta = \beta \cup \cup_j\{(t_{np+1}, a_\lambda), (t_{np+2}, e_j\}$, for all e_j;

   end

After the Petri net representation is finished(figure 46), we have used the GSPN software to simulate some task sequences on the system Petri net. It has also been verified that the net possesses the properties of safeness, boundedness and liveness. It does not satisfy the property of conservation because of the geometric characteristics of the system. All possible task sequences have been tested to go through the system Petri net from the initial states to some defined final states. And then the final states could be considered as the updated initial states and by following another possible sequence, the system could go back to its original initial states. We consider this one working iteration of the system.

Data Structure for Searching Sequences in Petri Net

The task planning problem is concerned with reachable markings when the system has been modeled with Petri net. The reachability problem is the most basic Petri net analysis problem[34]. Many other analysis problems can be stated in terms of the reachability problem.

Definition 8 The Reachability Problem Given a Petri net C with marking μ and a marking μ', is $\mu' \in R(C, \mu)$?

Another problem which is similar to reachability but is slightly different is called the coverability problem. A marking μ'' covers a marking μ' if $\mu'' \geq \mu'$.

Definition 9 *The Coverability Problem* Given a Petri net C with initial marking μ and a marking μ', is there a reachable marking $\mu'' \in R(C, \mu)$ such that $\mu'' \geq \mu'$?

Briefly speaking, finding a possible sequence concerns the reachability problem and finding all possible sequences concerns the coverability problem.

Before we describe the algorithm for obtaining the reachability tree for the system Petri net, we state some useful definitions, i.e., *frontier nodes, terminal nodes and duplicate nodes* [34].

Definition 10 *Frontier nodes, terminal nodes and duplicate nodes:* When we develop the reachability tree, all current new markings are called frontier nodes. The markings, in which no transition is enabled or depending on some criterion, no further transitions are necessary, are called terminal nodes. The markings which have previously appeared in the tree are called duplicate nodes.

In the system Petri net, we define the states of the system as markings. The algorithm therefore begins by defining the initial marking, which corresponds to the initial state in the system, to be the root of the tree. The final state of the system are defined as the final marking in the Petri net.

Algorithm 2 *The generation of reachability tree*
Input: Petri net $N_p = (P, T, \alpha, \beta, \mu_0)$, final marking μ_f.
Output: The corresponding reachability tree.
Let x be a frontier node to be processed.

1. If $x = \mu_f$, stop developing and classify x as a terminal node.

2. If there exists another node y in the tree which is not a frontier node, and has the same marking associated with it, $\mu[x] = \mu[y]$, then node x is a duplicate node.

3. If no transitions are enabled for the marking $\mu[x]$, [i.e., $\delta(\mu[x], t_j)$ is undefined for all $t_j \in T$], then x is a terminal node.

4. For all transitions $t_j \in T$ which are enabled in $\mu[x]$, [i.e., $\delta(\mu[x], t_j)$ is defined], create a new node z in the reachability tree. The marking $\mu[z]$ associated with this new node is, for each place p_i, $\mu[z]_i = \delta(\mu[x], t_j)_i$. An arc, labeled t_j, is directed from node x to node z. Node x is redefined as an *interior node*; node z becomes a *frontier node*.

5. When all nodes have been classified as terminal, duplicate, or interior, duplicate nodes will copy all descendants from its corresponding nodes. Then duplicate nodes will be reclassified as interior nodes.

6. When all nodes have been classified as terminal or interior, the algorithm stops.

We have 32 transitions and 17 different places in the Petri net. Place 1 to place 17 are Hc, Ho, G, HoG, HcG, R, RHcG, RHoG, RG, Pc, Po, RPoG, PoG, RPcG, PcG, SMOOTH and SEAM-ALIGN, respectively. Originally, we need 17 storage units to represent the current system states. Of course, the unit has only two value 1 and 0 to represent whether the single state exists in the whole combined states. The following is the linked data structure for the relationships among all transitions and places in the system. Obviously, it will not lose any information if we convert the markings into decimal digital numbers. The three tuples include three elements. i.e., the value of the next states after transition finished, the transition and next possible transition from the current states.

1 ⇒ 1(4096, 31, 0)
2 ⇒ 2(16, 30, 0)
4 ⇒ 3(16, 23, 0)
8 ⇒ 4(32, 20, 5) ⇒ 5(2052, 21, 0)
16 ⇒ 6(4, 24, 7) ⇒ 7(16448, 26, 8) ⇒ 8(2, 29, 0)
32 ⇒ 9(2064, 18, 10) ⇒10(8, 19, 11) ⇒11(320, 15, 0)
64 ⇒ 12(128, 27, 0)
128 ⇒13(64, 28, 0)
320 ⇒14(32, 16, 0)
512 ⇒15(10240, 9, 16) ⇒16(1024, 11, 17) ⇒17(33024, 13, 0)
1024 ⇒18(6144, 8, 19) ⇒19(512, 12, 0)
2052 ⇒ 20(8, 22, 0)
2064 ⇒ 21(32, 17, 0)
4096 ⇒ 22(1, 32, 23) ⇒ 23(8192, 5, 0)
6144 ⇒ 24(1024, 7, 0)
8192 ⇒ 25(49152, 4, 26) ⇒ 26(4096, 6, 0)
10240 ⇒ 27(512, 10, 0)
16448 ⇒ 28(16, 25, 0)
32768 ⇒ 29(65536, 1, 0)
33024 ⇒ 30(512, 14, 0)
49152 ⇒ 31(8192, 3, 0)
65536 ⇒ 32(32768, 2, 0)

For each transition, we have only one start state and end state. The following set of two tuples describes the corresponding start state and end state for each transition.

t_1(32768, 65536), t_2(65536, 32768),
t_3(49152, 8192), t_4(8192, 49152),
t_5(4096, 8192), t_6(8192, 4096),
t_7(6144, 1024), t_8(1024, 6144),
t_9(512, 10240), t_{10}(10240, 512),
t_{11}(512, 1024), t_{12}(1024, 512),
t_{13}(512, 33024), t_{14}(33024, 512),
t_{15}(32, 320), t_{16}(320, 32),
t_{17}(2064, 32), t_{18}(32, 2064),
t_{19}(32, 8), t_{20}(8, 32),
t_{21}(8, 2052), t_{22}(2052, 8).

$t_{23}(4, 16)$, $t_{24}(16, 4)$,
$t_{25}(16448, 16)$, $t_{26}(16, 16448)$,
$t_{27}(64, 128)$, $t_{28}(128, 64)$,
$t_{29}(16, 2)$, $t_{30}(2, 16)$,
$t_{31}(1, 4096)$, $t_{32}(4096, 1)$.

We then consider the initial states of the system as root node and generate all possible firing sequences based on breadth-first forward chaining method. We will develop the reachability tree step by step. If we have no final states for the system, the system will continue changing states and will never stop. Given the final states, we could develop some efficient algorithm to prevent the developing of the tree and then find all possible firing sequences at last. One possible stop condition is that if we find some developed node is equal to the final states, we just stop the developing procedure from this node. We will save all developing nodes in an array called the traversed array. Each time when we develop a new node, we should compare it with each node in the traversed array to see whether they are equal. If so, we stop the developing procedure from this new node and do not save it in the array. And we make a label for this node to copy all possible following nodes to this branch from the root node and ensure all possible sequences. If not, we save the new node in the traversed array and continue the developing procedure again. This is the second stop condition for reachability tree. The reachability tree of a Petri net has been proven to be finite[34].

Generation of Sequences from Petri net

Petri-net based representation of the system and the algorithm for generating all possible task-level sequences given the initial states and final states has been implemented. Some operations could be performed at the same time in the case of non-conflicting of resources. Therefore, Parallel operations are also implemented in this planning system. All possible sequences could be found in the reachability tree. One possible path from the root to the terminal node corresponds to a possible sequence to go from initial states to the final states.

Here we provide an example to see how this planner works. For the sake of convenience, we did not only set initial states and final states for the system to get the task sequences because a great number of sequences will be generated. To emphasize some important constraints and see whether the system really finishes some tasks, we add some intermediate states. For example, we define the initial states as (p_1, p_3, p_6, p_{10}) (all objects are in the static original states). The final state is (p_1, p_9, p_{10}) (robot holds handled garment). If we need a pressed garment from the robot gripper, the garment should have passed the state of seam align as well as smooth and pressing state. Therefore, the following procedure obviously represents the whole pressing sequences which should pass through initial states, intermediate states and final states.

(p_1, p_3, p_6, p_{10})**(initial states)** $\Rightarrow$ (p_6, p_{10}, p_{17})**(seam align)** $\Rightarrow$ (p_1, p_9, p_{11})**(robot holds garment)** $\Rightarrow$ (p_1, p_6, p_{16})**(smooth)** $\Rightarrow$ (p_1, p_6, p_{15})**(pressing)** $\Rightarrow$ (p_1, p_9, p_{10})**(robot holds handled garment-final states).**

We just choose one sequence from all possible sequences for this particular problem. The result is shown as follows:

$(p_1, p_3, p_6, p_{10}) \rightarrow t_2$**(handler opens)** $\rightarrow t_3$**(garment reaches handler)** $\rightarrow$ t_6**(handler closes and grasps garment)**$\rightarrow t_{32}$**(seam align)** $\Rightarrow$

$(p_6, p_{10}, p_{17}) \rightarrow t_{28}t_{31}$**(press opens and handler holds garment)** $\rightarrow$ t_7**(robot reaches handler which holds garment)** $\rightarrow$ t_{12}**(handler opens and robot grasps garment)** $\rightarrow$ t_{13}**(robot holds garment and leaves handler)** $\rightarrow$ t_1**(handler closes)** $\Rightarrow$

$(p_1, p_9, p_{11}) \rightarrow t_{16}$**(robot which holds garment reaches press)** $\rightarrow t_{18}$**(robot puts garment on press and leaves press)** $\rightarrow t_{29}$**(smooth)** $\Rightarrow$

$(p_1, p_6, p_{16}) \rightarrow t_{30}$**(garment on press and press opens)** $\rightarrow t_{24}$**(press closes and works)** $\Rightarrow$

$(p_1, p_6, p_{15}) \rightarrow t_{23}$**(press opens)** $\rightarrow t_{17}$**(robot reaches press)** $\rightarrow t_{15}$**(robot grasps garment and leaves press)** $\rightarrow t_{27}$**(press closes)** $\Rightarrow (p_1, p_9, p_{10})$

Conclusions

The application of Petri net concepts in the task level planning for robotic garment handling system has been described in this section. We could obtain all possible task sequences after constructing the reachability tree from the Petri net. The method for mapping system AND-OR network to Petri-net is straightforward. This off-line planning system has been implemented.

2.4.1.2 Object-Oriented Integration

This section describes an object-oriented approach to the integration of an automated garment handling system. Both the software architecture and the hardware configuration of the system have been carefully addressed. To implement the object-oriented integration of the whole system and coordinate the operations of all devices, we have set up a hierarchical structure of the object-oriented environment. The highest layer is a task planner which receives input from the user and the system Petri net descriptions. The second layer is the operations planner which receives input from the task planner and generates the operation sequences and sends the sequences to the lowest control and execution layer. The commands generated at the lowest layer could be directly sent out to control the robot, sensor and other mechanisms in the system. According to the hierarchical structure of system integration, we have also developed a hierarchical command language to implement the software architecture. The definitions of high level commands will be proposed in this paper.

The high level algorithm for performing the pressing task in the garment handling system could be described as follows:

Algorithm Performing the pressing task.
Input: Waiting garment.
Output: Garment which has been pressed.
Background: Garment handling system
Constraint: Well pressed.
Let g be a garment object to be processed.

1. Transfer g from the garment handler to the seam alignment device. This is currently accomplished by human hands.

2. Use vision sensor to detect the seam first. Run a corresponding algorithm to calculate how far (how many turns) seam should be moved to be aligned. Then the vision sensor might be used again to confirm the seam alignment.

3. Control the robot gripper to grasp g and transfer it to the press.
    3.1 The two clamps of the robot gripper hold the two ends of g which is grasped by the seam alignment device.
    3.2 The seam alignment device releases the garment and the robot gripper fully holds g.

4. Robot transfers g to the press following a fixed path.

5. Press cycle starts.
    5.1 After the robot gripper reaches the surface of the press, the cuff end gripper opens and traverses off the press buck.
    5.2 Smoothing is done by the robot if required.
    5.3 Press closes and starts cycle.

6. Press opens and the g is removed.

The *interface* between the human and the system is the host computer. Before the system starts working, the human needs to input the initial states and final states of the system to the computer. The computer will first generate all possible high level task sequences and among them, choose the most efficient one and then generate the lower level operation sequences corresponding to each task sequence. Each lower level sequence will at last be mapped to a machine level command program. Each program will be directly used to control the robot and other mechanisms in the system. During the working of the system, new information might also be fed back to the computer and the control commands might be updated to coordinate the system.

The initial task in the development of system integration is the definition of hardware and software architectures to control the various devices and operations which will be required in the implementing of the automated garment handling. Based on our initial definitions of the task, we have defined these architectures as follows:

Hardware Architecture:

The system includes a set of control mechanisms and sensor devices which interact in the execution of the overall task. The integration of these devices and sensors is accomplished by a host computer(figure 47). This host computer incorporates centralized planning and control functions as well as coordinating communications among the sensors and devices. From a hardware standpoint, the host computer is physically linked through communications interface boards to each of the control mechanisms and sensors. Some control mechanisms such as the GMF robot have their own controllers and controller languages. In these cases the host computer provides controller level commands across the communications link. In other cases, the mechanisms and sensors are interfaced directly to the host computer, and in this case local controller boards and software has been developed to provide direct interface to these devices. In the initial phase of the project, reliability of communication and control has taken priority over real time speed of execution. Therefore, this first generation architecture emphasizes structured modular interfaces and communications protocols. More complex dynamic interactions between sensors and controllers will be added to the system in later developments.

Software Architecture:

The software architecture for the system is based on a configuration state description of the mechanisms and objects being handled. Systematic enumeration of states and sub-states of the system is utilized as a basis for generating sequence plans of operations for execution of the tasks. The systematic state description of the process also facilitates the generation of error

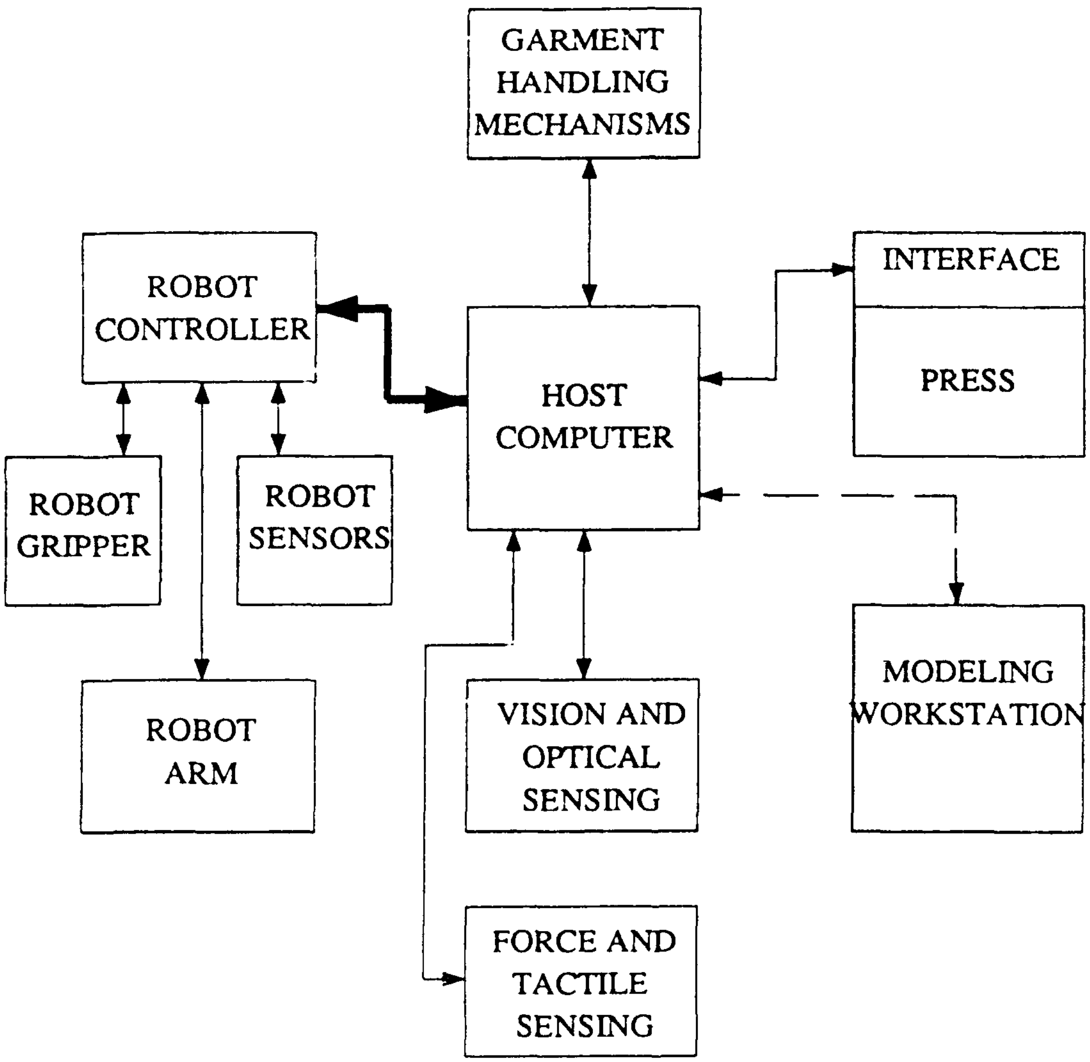


Figure 47

HARDWARE CONFIGURATION
AUTOMATED GARMENT HANDLING SYSTEM

recovery and state verification techniques. The overall software includes a user interface which permits operator interaction with the system, a planning level which permits sequencing of operations, and an execution level. Within the execution level is a set of device and sensor software modules and drivers which incorporate the specific motion control, sensor interpretation, and communication protocols for the various devices. The execution level of software specifies three classes of motion which may be mapped onto various devices and mechanisms. These three classes include gross motion, fine motion, and grasping. The execution level also recognizes three classes of sensor functions. These include state validation, state interrogation, and sensor-based tracking. The task level control structure coordinates the mapping of task and operation sequences from the planner onto functional modules at the execution level. In addition, the software system incorporates a data-base structure which accommodates model-based information regarding state configurations, geometry, and material properties. In the first stage of development of the systems, this model-based information is not used for control and execution. However, as the system software architecture develops, model-based information becomes fundamental to the development and implementation of task plans.

Background

An object-oriented approach to software design was derived from work on information hiding[46], abstract data types[47], and, most significantly, from work on object-oriented programming languages like Smalltalk[48], C++[49], and concurrent Prolog[50]. Object-oriented languages simplify the implementation of an object-oriented design, but the principle of designing a system as a set of interacting objects is distinct from implementing that system[51]. Object-oriented design differs from the more familiar functional approach to design in that it views a software system as a set of interacting objects, with their own private state, rather than a set of functions. Conceptually, objects communicate by passing messages to each other, and these messages initiate object operations. Thus, communication may be asynchronous and an object-oriented design may be realized as a parallel or a sequential program. In practice, although the operations in an object are implemented by corresponding procedures or functions, they are actually the point of implementation rather than the original idea of design. Using object oriented design to implement the integration of our garment handling system, we can limit the communications between the objects in the system to message sending. No shared data areas exist among all the objects in the system. Because each object will become an independent entity, the changes and accesses of the state and information on a particular object in the system may be made without reference to other system objects. Furthermore, object oriented design may eventually implement the parallel operations in the distributed objects given by the Petri net based task planner.

In an object-oriented language, everything is defined as an object or system of interacting objects. Individual objects are implemented as framelike structures whose slots contain either variables or procedural information. Procedures are expressed in an appropriate programming language as mentioned above or even a rule-based expert system language[52]. The major goals of choosing a convenient object-oriented language are to improve programmer productivity and cost of software maintenance. When object-oriented programming is used, the design phase of software development is linked more closely to the implementation phase.

Many discussions have been posed on the concepts in the area of object-oriented design and object-oriented programming[53-56]. For the sake of more clear understanding of our future discussions, we first propose some relevant definitions and descriptions as follows:

class: A class is a description of one or more similar objects. For example, the class *garment* is a description of the structure and behavior of instances, such as garment_1 and garment_2, which have different color but the same size, style and materials. Objects of the same class have common operations and therefore uniform behavior. Classes have one or more "interfaces" that specify the operations accessible to clients through that interface. A "class body" specifies code for implementing operations in the class interface.

object: The primitive element of object-oriented programming. An object has a set of "operations" and a "state" that remembers the effect of operations. Objects combine the attributes of procedures and data. Objects store data in variables, and respond to messages by carrying out methods.

instance: The term "instance" is used in two ways. The phrase "instance of" describes the relation between an object and its class. The methods and structure of an instance are determined by its class. The noun "instance" refers to objects that are not classes.

class variable: A class variable is a variable stored in the class whose value is shared by all instances of the class. Compared with *instance variable.*

instance variable: Instance variables (sometimes called *slots*) are variables for which local storage is available in instances. This contrasts with class variables, which have storage only in the class.

method: In object-oriented parlance, the operations that are defined for class are called methods. These methods are analogous to procedures and functions in non-object-oriented languages. *Method* implements the response when a message is sent to an object.

message: The specification of an operation to be performed on an object. The process of invoking a method is called *sending a message* to the object. Such a message typically contains parameters just as in a procedure or function called invocation in a non-object-oriented language. The invocation of a method (sending a message to an object) typically modifies the data stored in that particular object.

abstract data types: Abstract data types are the centerpiece of object-oriented programming. An abstract data type is a model that encompasses a type and an associated set of operations. These operations are defined for and characterize the behavior of the underlying type.

For this project, we use C++ programming language to implement the object-oriented integration for the garment handling system. C++ was created by Bjarne Stroustrup in the early 1980s. C++ is an extension of C with various facilities, not all of which have to do with object-orientedness. Though C++ is at the problem level, it still maintains contact with the machine level implementation if applicable. Class in C++ is a very good implementation of the abstract data type. Before we discuss an example of class which we implement in C++ language, we will briefly describe the differences between *class* and *object.*

Objects are real entities which could be comprehened in the real world. The integration of our system is based on the characteristics as well as some operations relating to the objects. In the C++ environment, one or more similar objects, i.e., the same kind of objects, are represented as a class. In other words, one kind of object corresponds to one class.

Class is illustrated by the following example:

```
class ROBOT
{
int gripper_state;
int arm_load_state;
```

```
int where;
public:
robot(int, int, int);
int get_gripper_state();
int get_arm_load_state();
int get_where();
void change_gripper_state(int);
void change_arm_load_state(int);
void set_r_place(int);
};
```

In this class, there are three private variables which could only be accessed within the object ROBOT. The first operation defined in the member functions of the class is the instance initializer. The following operations perform the functions either to get the current state or to update the current state. We propose a detailed explanation as follows:

int gripper_state: gripper is open or not. "0" represents *open* and "1" represents *closed.*
int arm_load_state: gripper is loading garment or not. "0" represents *empty* and "1" represents *full.*
int where: where the gripper is. We define a set of integers corresponding to the places of all mechanisms in the system. Therefore, *where* represents the value of this integer.
get_gripper_state(): obtain the value of gripper_state.
get _arm_load_state(): obtain the value of arm_load_state.
get_where(): obtain the value of where.
change_gripper_state(int): reset the value of gripper_state.
change_arm_load_state(int): reset the value of arm_load_state.
set_r_place(int): reset the value of where.

High Level Object-Oriented Design for the System

From figure 44, we know that the main objects of the system are ROBOT, GARMENT, HANDLER, SENSOR, PRESS and HOST COMPUTER. The host computer will interpret the high level operation commands to the lower level control and execution commands and then sends these commands to the controller interfaces and other interfaces to directly drive the corresponding devices to do some desired operations so that the current states of the system could be updated. Although the host computer takes the most important and key role in keeping and changing the states of the whole system, from the view of the users, they could only directly see the changes of the states of other mechanisms such as robot, handler, sensor, etc. Moreover, the users need a convenient interface to interact with the system, i.e., the host computer. Therefore, we don't consider the host computer as one class in our object-oriented system. This is due to the fact the host computer is an implementer of the system integration and object-oriented programming. Because the command *(including the high level command and lower level command)* is the only interface between the host computer and outside world of the system, to make the implementation more convenient, we also consider the command as an class in the system.

Therefore, we consider the six main classes for the system, i.e., ROBOT, GARMENT, HANDLER, SENSOR, PRESS and COMMAND. We only consider the high level object-oriented implementation here. COMMAND has three different formats. One format of command need not specify the name of the object but just describe the characteristics of the operation. For example, if we want to assign a command for *seam_align,* we could just give out the description of the operation. Another kind of command needs one object as operand. This is because many objects might do the same operation like *"open"* or *"close"*. The third kind of operation has two operands and one operator. For example, if the robot is assigned a task to move the garment to the press, this command needs one *"move"* operator and two operands *"garment"* and *"press"*. It will be convenient if we separate the command into several parts such as operands and operator. Thereafter, we could easily construct each separated command part from *string* structure or class and then construct the command from those separate parts. We introduced the class COMMAND_OP for the reason mentioned above.

We also include the two classes MENU and INTERFACE for the sake of conveniently making the communications between users and the host computer and between the host computer and the system. The diagram of these communications are shown in figure 48.

Now we identify the hidden data field as well as operations in each object. The hidden data field could not be directly accessed by other classes. For the high level system design, we pay the most attention to the current states of the system. The meaning of the system states is the combined states of all objects in the system. The high level command language is based on the high level system state descriptions. Therefore, for each class in the system, we pay attention only to the state description of each object. The specific descriptions of the private data field for each object are as follows:

1. ROBOT:

    gripper_state: the gripper is open or not,
    arm_load_state: the gripper is full or not,
    where: the place of the gripper.

2. GARMENT:

    align_state: the garment is aligned or not,
    smooth_state: whether the garment is smooth or not,
    where_on: the place of the garment.

3. HANDLER:

    handler_state: the handler is open or not,
    handler_load_state: the handler is full or empty.

4. PRESS:

    press_state: the press is open or not,
    press_load_state: the press is full or not,
    press_work_state: the press is working or not.

5. SENSOR:

    sensor_place: the place of the sensor,
    smooth_info: the degree of the smooth state,
    align_info: whether the seam is aligned or not.

6. COMMAND:

    command_op1: the operator of the command,
    command_op2: the first operand of the command,
    command_op3: the second operand of the command.

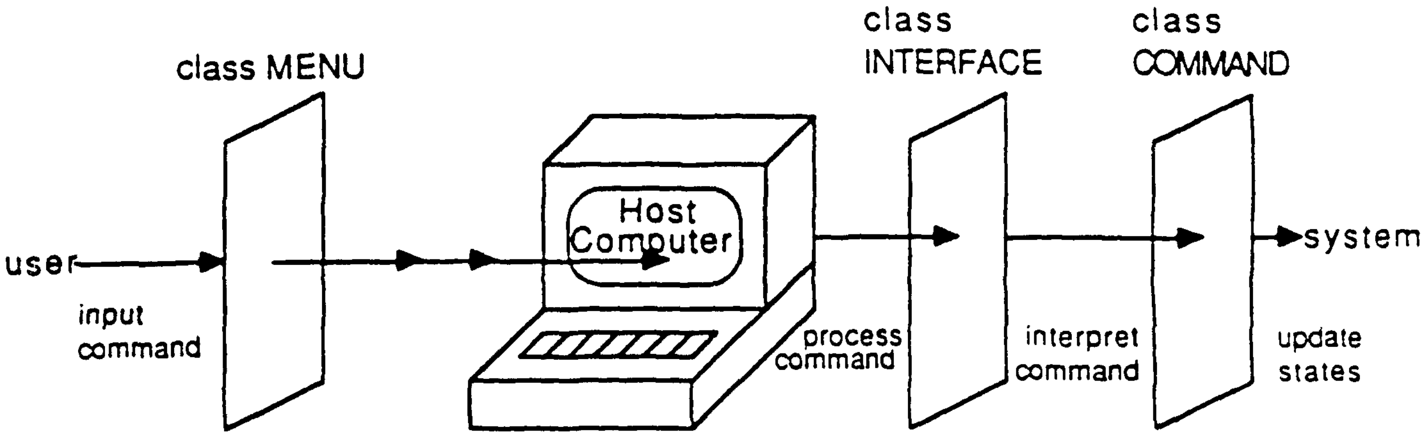


Figure 48 High-Level Command Flow from USER to SYSTEM

7. COMMAND_OP:

size: the string length,
op: string of characters.

8. INTERFACE: *none.*

9. MENU: *none.*

The operations contained in each object are also called methods, which could be thought of as functions or procedures existing in the object. When a message is sent from other object to this one, one particular method will be fired and the system states will be updated. Those operations of high level objects are listed as follows:

1. ROBOT:

initialize_robot_instance: create a new *robot* object instance,
get_gripper_state: obtain the value of gripper_state,
get_arm_load_state: obtain the value of arm_load_state,
get_where: obtain the value of where,
change_gripper_state: reset the value of gripper_state,
change_arm_load_state: reset the value of arm_load_state,
set_r_place: reset the value of where.

2. GARMENT:

initialize_garment_instance: create a new *garment* object instance,
get_align_state: obtain the value of align_state,
get_smooth_state: obtain the value of smooth_state,
get_where_on: obtain the value of where_on,
set_where_on: reset the value of where_on.

3. HANDLER:

initialize_handler_instance: create a new *handler* object instance,
get_handler_state: obtain the value of handler_state,
get_handler_load_state: obtain the value of handler_load_state,
set_handler_state: reset the value of handler_state,
set_handler_load_state: reset the value of handler_load_state.

4. PRESS:

initialize_press_instance: create a new *press* object instance,
get_press_state: obtain the value of press_state,
get_press_load_state: obtain the value of press_load_state,
get_press_work_state: obtain the value of press_work_state,
set_press_state: reset the value of press_state,
set_press_load_state: reset the value of press_load_state,
set_press_work_state: reset the value of press_work_state.

5. SENSOR:

initialize_sensor_instance: create a new *sensor* class instance,
get_smooth_state: obtain the value of smooth_state.

get_align_state: obtain the value of align_state,
get_sensor_place: obtain the value of sensor_place,
set_place: reset the value of sensor_place.

6. COMMAND:

constructor: construct a *command* object instance,
destructor: destruct a command object instance,
initialize_command_instance: create a new command object instance,
get_op1: obtain the operator of the command,
get_op2: obtain the first operand of the command,
get_op3: obtain the second operand of the command,
command_interpreter: analyze and interpret the command obtained,
print_command: show the command on the screen.

7. COMMAND_OP:

constructor: construct a *command_op* object instance,
destructor: destruct a command_op object instance,
initialize_command_op: create a new command_op object instance,
get_cop: obtain the value of command_op,
print_command_op: show the value of command_op on the screen.

8. INTERFACE:

display_get_choice1(character): show the choice given by the user on the screen which is represented in character,
display_get_choice2(integer): show the choice given by the user on the screen which is represented in integer,
process_command: receive the command typed by the use and scan it.

9. MENU:

display_title_menu: display the title menu on the screen,
display_main_menu: display the main menu on the screen,
display_manual_menu: display the manual-driven menu on the screen,
display_auto_menu: display the auto-driven menu on the screen.

Software Architecture for the Object-Oriented Environment

The system software architecture consists of three layers. Each layer is supported by and supervises the layer below. The highest layer is designed for the user and the lowest layer is designed for the controller interface of the various devices in the system. The user could also interact with the lower layer if needed. The hierarchical structure is also embodied in the command language and object-oriented programming for the system. Using layer concepts, the users could only care about the high level description of the system and send high level commands to the system. The implementation details of these commands are normally not transparent to the user. Therefore, we define the system software architecture as shown in figure 49. As mentioned above, from figure 49, we notice that the user could interact with three layers, i.e., task planner, operations planner, and control and execution layer. However, the user will have the closest relationship with the task planner. Even if the lower layer feeds back errors or help information to the task planner layer, the task planner could re-select another sequence from all possible sequences. If necessary, the planner could also replan the

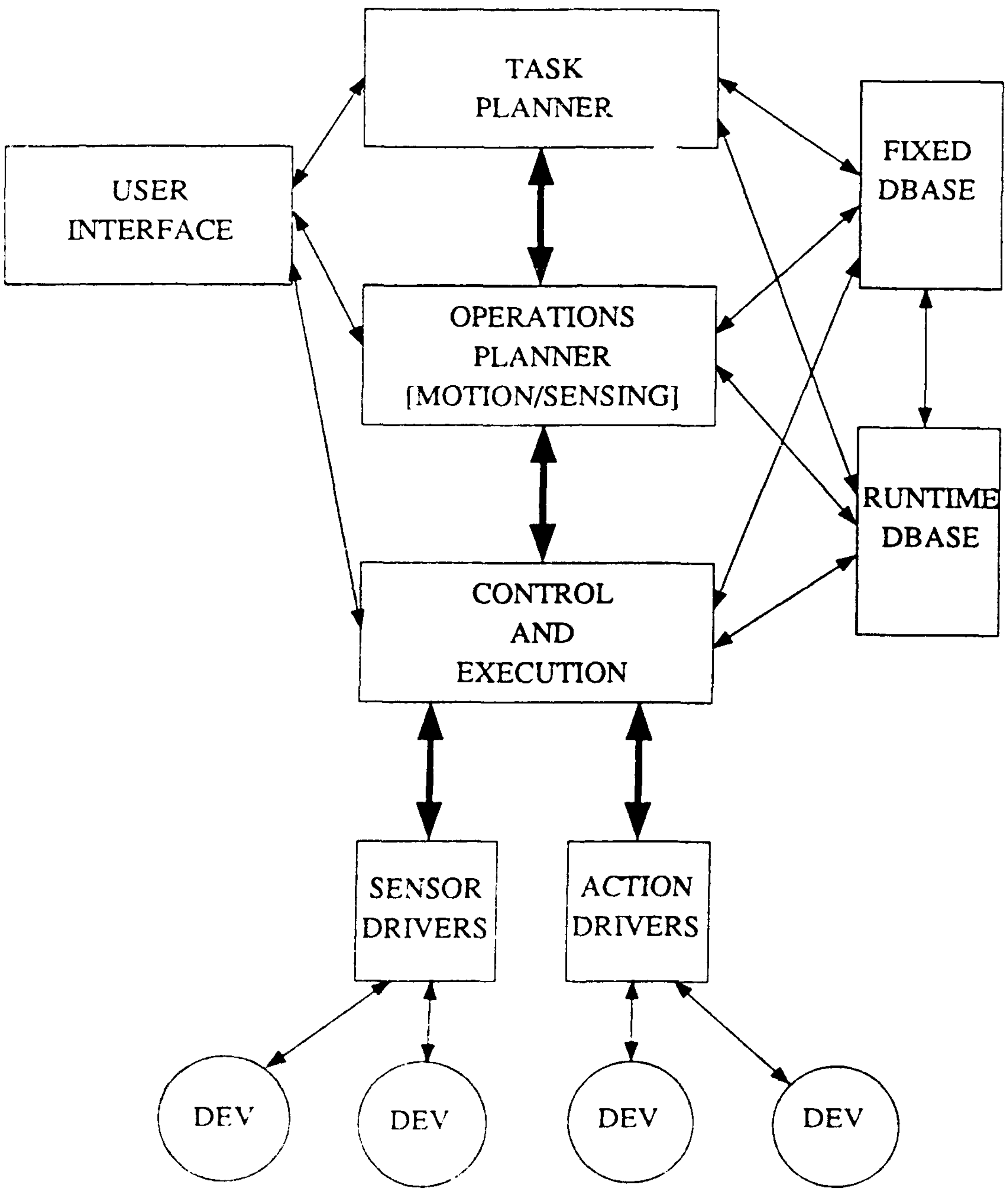


Figure 49

SOFTWARE ARCHITECTURE

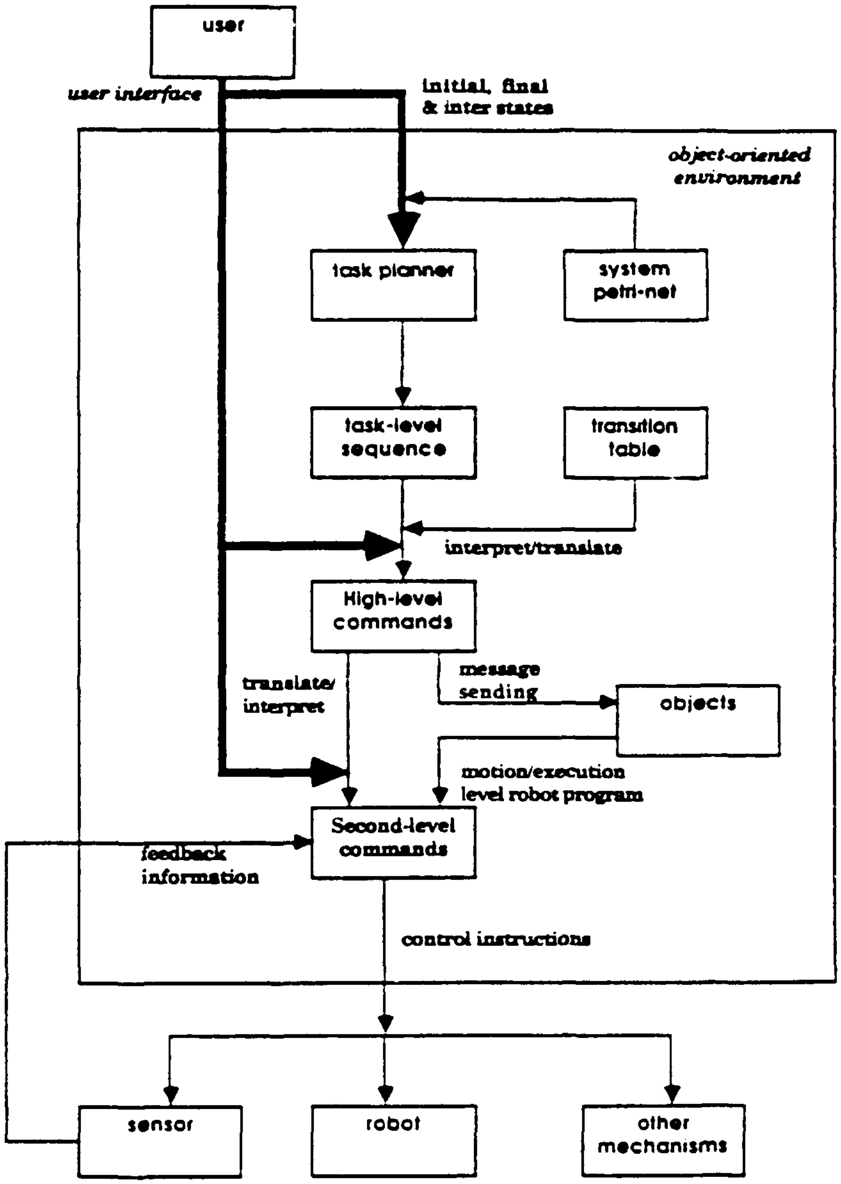


Figure 50 Object-Oriented Environment for the Garment Handling System

sequences or ask the user to provide a new set of initial state, intermediate states and final state. Thus, the error recovery could be handled systematically and efficiently.

The initial planning sequences could be generated using off-line programming methods. The details of the work of task planning as well as sequences generations have been discussed in previous sections. The highest level task planner will then send the selected task sequence to the lower layer operations planner. The planner of this layer will scan the task sequence and plan each motion/sensing sequence corresponding to each distinct operation in the high level sequence. For example, if the operation is to order ROBOT which is grasping GARMENT move from the place of HANDLER to the place of PRESS, the operations planner will decide which path to follow. The problem of path planning for the robot will then be proposed and solved. Another example is that if the seam_align operation is needed to be executed, then the first operation should move SENSOR to the place of HANDLER and then the sensing operation could start. We should plan both motion and sensing sequences in this case. If the operations planning could not be accomplished because of the dead-lock of resources, the operation planner layer may send a message to its upper layer and suggest the task planner to replan the sequences.

Although the operations planner contains more details about the practical updating of the states as well as operations of the system, it is not the lowest level of the software integration system because it could not be used to directly drive the devices in the system. This layer is the bridge between the user and the system. Each operation command should be mapped to or interpreted as more complete and more realistic commands which could be directly recognized by the objects in the system. Therefore, the task of the third layer in the software architecture is to receive the intermediate commands from the layer above and interpret or translate them. Then the machine-level control commands will be sent out in a real time fashion. The feedback information as well as error handling request from the system should be also passed to the control and execution layer as shown in figure 50.

As we will mention in the next section, we will define a command language which will include both high level commands and lower level commands. Although it is not necessary for the operator of the system to be familiar with the command language because the information of the states of the system is sufficient for him to operate the system, it is still important to make the system more flexible and reliable through the development of the command language. The connections of the user interface with the operations planner and control and execution layer is through the commands from the command language.

Additionally, we define the connections between fixed database or runtime database to all three layers and the connection between the two databases themselves. The fixed database contains a Petri net representation of the system, garment handling knowledge, modeling information, descriptions of command language and other relevant knowledge. The runtime database includes the system states, error information and other information which describes the run-time records. The fixed database may accept the information from the runtime database as fixed data. The runtime database may also absorb the data from the fixed database as reference. The information in the database is accessible by all three layers of the system architecture.

It seems that we could not feel the existence of the fixed database or runtime database in our object-oriented environment. Actually, the contents of these two databases have been distributed to all objects in the system. The independent and safety characteristics of data in the object-oriented system are consistent with those in the fixed or runtime database. The communications with the fixed or runtime database could be considered as message sending procedures.

Object-Oriented Environment for the System

We combine the object-oriented approach with the hierarchical structure and derive the object-oriented environment for the garment handling system as shown in figure 50. We could consider the object-oriented environment as the interface between the user and the devices in the system. The data the user inputs to the environment is ultimately transformed to the data faced to the machines. We give a formal definition for the object-oriented environment as follows:

object-oriented environment: Object-oriented environment is a convenient and efficient interface between the operator and the object-oriented system. The objects and command generation in the system are all defined as hierarchical structures. The machine level programs corresponding to the special high level task will be generated in sequence and the implementation of the object-oriented environment is through object-oriented programming.

We have actually discussed most of the environment in the previous sections. As mentioned in previous sections, the task planner will accept initial, final and intermediate states from the user and system Petri net description from the fixed database as input and generate the high level task sequences. The elements in the task sequences correspond to the transitions in the Petri net. However, some transitions in the Petri net have similar characters or relate to the same operations. Thus, we should interpret each transition in each sequence to a sequence of macros or high level commands based on a transition table (figure 51). The generated high level commands will then be sent to the high-level objects. We call this procedure message sending. Before a message is sent to a desired object, the object matching process should be performed. After we eventually find the corresponding method in the object, we then fire this method and the corresponding data or states should be modified. Then message sending could be considered as finished. For the sake of the completeness, we compare the descriptions of high-level objects and lower-level objects as follows:

high-level object: A high-level object is a task-oriented entity when we use the object-oriented approach to implement the system. The set of high-level objects may be equal to that of major objects in the system because most of the operations are concerned with the major objects. All methods in the high-level objects are described as high-level commands in the corresponding command language.

lower-level object: A lower-level object is an implementation-oriented entity in the object-oriented system. Each lower-level object is either a lower-level command generator or controller or some other object that is not a major object in the system. When the method in a certain lower-level object is fired, a corresponding control instruction will be generated to directly drive some device in the system.

The communications between high-level objects and lower-level objects are also through *message sending* (figure 52). The motion/execution level robot or sensor programs generated by the lower-level objects form the second-level commands. These commands construct the control and execution layer of the system software architecture. The sensory information may be fed back to this layer and if necessary, the lower layer may ask for help from upper layer.

Under the object-oriented environment, each object will coordinate with others and data flow in the system could be clearly tracked. Moreover, if we want to add some new objects to the system, we might only pay attention to a certain part of the system. Therefore, the system will have a strong extensibility. Also, the error handling in the system will be convenient because the errors may concern only one or several objects.

Transition	Set of Macro
$t1$	*Macro 1, Macro 2, ..., Macro* i_1
$t2$	*Macro 1, Macro 2, ..., Macro* i_2
...	...
t_n	*Macro 1, Macro 2, ..., Macro* i_n

Figure 51 Transition Table

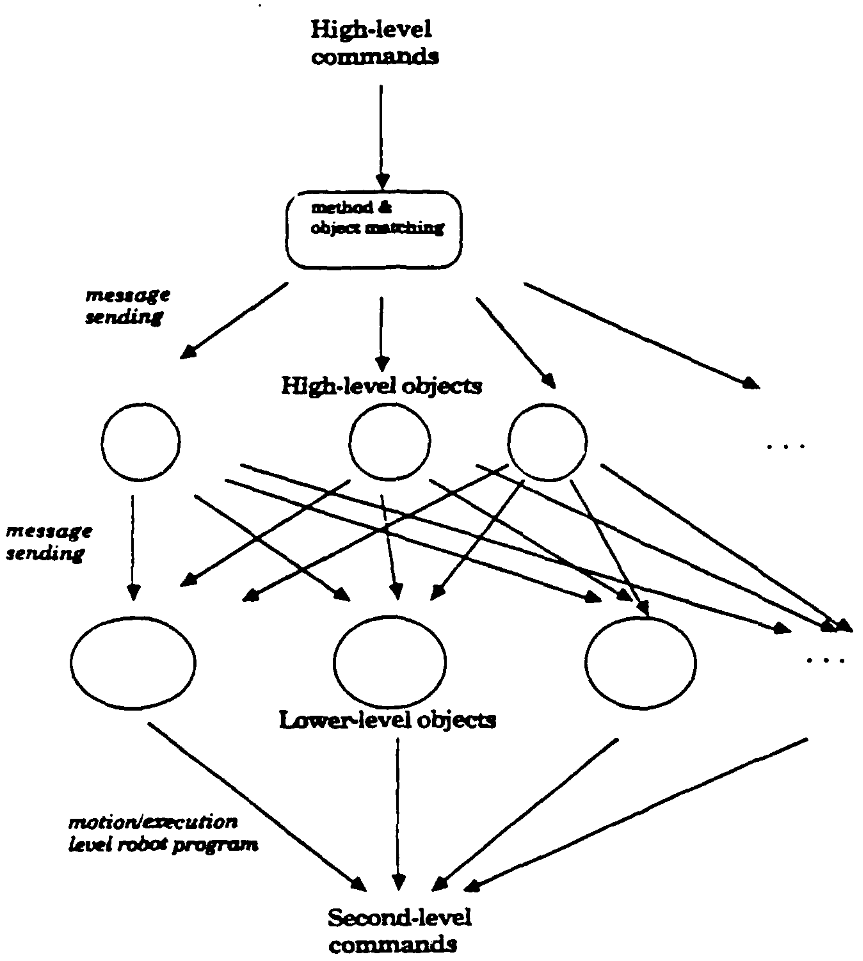


Figure 52 Hierarchical Structure of Objects and Their Communications

High-level Command Language for the Object-oriented Environment

As mentioned above, suppose the user only interacts with the task planner in the object-oriented environment, i.e., the input of this environment is the state flow chain shown below:

START st→ INTER st_1→INTER st_2→ →INTER st_n→FINAL st

A complete sequence from START state to FINAL state could thus be combined as:

(sequence 1)(sequence 2)(sequence 3) . . . (sequence n+1),

which corresponds to a sequence of transitions shown in the Petri net diagram.

Due to the hierarchical structure for the planning system, using the translating and/or *interpreting technique, we could ultimately generate the corresponding command sequence,* which could be directly implemented or executed by the control or execution mechanisms in the system.

To accomplish this aim, we should first generate the macro-instructions. Every transition in the Petri net model of the system could then be considered as just a macro or a set of macros. Notice we only consider the high level objects here.

Each macro consists of operator and/or operand(s). Some macros may have no operand because the operator itself contains enough information. For instance, *seam_align* will need only one operator to express the operation. Some macros need one operator and one operand. *Open press* and *open handler* are two examples of this case. Other macros need two operands and one operator. This kind of macro is mostly related to the robot. If the robot is ordered to move the garment to the press, the macro needs both garment and the place of press as two operands in the command. The initial high level commands are defined as these three kinds of macros.

The following is the definitions of the high level command language for the object-oriented environment of the system. We first give the five high level objects. They will appear in the macros as operands.

handler, garment, press, robot, sensor

Then, a set of operators which will have either default operand, one operand or two operands are defined as follows:

tse ⇒ to the state of SEAM_ALIGN
fse ⇒ from the state of SEAM_ALIGN
tsm ⇒ to the state of SMOOTH
fsm ⇒ from the state of SMOOTH
ope object_name ⇒ open (some mechanism)
clo object_name ⇒ close (some mechanism)
onw object_name ⇒ electricity is on
ofw object_name ⇒ electricity is off
rea object_*name* $_1$ object_*name* $_2$ ⇒ object *1* reaches object *2*
lea object_*name* $_1$ object_*name* $_2$ ⇒ object *1* leaves object *2*
rel object_*name* $_1$ object_*name* $_2$ ⇒ object *1* releases object *2*
obt object_*name* $_1$ object_*name* $_2$ ⇒ object *1* obtains object *2*

Every transition in the Petri net model (figure 46) could now be presented by the macros:

$t_1 \rightarrow$ clo handler

$t_2 \rightarrow$ ope handler

$t_3 \rightarrow$ rea garment handler

$t_4 \rightarrow$ lea garment handler

$t_5 \rightarrow$ ope handler
rel handler garment

$t_6 \rightarrow$ obt handler garment
clo handler

$t_7 \rightarrow$ rea robot handler

$t_8 \rightarrow$ lea robot handler

$t_9 \rightarrow$ lea robot handler
lea robot garment

$t_{10} \rightarrow$ rea robot garment
rea robot handler

$t_{11} \rightarrow$ obt handler garment
clo handler

$t_{12} \rightarrow$ ope handler
rel handler garment

$t_{13} \rightarrow$ obt robot garment
lea robot handler

$t_{14} \rightarrow$ rea robot handler
rel robot garment

$t_{15} \rightarrow$ obt robot garment
lea robot press

$t_{16} \rightarrow$ rea robot press
rel robot garment

$t_{17} \rightarrow$ rea robot press
rel robot garment

$t_{18} \rightarrow$ lea robot garment
lea robot garment

$t_{19} \rightarrow$ clo press

t_{20} → ope press

t_{21} → lea robot press

t_{22} → onw press
rea robot press

t_{23} → onw press
ope press

t_{24} → clo press

t_{25} → rea garment press

t_{26} → lea garment press

t_{27} → clo press

t_{28} → ope press

t_{29} → tsm

t_{30} → fsm

t_{31} → tse

t_{32} → fse

Using the above definitions of the high-level command language, we obtain the high-level commands for the operations planner layer according a special task. We show a simulation example to see how the high level command is analyzed and executed as follows:

In response to: rea *robot handler*, the system will perform the following operations:

(1) Send message to ROBOT class to create a new instance.
(2) Send message to HANDLER class to create a new instance.
(3) Check operator. Get *REACH*. Send message to ROBOT instance with the parameter *HANDLER*.
(4) Create a corresponding lower level class instance. ROBOT send a message to this instance.
(5) The lower level class instance send program to the robot controller, sensor and/or other mechanisms.
(6) The lower level class instance notify ROBOT that the task has been finished.
(7) Go to the next macro-instruction.

Up till now, we have completed implementation of some parts of the object-oriented environment for the garment handling system as shown in figure 50. We have finished the design and simulation of the user interface, task planner, system Petri net representation, transition table, high level command language as well as the high level objects and their relationships. The second-level command language and the design of machine level programs are ongoing.

Conclusion

The object-oriented approach described provides flexible architecture for integrating the garment handling system. The representation of objects as a hierarchical structure offers a natural grouping of real entities in the system. The object-oriented environment could be considered as a transformation function which takes the state descriptions as input and generates the machine-level control instructions as output. Sensory feedback information and error handling request could also be sent to the environment as adjust information. Much effort has been spent on defining the high level command language. The most important approach we have performed may be the unity of the description of the hierarchical software structure, objects and command language. The concepts of object oriented environment could be further derived and applied to other robotic application areas.

2.4.2 Control Hardware and Software

Introduction

This section of the report will cover the work done to bring together the necessary resources and technology, both physical and computer software, to enable the eventual integrated control of all components in the garment handling system. It does **not** cover the research and software for developing the object-oriented integration software and petri net based task planning (this is covered in other sections), but rather addresses the other hardware and software requirements that the integration software will need to control the operation of the system. So, in this section, integration refers to the high level, object-oriented, integration and task planning research and software, whereas control refers to all of the other assets required by the integration software to manage operation of the system's constituent components.

Overview

In the integration software, there will essentially be an internal representation or model of each of the various components of the garment handling system. The model for each component will represent the different states the component can be in, as well as the various operations the component can perform in its different states. The task of the control hardware and software, then, is to make certain that the state of each component's model is kept identical to the actual physical state of the corresponding component in the system. Note that this is a two way process, as the integration software must both initiate operations that change the state of the component, as well as respond to changes in the state of the component initiated by external events.

The task of control then, is to present the integration software with a set of interfaces from each of the component's models to the hardware of its corresponding component. These interfaces should be consistent in their structure, so as to eliminate duplication of effort in constructing the interface for each model. Each interface must, however, differ in content to reflect the different requirements for each component. The interface is then the medium of communication between the model and the component, but not the message.

Since there is a wide variation in how the host computer will control the various components of the system, both from component to component and from the research system to the final, commercialized system, the integration software will have to be tailored to suit each component. The structure of each interface unit (one for each model, component pair) will then only remain the same between the unit and the model, and differ between each unit and its component. The control hardware consists of all of the various devices that allow the various integration modules to control the components.

General Approach

The entire garment handling system will be controlled by one central computer, called the host computer. The sum total of all the software that will run on the cell controller will be called the system control software. It will consist of the integration software, control software and the operating system. In any software project, it is desirable to break down the task into smaller parts that are more easily developed and tested, and then to assemble the parts together. In a large project such as this, it is absolutely mandatory.

This modularization can be achieved with two basic methods. The first, is to make each module a sub-program or function. Then, the functions are combined, along with the main program, to form one large program. This approach is really only viable for smaller scale projects, since it requires the software engineers to keep track of all of the details necessary to have the functions interact with each other and to allocate the resources of the computer to each function. It also complicates the testing of each function. Since functions cannot be run by themselves, a test program must be written to verify the proper operation of each function.

A more flexible approach is to employ an operating system that handles the interaction and resource allocation chores, and allows each module to be written as a stand alone program, thus simplifying testing. Such an operating system is known as a multi-tasking operating system with inter-process communications. Furthermore, the operating system should be real-time, which means that it will respond to external events within a predictable, repeatable amount of time.

For development of the cell control software, we chose the UNIX operating system. While UNIX is not real-time, most real-time operating systems are modelled after UNIX and employ the same or similar multitasking and inter-process communication features, thus simplifying the task of porting the cell control software to a real-time operating system. There are even real-time versions of UNIX available. Since real-time operation is not necessary for development, we chose UNIX because it is multi-user and provides what is probably the richest software development environment available for any operating system. This greatly enhances and simplifies the process of having many different persons programming various modules of the software. UNIX was also built from the ground up on the philosophy of having small, specialized, easily understood and tested stand-alone programs combined to make large applications, which is precisely the approach we require.

Specifics

The Host Computer

The basic computer is an IBM(TM) compatible ISA (PC/AT) bus machine with a 33 MHz 80386 CPU, 8 Mb of DRAM, a 64 KB SRAM cache, a 150 Mb ESDI hard disk, and Super VGA color graphics. The machine was originally run under MS-DOS version 3.3 in a 40 Mb partition of the hard disk. The machine now runs under SCO's Open Desktop(TM) operating system, which is an implementation of UNIX System V Release 3 bundled with X-windows and the Motif(TM) user interface. The software also includes the Ingres(TM) relational database and Merge386(TM) (for running DOS under UNIX). UNIX resides in the remaining 110 Mb of the hard disk. The computer is connected to RPI's fiber optic backbone network via the CMPTT's Ethernet sub-network, allowing access to ARPAnet, NSFnet, "The INTERNET" et. al.

Software development is done using the Open Desktop Development package and a C++ compiler from Oregon Software. All I/O is currently being conducted through existing DOS based packages running under UNIX. This currently includes communications with the GMF robot and the seam alignment device. I/O is being ported to UNIX for consistency and improved performance. Besides offering a temporary solution, the capability of running the DOS based packages under UNIX gives us a known benchmark for testing of the ported software. The ISA bus and DOS combination is far and away the hardware and software architecture with the largest variety of products available that are suitable for use in control applications, and this was a major factor in the choice of architecture.

Setup, configuration and maintenance of a large system like this was in itself a major task throughout the year. In addition, several devices were added and features upgraded. A math co-processor was added to speed up floating point math operations, such as those used in the vision and sensing requirements. A 320 Mb (formatted) ESDI hard drive was added as an additional UNIX file system, bringing total disk capacity available for UNIX to over 400 Mb formatted. A tape drive was added for backing up the UNIX operating system and user files, and a standby power supply was attached to the system to prevent damage to the UNIX filesystems in the event of an unscheduled power outage. The Open Desktop Server Upgrade was installed to provide support for more simultaneous users. An additional 4Mb of DRAM was added, bringing the total to 12Mb, to support this. Also, a four-port serial interface card was added to support extra terminals and I/O. All of these upgrades and additions required a substantial amount of work to install, test and maintain.

The Robot

The robot is a GMF S-700 robot with the KAREL R-H controller and the Insight(TM) vision system. In addition, we have acquired several additional tools for using the robot in conjunction with a DOS based PC. The OLPC package allows off-line programming in the robot's KAREL language, file transfer to and from the robot's bubble memory file storage devices, and monitoring of the robot's status. The KCS-PC package is a set of C language callable DOS routines for transferring files, variable, and status information between the KAREL controller and a DOS program. The package also allows a DOS program to assume control of the robot by starting and stopping programs, issuing commands, etc.

The host computer was originally configured with MS-DOS 3.3 operating in a 40 Mb partition on the first hard drive. The OLPC package, which arrived first, was used to support software development (in KAREL) of the original seam alignment and wrinkle detection vision experiments using the Insight(TM) system. Later, the file transfer mechanisms were used to transfer images acquired with the GMF to DOS files. When UNIX was added to the host computer, the DOS under UNIX feature was employed to transfer the images to UNIX on the host computer and other UNIX workstations (using the network connections) for further research.

The KCS-PC package was used to communicate the results of the initial seam detection software running under KAREL back to the host computer to perform the actual seam alignment. As the vision tasks are now being moved from the KAREL controller to the host computer, this function for the KCS-PC link will no longer be necessary. KCS-PC will be employed, initially with DOS under UNIX, for the control software to manage the GMF robot. Work has been progressing to determine the best way to port the KCS-PC functions to run directly under UNIX. OLPC and KCS-PC use part of the MAP suite of protocols (MMFS) running on top of DDCMP (a Digital Equipment Corp. serial protocol that is widely used) to perform communications with the KAREL controller. All control functions that are currently implemented with DOS under UNIX are being ported to "pure" UNIX to improve performance and increase portability.

The Press

The press's controller is a very simple dedicated digital sequencer. It's built-in "intelligence" or computational power is extremely limited, and it has no provision for input signals from sensors that may need to be added or from other components in the system. Since it only performs the function of turning on and off various relays and solenoids in the press in a simple programmed sequence, it was decided to bypass this controller directly and replace its functionality with some of the available digital I/O channels in the GMF robot. This approach was chosen so as to obviate the immediate need for performing low-level I/O operations in the host computer directly. It is more efficient for the host computer to communicate with "intelligent" devices, thus reducing its computational, real-time burden. Also, the operation of the press is most closely tied-in with that of the robot, so it was a logical choice for assuming the responsibilities of press controller.

The Seam Alignment Device

The seam alignment device is actuated by one solenoid valve operated air cylinder and one stepper motor. The solenoid valve is controlled by a digital I/O line from the robot controller. Under most experiments to date, its operation, which grabs or releases the trouser leg, is controlled by the robot program. The KCS-PC package, however, allows direct operation of the robot I/O ports from the host computer as well. The stepper motor is controlled by an intelligent control card in the host computer. Its driver software is provided by the manufacturer to run under DOS, and we have been using it first with stand-alone DOS, then DOS under UNIX. Efforts to port the driver for this card to UNIX have been underway.

The Trousers Transfer Gripper

This device, which is used to transfer the trousers leg from the seam alignment device to the press, is actuated by two solenoid valve operated air cylinders and one servo motor. The valves are controlled by robot I/O, and since the gripper is attached to robot, they are always under control of the robot program. The servo motor is controlled by a stand-alone, intelligent motor controller that interfaces with the host computer over an RS-232C serial communications line. The KCS-PC package is being used now, pending full implementation of the integration and control software, to coordinate control between the robot and the gripper.

The Vision System

The vision system is used for both detection of the seam for the alignment device and for wrinkle detection on the trousers legs once they are placed on the buck. For initial investigations, the Insight$^{(TM)}$ vision system was used. However, this system is not user programmable at the lowest levels. One is limited to a package of routines that are extensions the regular KAREL programming language. This proved to be a limitation as more progress was made, and for a while, the Insight$^{(TM)}$ system was used as basically only a frame grabber. Images were transferred to the UNIX system and processing was accomplished on the host computer or on other UNIX workstations. This approach is not suitable for extended or extensive use because of the long transfer times required to send a file the size of an entire image.

To eliminate this bottleneck, we have acquired a frame grabber for use in the UNIX host computer, but we are temporarily restricted from using it due to hardware constraints. At this time, the frame grabber is being used in a stand-alone DOS PC that is separate from the host computer. Images are transferred by floppy disk to the UNIX host computer for processing.

Conclusion

The use of the IBM compatible ISA architecture machine running first DOS, and then UNIX with DOS compatibility, has proven to be a tremendous benefit. It has allowed us to realize some preliminary forms of operational control software, mostly using DOS, while work on setting up all the resources for UNIX proceeded in parallel. Had a non DOS and ISA compatible UNIX system been chosen, efforts to establish a base of resources for the host computer and control software would have hampered progress in other areas of the project.

Proceeding from experience gained using DOS for the preliminary control software, we are proceeding apace with a full UNIX based implementation of the control software. This will allow rapid and painless interface of the control software with the developing integration software.

2 4.3 <u>Wrinkle Sweeping</u>

2.4.3.1 Non-Contact Wrinkle Sweeping

Introduction

One function that may be necessary in the sequence of operations that defines the automated pressing system is removing wrinkles in the trousers after they have been loaded onto the press by the transfer mechanism. With some luck this function may be unnecessary, but until that proves to be true we are progressing with experiments with two different types of wrinkle removal devices: direct contact and non-contact devices. The direct contact device is discussed elsewhere. The non-contact device involves the use of directed compressed air to blow the wrinkles out of the trousers. The device at this time is attached to the robot as an end-effector after the trousers have been placed on the steam press buck and is moved by the robot in a pattern that should "sweep" the wrinkles from the trousers.

Previous Work

There was no evidence of previous work of this type although the use of compressed air for movement of cloth in this industry is widespread.

Description

The Gantt chart which was the outline of projected work for the Automated Garment Handling project makes no mention of non-contact smoothing but at some point during the summer it was determined by the group that it would be interesting to look at directed compressed air as a means of removing possible wrinkles in the trousers once placed on the buck. Work on this device started back in September and has progressed through the Fall. Experiments are still being conducted with the device to determine whether or not it will be possible to use some type of compressed air based smoothing device.

The non-contact wrinkle remover (sweeper) in its current form is shown in figure 53. It combines the function of the sweeper with the holding device for the light line and camera which are used to detect wrinkles as described in the sensing section. The wrinkle sweeper and detector is designed currently to sweep wrinkles from the crotch down to the cuff and then follow behind with the camera and light line apparatus to check to make sure the trousers are either wrinkle free or that any existing wrinkles are too small to matter. There has been some discussion that the direction of the sweeping should be changed from cuff to crotch. This could be done with a relatively straightforward rebuilding of the device, but that has not yet been tried.

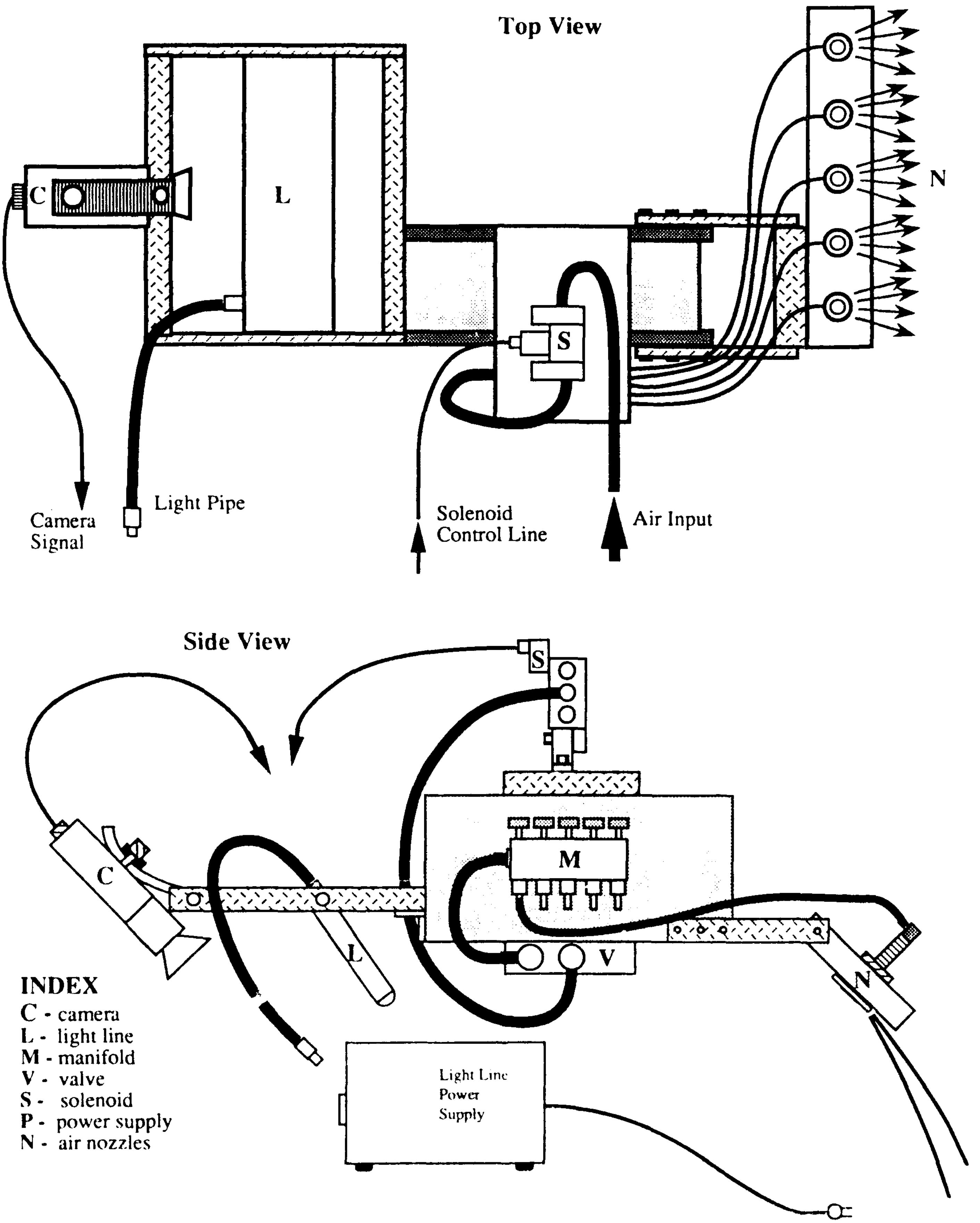


Figure 53: Side and Top Views of Non-Contact Wrinkle Sweeper and Detector

The device is based on using a commercially available air-jet system from Italy used primarily in the sewing industry. That system consists of a valve **V** which intakes compressed air and directs it to a manifold **M** which splits the air supply into five paths with manually adjusted flow rates. These five paths lead to five air nozzles **N** which direct the air onto the surface of the press. The air exits these nozzles in a relatively flat stream with a spread of approximately 80 degrees each. The nozzles are arranged such that the flows overlap and the resulting sheet of air flow is wide enough to span the width of the trousers on the buck.

The flow through each nozzle is adjustable separately but only by manually turning the adjustment knobs on the manifold, and thus there is no easy way yet of remotely adjusting the flow through each nozzle. The plate that the nozzles are mounted on is manually adjustable in attack angle so that the air flow which sweeps down to the trousers on the buck can be experimented with until the sweeping action works properly. The solenoid **S** is used to turn on and off the compressed air supply so that the flow through the nozzles can be pulsed.

Experiments with the non-contact sweeper have shown several problems. At first the flow rate through the nozzles was too low to adequately move the cloth. The diameter of the air supply lines was increased and the flow increased due to the decreased loss in pressure head in the system, but the flow is perhaps still too low. A higher velocity nozzle which pushed air in a stream was tried in addition to the other five nozzles but it was actually too strong and tended to push the cloth about indiscriminately. The flat streams of air from the other five nozzles seem to cause less havoc with the cloth, but are borderline on their ability to overcome the effects of heavier cloth and the vacuum being applied by the press. Also, since the press vacuum is probably under powered at present this problem may become exacerbated in the future. The placement of the nozzles on the plate which determines the distance from the cloth on the buck is also a parameter which needs to be experimented with in order to get the proper air flow pattern.

Current experiments are concentrating on ways to increase the flow through the flat stream nozzles, combining the effects of the high velocity but uncontrolled nozzle with the five lower velocity flat stream nozzles, and the effects of pulsing either or both of these types of nozzles on and off repeatedly. The flow through the flat stream nozzles can be increased by raising the inlet air pressure or by physically modifying the dimensions of the jet itself. Increasing the pressure would involve adding capacitive air storage tanks and a pressure regulator since we currently feed the compressed air directly from the building supply. Modifying the actual nozzles at this time is not very attractive because they are the only ones that we have at the moment. Also, it is unclear whether modifying the nozzles might result in creating turbulent flow rather than the laminar flow which currently exists.

Combining the two type of nozzles looks promising because we might be able to initiate movement in the cloth with the higher velocity nozzle and then control the placement of the cloth with the flat stream nozzles. The option of pulsing either one of these or both is attractive since using impulses in force often works better at creating movement by overcoming the friction rather than overpowering it with so much force that the motion subsequently becomes uncontrollable. If the compartmentalized vacuum system in the steam press buck which has been discussed in the past as an option materializes as a reality, the effectiveness of this device should increase significantly since it will be possible to "lock" down the part of the trousers which have already been smoothed and allow the part which has not been smoothed to move more freely.

One difficulty with this approach is its limited ability to adapt to different conditions. Without adding a vast amount of complexity to the system there is no way to account for different fabric characteristics or environmental conditions such as humidity. Thus the wrinkle sweeper in its current form would use no feedback from any sensors and would perform the same motions all the time regardless of whether there were wrinkles present or not. Whether it is worth adding the levels of complexity necessary to deal with the different information available and the different conditions possible depends on how the other parts of the system perform. For instance, whether or not the transfer mechanism reliably places the trousers on the buck in a wrinkle free condition and whether or not the contact smoothing device may work better or more simply will determine in part the future of this non-contact smoothing device.

Conclusion

The work on the non-contact smoothing has not proved yet that the idea is viable, but neither has that possibility been eliminated. The variability in the effectiveness of the device over different ranges of operating conditions is of concern but it seems likely that these problems can be resolved. The resolution may have to come at the expense of increased complexity and cost, however, and the direct contact smoother may prove to be a cheaper alternative.

2.4.3.2 Contact Wrinkle Sweeping

Description

A contact wrinkle sweeping device protoype was designed and constructed at the end of year 1 to provide a back-up method to the non-contact device described in the previous section. A pair of these devices will be used to emulate the motions used by the human operator, which is basically a diagonal sweeping motion from the center of the trouser leg. The contact wrinkle sweeping device (figure 54) consists of a set of brushes mounted on a pneumatically actuated slide. During extension of the cylinder the brushes contact the fabric to smooth the fabric. During the retract stroke the brushes are lifted off the fabric.

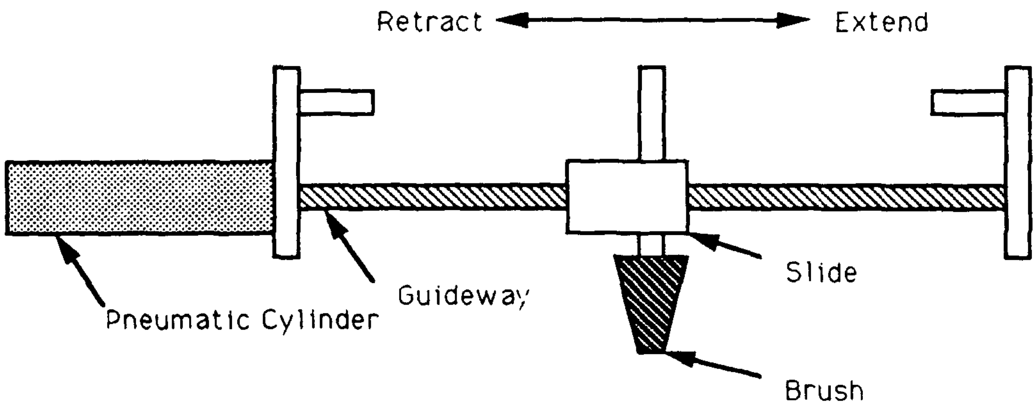


Figure 54: Contact Wrinkle Sweeper

3.0 SECOND YEAR RESEARCH PLANS

3.1 MODELING

Goals and Objectives

During the second year, we will work towards three new goals that will build directly on progress made during the first year. The emphasis will be on making these techniques more robust, physically realistic, and useful in a practical setting. The new goals will be 1) developing a methodology that will allow us to automatically determine reasonable parameter settings for the general model so that it can be used to represent specific fabric and weave types, 2) using this methodology to build up a "catalog" of parameter settings for a range of fabrics, 3) develop and investigate the particle interactions and geometric structures necessary to model the draping and puckering behavior of trouser legs, 4) developing tools that will allow engineers working on both automation and sensing to utilize our model as part of their design process.

The work on automating the process of parameter setting is designed to allow us to take our general model and "tune" it so that it can be used to represent fabrics under realistic conditions. The process will involve taking results of tests done on actual materials, and adjusting the model so that it will mimic the results of these tests. Target material characteristics to match include such difficult properties to quantify such as drape, wrinkling, crease retention, the action of hydrophilic and hydrophobic fibers, weaves and finishes of yarns and fabrics, as well as more traditional material properties, such as Young's Modulus and Poisson's Ratio. Young's Modulus is the slope of the stress-strain of a material in the elastic region. Poisson's ratio is the ratio of the lateral unit deformation to the longitudinal unit deformation of a material under tension or compression. This will serve to both verify the correctness of the underlying model, and make the model of practical use in representing real fabrics.

A direct result of the parameter setting work will be the production of a small catalog of parameter settings that can be used to represent a range of actual fabrics. This result will be key in making the model useful to engineers, since the catalog will provide a compact quantification of the properties of various fabrics within the context of a simple theory of fabric behavior that is encapsulated in the structure of the model.

The modeling of trouser legs poses difficult problems during simulation. The first problem involves capturing the non-regular geometric tubular structure of a trouser leg. The second involves modeling the mechanical behavior of the seams of the leg. In order to solve these problems we will extend the geometric aspects of our basic model to represent the non-regular tubular structure of trouser legs. We will also develop new particle relationships that will model the complex interactions occurring along the trouser leg seams.

Finally, to allow the integration of the model into the design process for the automation control systems, we will imbed a version of the model within a set of software tools that can be readily used by the other members of the project team. This will be of great assistance in the design and testing of control strategies for automation equipment, providing a cost effective means for evaluating and modifying designs before committing to the construction of actual hardware.

Besides our main goals, and as a demonstration task during the second year, we will produce a series of animated film segments showing different types of fabric and illustrating their unique dynamic characteristics and how these characteristics impact handling. A film demonstrating the draping and puckering of trouser legs will also be produced.

3.2 SENSING

Objective

The second year sensing activities focus on the redesign of the sensors to reflect the knowledge which is a result of the first year program. The redesign is necessary to move the program toward a configuration that will facilitate the transfer of this capability to the vendor community during the third year of the research program. The software associated with the redesigned sensors is alsc to be rewritten to incorporate the refined hardware operation and to enhance the integration of the sensing capability into the overall automated garment handling and pressing system. In addition, extensions to the class of materials which are handled will be explored to broaden the range of applicability of this technology.

Sensing Requirements

Seam alignment is performed external to the press on each leg in a hanging orientation. There are at least two places where the seam alignment is to be accomplished, at the hem end, and above the knee (in some instances, to within two inches of the crotch region) of the trouser leg. At the hem end, seam location is integrated with an internal seam alignment mechanism. At the above knee position, the seam alignment can be accomplished with either internal or external mechanisms. Current research is in progress to address these issues.

The detection of wrinkles and other departures from the desired surface contours is performed when the trousers are on the pressing form. If the placement on the form is sufficently consistent, only a surface smoothness validation check is required. If the placement produces variable wrinkles, puckers, or other departures from the nominal surface, then active smoothing requires that the detected wrinkles be described in terms of location, orientation, height and length.

Sensors

Three fundamental sensors have been identified during the first year: reflected light sensors for seam location, a transmitted light sensor for both seam and selvage location, and reflected light sensors for wrinkle detection. These sensors are to be further refined and developed during the second year.

It is desirable to employ the most simple and robust sensing techniques to accomplish the sensing requirements. It is clear that reflected light sensors are easier to use than transmitted light sensors because reflected light sensors can be completely external to the trousers and do not require the introduction of lighting or sensing devices inside the trousers as do transmitted light sensors. Hence the emphasis on reflected light techniques in this research. However, the results to date do not support the notion that reflected light sensing alone can locate the seams for all the kinds of materials which are being considered. In particular, there is some difficulty with the reliable detection of the seam location for some samples which have pronounced visual or geometric textures. Therefore, it is prudent to continue to consider transmitted light techniques as a backup to the reflected light seam location techniques. Further, it serves as a redundant capability to provide more robust performance in the event of unforeseen difficulties.

Reflected Light Sensor for Seam Location

By analyzing the reflected light from the surface of the material in the vicinity of a seam the seam can be located. This is done for both the in and outseams at both the hem and above the knee for both legs. The location information is passed to the seam alignment mechanism to bring the seams into alignment. The location of the seam is especially important in the above knee region of the trouser leg. This is because the selvage material is not usually distributed in a symmetric fashion relative to the seam. Hence the seam location is not guaranteed to be predictably related to the midline of the selvage boundary; but must be located by direct means. This reflected light sensor will be the preferred method for seam location.

Transmitted Light Sensor for Seam and Selvage Location

A second generation, two-sided light and viewing setup will be built and integrated into the seam alignment hardware and control. This sensor locates the boundaries of the double-ply region of the selvage portion of busted seams. The sensor detects the light that passes through the material. For most fabrics, the amount of light that shines through a single ply is reliably more intense than when there are two plies; generally more than one ply of material will substantially block the transmission of light.

The selvage boundary can be determined by a transmitted light sensor. The introduction of a transmitted light sensor element into the upper leg requires another mechanism to transport it to the proper position. It would be desirable to eliminate the need for this mechanism. Since the seam alignment mechanism is inserted into the leg opening, both transmitted and reflected light sensing can be used.

This sensor can also be used to verify the location of the seam relative to the selvage boundaries by detecting the light that shines through the sewn seam when the trouser leg is put under spreading tension as it undergoes seam alignment manipulation. However, in a small fraction of the cases, the seam alignment operation does not always result in a detectable amount of light shining through the seam. Thus, a reflected light sensor as described above is employed as the primary seam location means, with transmitted light sensing used in cases where reflected light does not work.

The mechanical redesign of this sensor is for the purpose of integrating it more completely into the seam alignment mechanism. The software redesign is for the purpose of incorporating the perspective gained from the first year and to provide for ease of integration into the overall system.

Reflected Light Sensor for Wrinkle Detection

The detection of wrinkles and other departures from the desired surface contours is performed when the trousers are placed on the pressing form. Wrinkle detection will likely employ a laser light source, probably in the near infrared, and the light pattern may possibly be obtained by passive holographic means rather than by an active scanner.

The reflected light sensor analyzes projected light patterns when the trousers are on the pressing form. The analysis of the distortion of the patterns from their nominal shape allows the determination of the presence, location, height, and orientation of puckers, creases, wrinkles and other such departures from the desired surface contour.

The two modes of application of the reflected light sensor are associated with operations which can be performed simultaneously in the integrated system. Thus, it is necessary to actually have two distinct reflected light sensors, one for seam location and the other for wrinkle detection.

The light pattern projector needs to be redesigned to obtain a more robust and less complex means of extracting the wrinkle data. Associated software to accommodate the new design will also have to be written. This effort is an extension and refinement of the year one results.

3.3 GRASPING AND MANIPULATION

The objective of year 2 of this project would be the design and fabrication of an **integrated** automated machine that would:

1] Align the trouser legs at an alignment fixture.
2] Transfer the trousers while keeping them aligned and taut.
3] Place the trouser legs without wrinkles onto the press.
4] Clear mechanisms from press as required.
5] Remove the trouser legs when pressed.

This sequence is shown in Figures 55.1, 55.2, 55.3a, 55.3b, 55.4.

Alignment Fixture

The alignment fixture would be based on the currently developed prototype model. The trousers would be rotated on rollers while the seams would be sensed by the vision system. A second set of rollers may be needed for the thigh area if year 1 research dictates.

Transfer Device

The transfer device would function as 4 or more hands gripping the aligned trouser legs. Figure 55.3a shows the gripper with internal sliding gripper pads that will produce a three hand gripper per leg effect. It appears that only two hands are required per trouser leg from work done during year 1. It would be modifiable for different size trousers. The device would rotate and translate the trouser leg to the press. This device would also remove the completed pressed leg.

It is planned that the GMF robot will be used as the transfer device during the first four months of 1991. At that time, a systematic comparison of a robot vs. hard automation would be addressed. The comparison criteria would consist of the issues listed in the Integration section of this proposal. The current concern about automation found in the garment industry today must be reconciled. If hard automation is selected, it would consist of slides, pneumatic cylinders and limit switches as opposed to a robot.

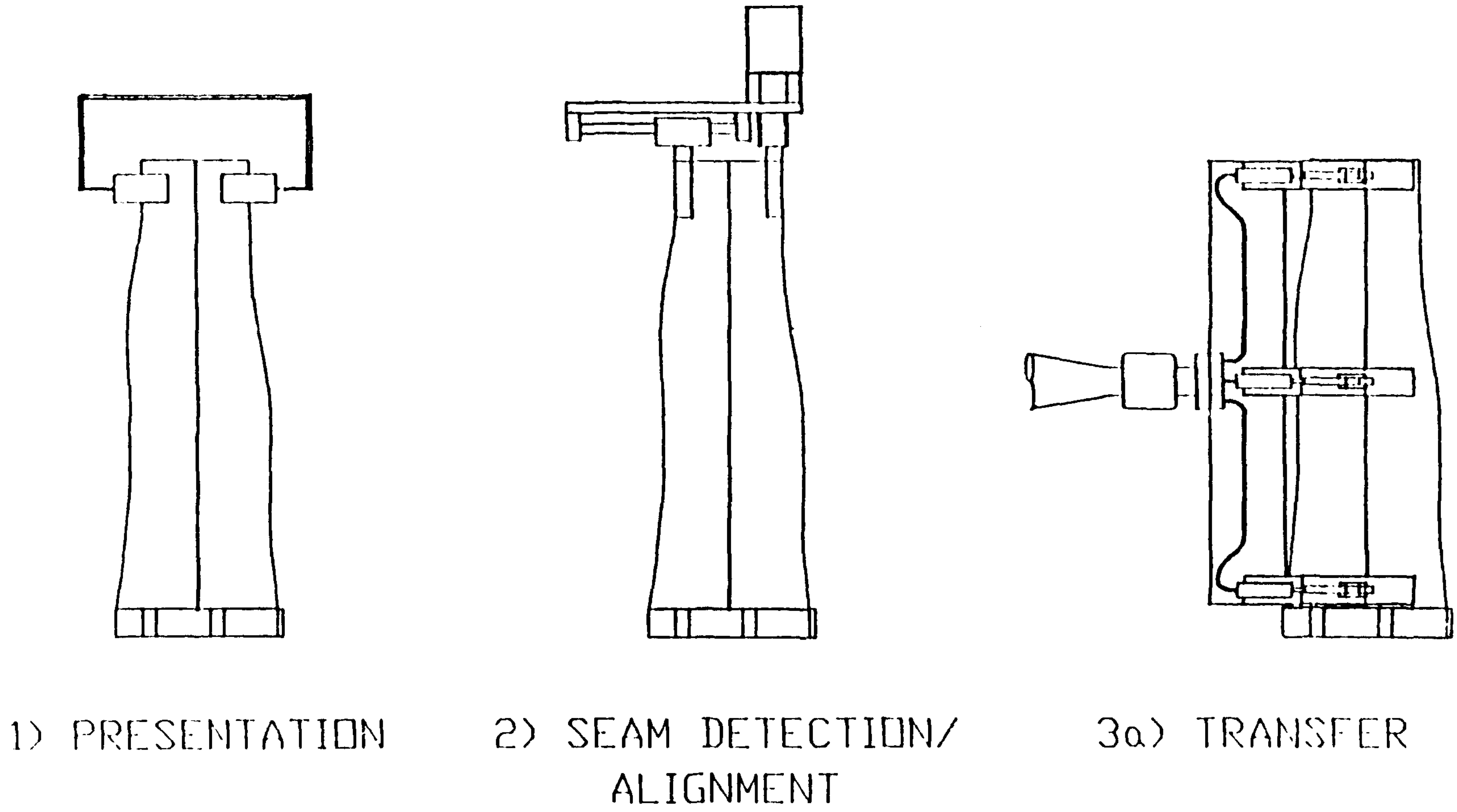
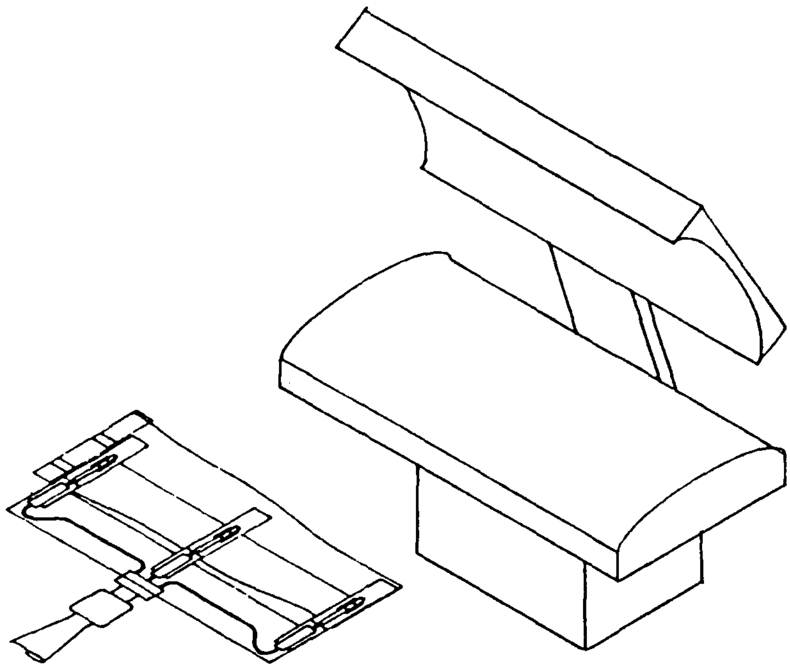
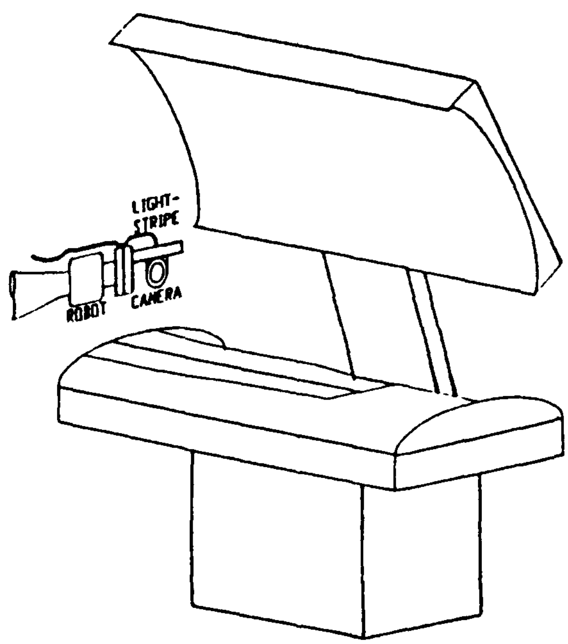


Figure 55 Grasping and Manipulation Sequence



3b) LOADING



*4) WRINKLE DETECTION/
ELIMINATION*

Figure 55 Grasping and Manipulation Sequence

Developing a Working Integrated Prototype

The main goal of year 2 would be the integration of the Alignment Fixture and Transfer Device. RPI is proficient at conceptual and preliminary design, but industrial designers are more experienced at detailed design and fabrication of manufacturing systems. It is proposed that engineers from our industrial partners, and special consultants be utilized during design reviews to bring the integrated prototypes to fruition.

3.4 INTEGRATION

Integration Tasks for Year 2

In the second year of the project, we propose to integrate these subsystems into a working prototype which will demonstrate planning and execution of coordinated functions of sensing and manipulation for garment pressing. This demonstration system will be based on continued development around the host computer hardware and software system and communications developed during the first year. The goal of this effort is to obtain the maximum capabilities from the system based on extensions to the existing host computer hardware and software. This approach has the advantage of maintaining a principal component of the system as a standard and well-supported commercial product, but has the disadvantage that some modes of control which would be required for high speed sensor-based operation are not available. We plan to overcome this difficulty by placing critical sensor-based control functions within the specialized transfer and handling mechanisms which are being designed and integrated directly with the host computer. This approach considers the robot arm as a primary transfer mechanism of the process during the early part of year 2, but separates it from some of the critical sensing and manipulation which is done by special purpose subsystems.

A commercially available robot has been employed as a transfer mechanism during the early research. This quickly provided the researchers with a flexible platform for the exploration of automation concepts. However, for several reasons, a simpler hard automation system may be the preferable transfer mechanism in an industrial setting. During the first phase of year 2 the team will compare two alternative concepts to perform the transfer function. This comparison will consider a number of objective and subjective criteria, including the following:

- capital costs
- maintenance
- flexibility requirements
- stability of vendor corporation
- experience of vendor in manufacturing and supporting automation
- requisite programming/operation support
- requisite maintenance support
- impact on pressing quality

By building on the same basic host computing environment from the first year, we feel that the integration effort will be efficient and make cost-effective use of existing resources. The principal effort will be in the continued development of the software architecture and communications protocols, and the development and implementation of real-time control algorithms which will permit efficient execution of integrated functions. Results of the flexible materials modeling activity will enter into this implementation in the form of parameterized materials characterization which is used to set control modes and parameters.

3.5 BETA TEST SITE SELECTION AND COMMERCIALIZATION

This second phase of the project will begin to bring us from pure research to the beginnings of commercialization. There are generally accepted industrial marketing techniques that we will utilize in developing suggested marketing strategies for the technology. The following section provides a framework for that strategy.

3.5.1 The Industrial Marketing Environment

3.5.2 The Industrial Marketing System

Marketing is defined as, "the process of planning and executing the conception, pricing, promotion, and distribution of ideas, goods, and services to create exchanges that satisfy individual and organizational objectives". (1)

The Marketing concept proposes 3 basic ideas:

1) Companies should produce only what industrial customers want.
2) Management must integrate all company activities to develop programs to satisfy those wants.
3) Long-range profit goals rather than "quick" sales should guide management decisions.

The principal differences that set industrial marketing apart from consumer goods marketing are who the customer is and what is the end use of the product. Industrial Marketing is sometimes called Business-to-Business Marketing because the target market is not an individual consumer but rather a business organization. This business organization uses the good or service directly or indirectly in their own operations. Many products are developed and sold to be used in the creation of consumer goods or other industrial products. Others become part of other products.

This project involves developing a system that will be sold to apparel manufacturers to be directly involved in the production of trousers.

3.5.3 Industrial Marketing Management

The functions of Industrial Marketing Management are defined as the analysis, planning, implementation, and control of programs designed to create, build, and maintain mutually beneficial exchanges and relationships with target markets for the purpose of achieving organizational objectives. (2)

3.5.4 Target Market

The targets for industrial marketing are industrial customers. These customers can be classified into three sometimes overlapping groups:

1) Commercial enterprises buying goods and services.
2) Governmental organizations purchasing goods and services.
3) Institutional customers in the market for various goods and services.

Trouser manufacturers which are our target market, are usually classified as user customers in the commercial enterprise. A user customer purchases goods and/or services for use in producing other services, which are then sold to the industrial or consumer market or both.

3.5.5 The Industrial Product

Industrial goods that user customers purchase are also classified into sections:

1) Raw Materials
2) Installations
3) Supplies
4) Manufactured materials and parts
5) Accessory equipment
6) Business Services
7) Processed materials

The Automatic Press falls under the classification of Installations. The purchase of installations is a major decision and sales negotiations usually take place over long periods of time because machinery sometimes has to be made to engineering specifications. The purchaser of installation products are normally user customers, but sometimes a fabricator may purchase several installations, integrate them into one package, and then sell this package to a user customer.

3.5.6 Industrial Customer and Market Behavior

3.5.7 The Standard Industrial Classification (SIC) System

A useful tool in Industrial Marketing is the Standard Industrial Classification, commonly referred to as SIC. The SIC system is the standard that underlies all establishment - based federal economic statistics classified by industry. (3) These classifications can be obtained from a number of private and government sources. The SIC codes including men's and boy's pants are 2311 (Men's and Boy's Suits and Coats) and 2325 or 2327 (Men's and Boy's Trouser and Slacks). A possible secondary market would include 2337 Women's, Misses, and Juniors Suits, Coats, and Skirts. (Table 10)

Employment and Number of Establishments in the Apparel Industry 1984

SIC Code	Industry Description	1984 Number of Employees	Number of Establishments: Total	Under 20 Employees	20–49 Employees	50 or more Employees	Employees Per Establishment
23	Apparel & Related Products	1,192,578	22,948	12,579	4,549	5,820	52
231	M&B Suits & Coats	70,084	433	144	54	235	162
232	M&B Furnishings	308,937	2,279	552	362	1,365	136
2321	Shirts & Nightwear	93,606	659	136	106	417	142
2322	Underwear	11,502	63	13	10	40	183
2323	Neckwear	7,403	164	73	43	48	45
2327	Separate Trousers	54,443	317	77	34	206	172
2328	Work Clothing	95,629	504	60	47	397	190
2329	Other Clothing N E C	46,104	556	181	120	255	83
233	WM&J Outerwear	383,783	8,150	3,726	2,196	2,228	47
2331	Blouses	81,848	1,531	612	376	543	53
2335	Dresses	127,275	3,799	1,949	1,141	709	34
2337	Suits, Coats & Skirts	63,882	1,204	522	294	388	53
2339	Outerwear N E C	107,324	1,440	509	357	574	75
234	W&C Undergarments	82,174	683	190	125	368	120
2341	W&C Underwear	69,022	545	141	103	301	127
2342	Corsets & Allied Garments	13,133	136	47	22	67	97
235	Millinery, Hats & Caps	16,610	403	223	96	84	41
2351	Millinery	2,710	89	45	32	12	30
2352	Hats & Caps	13,331	261	131	61	69	51
236	C&I Outerwear	71,107	847	264	199	384	84
2361	Dresses & Blouses	35,877	479	145	123	211	75
2363	Coats & Suits	4,448	66	28	12	26	67
2369	Outerwear N E C	30,746	297	86	64	147	104
237	Fur Goods	2,175	466	446	16	4	5
238	Miscellaneous Apparel	49,105	1,082	568	256	258	45
2381	Fabric Dress & Work Gloves	6,177	88	24	25	39	70
2384	Robes & Dressing Gowns	8,545	102	34	28	40	84
2385	Waterproof Outer Garments	10,529	102	28	30	44	103
2386	Leather & Sheeplined Apparel	3,335	161	112	30	19	21
2387	Apparel Belts	11,397	296	153	74	69	39
2389	Apparel N E C	9,122	333	217	69	47	27
239	Fabricated Textiles	186,182	6,260	4,495	953	812	30
	Knit Apparel Mills	150,923	1,477	552	371	554	102
2251	Women's Hosiery	33,209	176	50	34	92	189
2252	Hosiery N E C	33,115	393	131	109	153	84
2253	Knit Outerwear Mills	61,327	833	356	221	256	74
2254	Knit Underwear Mills	23,272	75	15	7	53	310

Table 10 Employment and Number of Establishments in the Apparel Industry, 1984

As you can see from the accompanying data, the primary market of firms with 50 or more employees is 365 in 1987. (Table 11) The secondary markets to examine would be suits, and womens' suits and dress slacks. The domestic market has flattened over recent years, so manufacturers are faced with a three prong assault of foreign imports, rising domestic labor costs and a stagnant marketplace.

The industry tends to invest a minuscule 1.2% (Table 11 and Table 12) of sales in capital equipment while expecting greater returns in productivity from direct labor.

If you look at the military requirement of 1,032,000 trousers annually (source DPSC) and civilian 128 million (Table 13), then we can estimate that there are about 1638 leggers pressing mens and boys trousers. This expands with the inclusion of ladies suits.Marketing strategies will be developed to penetrate this market as suggested in the following sections during phases II and III.

Industry and employment size class	E[1]	All establishments (no.)	All employees: Number (1,000)	All employees: Payroll (million dollars)	Production workers: Number (1,000)	Production workers: Hours (millions)	Production workers: Wages (million dollars)	Value added by manufacture (million dollars)	Cost of materials (million dollars)	Value of shipments (million dollars)	New capital expenditures (million dollars)	End-of-year inventories (million dollars)
INDUSTRY 2323, MEN'S AND BOYS' NECKWEAR												
Total	E2	142	7.4	117.0	6.2	11.0	76.8	251.4	223.6	475.6	6.2	58.6
Establishments with an average of—												
1 to 4 employees	E9	23	(Z)	6	(Z)	1	.5	1.1	1.2	2.4	(Z)	3
5 to 9 employees	E9	11	.1	1.2	.1	.1	1.0	1.9	2.0	3.9	1	5
10 to 19 employees	E2	24	3	5.8	3	5	3.7	15.0	15.6	30.9	.2	8.6
20 to 49 employees	E2	33	1.1	15.1	1.0	1.7	11.6	27.8	22.6	50.6	5	4.1
50 to 99 employees	E2	29	1.9	31.8	1.6	2.9	20.6	62.4	67.6	129.9	1.3	18.1
100 to 249 employees	E2	19	3.9	62.5	3.2	5.7	39.4	143.2	114.6	257.8	4.2	27.0
250 to 499 employees	-	2	(D)	(D)	(D)	(C)	(D)	(D)	(D)	(D)	(D)	(D)
500 to 999 employees	-	1	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)
Covered by administrative records[2]	E9	47	3	3.8	.3	.5	3.2	6.5	6.9	13.3	.2	1.8
INDUSTRY 2325, MEN'S AND BOYS' TROUSERS AND SLACKS												
Total	-	481	94.2	1 112.7	83.2	147.5	867.0	3 314.6	2 751.2	6 059.2	74.6	795.4
Establishments with an average of—												
1 to 4 employees	-	24	2	5.0	2	.2	2.2	38.3	243.0	279.4	3	21.2
5 to 9 employees	E2	20	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)
10 to 19 employees	E7	22	3	5.0	.2	.4	2.9	28.9	26.2	55.0	3	5.3
20 to 49 employees	E1	50	1.7	20.8	1.3	2.4	12.5	55.4	92.1	148.1	1.0	20.5
50 to 99 employees	E1	80	5.8	64.6	5.1	9.1	49.7	138.0	139.5	286.8	6.0	45.2
100 to 249 employees	-	157	27.3	285.4	24.6	43.9	234.2	513.5	361.5	881.1	9.5	122.2
250 to 499 employees	-	87	29.6	335.0	27.0	48.4	289.1	841.1	538.6	1 382.0	24.0	171.0
500 to 999 employees	-	37	29.3	396.9	24.8	43.1	276.4	1 701.4	1 350.3	3 026.9	33.5	410.0
1,000 to 2,499 employees	-	4	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)
Covered by administrative records[2]	E9	48	.4	5.2	.4	.6	3.6	12.8	12.6	25.4	.3	4.6
INDUSTRY 2326, MEN'S AND BOYS' WORK CLOTHING												
Total	-	256	34.0	348.6	29.9	52.3	274.0	948.4	718.3	1 669.8	18.0	218.2
Establishments with an average of—												
1 to 4 employees	E9	26	.1	1.2	.1	.1	.7	4.9	4.3	9.2	.1	1.0
5 to 9 employees	E9	14	.1	1.1	.1	.1	.8	3.6	2.6	6.2	(Z)	6
10 to 19 employees	E4	21	.3	3.9	.2	.4	2.1	12.7	20.1	32.6	.1	4.7
20 to 49 employees	E7	24	.8	8.1	8	1.2	6.5	22.5	18.2	39.0	.2	6.3
50 to 99 employees	E2	45	3.3	33.8	3.0	5.4	28.0	99.1	69.4	167.0	2.0	15.0
100 to 249 employees	-	91	14.9	151.6	13.4	23.2	122.6	345.0	309.4	656.4	6.5	95.1
250 to 499 employees	-	29	9.9	97.3	8.5	15.1	76.7	332.9	223.8	565.2	5.4	54.8
500 to 999 employees	-	6	4.6	51.7	3.8	6.8	36.6	125.6	72.5	194.1	1.5	38.9
Covered by administrative records[2]	E9	61	.5	5.6	.4	.7	4.0	19.6	16.0	35.6	.2	3.6
INDUSTRY 2329, MEN'S AND BOYS' CLOTHING, N.E.C.												
Total	-	808	50.1	595.1	43.7	79.9	449.8	1 272.7	966.2	2 208.5	34.0	473.9
Establishments with an average of—												
1 to 4 employees	E9	98	.2	2.5	.2	3	1.8	6.5	8.9	15.3	.1	2.5
5 to 9 employees	E7	62	4	5.2	4	6	3.9	11.1	10.6	21.7	.1	5.2
10 to 19 employees	E9	64	9	12.1	8	1.3	8.1	31.2	23.8	54.7	4	8.8
20 to 49 employees	E2	128	4.4	53.0	3.7	6.5	37.3	117.5	102.2	217.4	3.0	45.6
50 to 99 employees	E1	105	7.5	79.6	6.4	10.7	58.3	169.3	143.0	310.8	4.1	55.0
100 to 249 employees	E1	111	17.8	206.7	15.6	28.9	156.5	416.0	289.6	696.1	12.0	117.2
250 to 499 employees	-	33	10.4	128.2	9.0	16.7	94.0	350.9	244.1	594.0	6.5	111.7
500 to 999 employees	-	5	3.2	41.7	2.5	4.7	28.2	65.0	72.6	136.5	2.8	52.0
1,000 to 2,499 employees	-	4	5.3	68.1	5.1	10.1	61.7	103.4	73.5	162.2	4.9	76.0
Covered by administrative records[2]	E9	163	1.0	9.4	8	1.4	7.0	22.1	18.4	40.5	3	7.0

Note: For qualifications of data, see footnotes on table 1a. Data shown as a (D) are included in underscored figures above.

[1]Payroll and sales data for some small single unit companies with up to 20 employees (cutoff varied by industry) were obtained from administrative records of other Government agencies rather than from census report forms. These data were then used in conjunction with industry averages to estimate the items shown for these small establishments. This technique was also used for a small number of other establishments whose reports were not received at the time data were tabulated. The following symbols are shown for those employment-size classes where estimated data based on administrative-record data account for 10 percent or more of figures shown: E1—10 to 19 percent; E2—20 to 29 percent; E3—30 to 39 percent; E4—40 to 49 percent; E5—50 to 59 percent; E6—60 to 69 percent; E7—70 to 79 percent; E8—80 to 89 percent; E9—90 percent or more.

[2]Report forms were not mailed to small single unit companies with up to 20 employees (cutoff varied by industry). Payroll and sales data for 1987 were obtained from administrative records supplied by other agencies of the Federal Government. Those data were then used in conjunction with industry averages to estimate the items shown. Data are also included in respective employment-size classes shown.

Table 11 Industry Statistics by Employment Size, 1987

Item	Men's and boys' suits and coats (SIC 2311)	Men's and boys' shirts (SIC 2321)	Men's and boys' underwear and nightwear (SIC 2322)	Men's and boys' neckwear (SIC 2323)	Men's and boys' trousers and slacks (SIC 2325)	Men's and boys' work clothing (SIC 2326)	Men's and boys' clothing, n.e.c. (SIC 2329)
Gross book value of depreciable assets:							
Total:							
Beginning of year	348.0	448.2	85.0	36.7	726.0	176.3	307.5
New capital expenditures[1]	29.5	51.9	7.7	6.2	74.6	16.0	34.0
Used capital expenditures	4.5	4.7	.6	.3	3.2	2.6	3.7
Retirements	16.0	27.0	5.4	1.5	36.9	8.1	18.9
End of year	366.0	477.8	87.8	41.7	766.9	186.7	326.2
Buildings and other structures:							
Beginning of year	122.3	147.8	32.4	11.1	245.8	61.0	101.7
New capital expenditures	6.9	6.9	1.8	.2	18.7	1.7	8.5
Used capital expenditures	3.1	2.1	.4	.2	1.5	1.1	1.6
Retirements	3.4	7.6	1.8	.1	11.2	1.9	4.4
End of year	128.9	151.0	32.8	11.3	254.8	61.9	107.4
Machinery and equipment:							
Beginning of year	225.7	300.6	52.5	25.7	480.2	115.3	205.8
New capital expenditures[1]	22.6	43.0	5.8	6.0	55.9	14.3	25.5
Used capital expenditures	1.3	2.6	.2	.1	1.8	1.5	2.1
Retirements	12.6	19.5	3.5	1.4	25.7	6.2	14.5
End of year	237.0	326.7	55.0	30.3	512.1	124.8	218.9
Depreciation charges during 1987:							
Total	30.5	37.0	4.9	3.2	67.5	14.7	24.3
Buildings and other structures	7.7	7.1	1.4	.5	12.9	2.9	5.0
Machinery and equipment	22.8	29.9	3.6	2.8	54.5	11.8	19.4
Rental payments:							
Total	26.9	16.4	2.9	7.3	27.2	8.3	26.5
Buildings and other structures	15.8	8.3	1.2	4.3	12.9	3.9	9.5
Machinery and equipment	11.2	8.1	1.7	3.1	14.3	4.4	16.9

Note: Retirements and depreciation data for establishments not included in the ASM sample were extrapolated from the historical ratio of retirements or depreciation to assets. These ratios were developed at the industry level.

[1]Data on new machinery and equipment expenditures by type are provided in table 3c.

Table 12 Gross Book Value of Depreciable Assets, etc., 1987

Year	Men's apparel: Suits	Dress & Sport Jackets	Non-Tailored Jackets	Trousers & Slacks	Jeans & Dungarees	Dress Shirts	Woven Sport Shirts	Knit Sport Shirts	Sweaters
1967	19.5	13.2	30.8	146.0	89.2	123.4	150.6	77.3	39.9
1970	17.7	11.8	33.5	173.6	135.3	145.3	104.2	94.3	38.6
1971	16.5	14.4	32.2	183.7	148.5	145.8	103.8	120.0	42.2
1972	18.7	21.3	37.7	166.6	177.3	158.3	105.7	159.0	45.5
1973	16.7	21.3	44.2	171.1	186.7	137.8	113.4	171.2	53.1
1974	16.8	18.6	48.6	156.6	181.6	125.1	112.3	200.6	55.5
1975	13.7	11.5	47.0	118.9	163.0	95.4	113.7	182.7	39.9
1976	15.7	12.7	45.3	127.0	198.1	110.4	124.5	243.6	36.6
1977	17.3	16.6	44.5	128.8	213.3	110.7	107.8	293.4	39.4
1978	17.5	15.0	37.1	122.7	209.6	97.5	82.6	305.4	37.5
1979	16.2	15.7	41.5	125.6	231.8	95.0	84.5	291.6	33.7
1980	13.9	16.8	39.2	125.1	224.8	88.6	69.0	320.3	33.4
1981	13.9	17.1	40.9	118.2	191.9	93.5	63.5	374.3	39.3
1982	11.5	18.0	44.6	115.1	179.9	89.8	50.0	372.1	38.8
1983	10.9	19.2	37.2	113.0	183.5	86.2	39.1	365.5	36.8
1984	12.5	19.2	36.5	123.0	179.5	81.6	42.5	354.8	30.5
1985	12.4	18.2	35.3	116.4	186.4	84.6	40.6	350.4	27.0

Year	Boy's Apparel					Men's & Boys' Underwear & Nightwear		
	Trousers & Slacks	Jeans & Dungarees	Woven Shirts	Knit Sports Shirts	Non-Tailored Jackets	Knit Under-Shirts	Knit Shorts & Briefs	Pajamas
1967	56.0	67.2	87.6	69.2	14.8	357.0	257.9	50.3
1970	53.5	72.9	64.6	56.4	14.5	314.5	246.3	43.5
1971	50.6	80.4	60.9	56.4	14.2	338.5	252.2	40.8
1972	46.5	89.8	67.8	88.4	14.1	397.1	323.0	45.9
1973	43.8	83.1	59.7	92.9	14.1	418.2	344.5	41.0
1974	39.7	83.1	52.4	110.7	14.9	401.6	339.7	38.7
1975	34.7	74.4	55.8	107.2	13.8	370.3	288.2	40.1
1976	33.7	89.6	59.0	132.2	13.4	388.4	377.3	37.2
1977	24.9	74.1	42.5	146.1	14.7	378.4	330.0	37.9
1978	27.5	80.9	35.0	148.1	13.7	361.6	348.4	33.5
1979	20.3	80.9	40.1	139.9	10.7	358.9	376.6	37.4
1980	17.2	107.0	34.6	152.5	9.6	293.3	338.6	32.8
1981	17.1	93.8	24.7	171.0	10.7	289.5	354.6	25.9
1982	14.9	84.5	26.6	141.7	8.9	262.8	393.4	31.0
1983	15.3	89.3	16.8	147.2	8.0	210.3	462.6	27.3
1984	12.9	75.5	12.7	135.2	6.8	254.6	427.8	28.1
1985	12.5	71.9	14.0	121.1	6.5	244.8	427.5	27.1

SOURCE: Current Industrial Reports MA23a and others in the MA23 series

Table 13 U.S. Apparel Production in Selected Garment Lines

3.5.8 Industrial Marketing Uses of the SIC System and Related Data Sources

The generalized SIC codes may be further refined and used in a variety of ways in formulating marketing strategies. Some of the outcomes we will consider are:

- •Developing a marketing information system (MIS) - system for collecting marketing data.
- •Defining and segmenting target markets.
- •Determining market and sales potentials.
- •Indentifying and locating potential customers.
- •Planning sales territories and sales quotas.
- •Designing marketing research samples.
- •Building direct mail lists for advertising.
- •Determining names and positions of buying influences.
- •Developing leads for sales personnel.
- •Indentifying competitors.
- •Indentifying and locating prospective middlemen.
- •Selecting trade publications for advertising.
- •Facilitating physical distribution decisions such as warehouse size and location.
- •Measuring market share and market penetration.
- •Measuring advertising effectiveness. (4)

Looking at the distribution of firms and their geographic locale (Table 14) it is probable that the first tier of manufacturers will be personally contacted. Subsequent market penetration will probably be made through a distributor network and presentations at major trade shows.

Industry and geographic area	E[1]	1987: All establishments: Total (no.)	1987: All establishments: With 20 employees or more (no.)	1987: All employees: Number[2] (1,000)	1987: All employees: Payroll (million dollars)	1987: Production workers: Number (1,000)	1987: Production workers: Hours (millions)	1987: Production workers: Wages (million dollars)	1987: Value added by manufacture (million dollars)	1987: Cost of materials (million dollars)	1987: Value of shipments (million dollars)	1987: New capital expenditures (million dollars)	1982: All employees[3] (1,000)	1982: Value added by manufacture (million dollars)
INDUSTRY 2311, MEN'S AND BOYS' SUITS AND COATS														
United States	-	343	241	56.4	799.6	49.1	87.6	611.7	1 995.0	1 275.8	3 250.6	29.5	75.2	1 683.0
Alabama	-	7	6	1.2	11.7	1.1	1.7	10.5	18.6	5.8	24.6	(D)	1.5	23.5
Arkansas	E2	4	4	.4	3.0	.3	.7	2.7	8.8	4.4	13.1	.2	(NA)	(NA)
California	E1	26	11	EE	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	2.6	53.0
Delaware	-	1	1	AA	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	AA	(D)
Florida	E1	14	9	1.1	12.3	1.0	1.7	9.5	35.5	27.5	62.1	(D)	.6	10.6
Georgia	-	21	18	FF	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	5.8	121.6
Illinois	-	12	7	FF	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	FF	(D)
Indiana	-	1	1	.3	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	.8	13.5
Kentucky	-	9	9	FF	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	2.6	35.2
Louisiana	-	1	1	BB	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	CC	(D)
Maine	-	2	1	AA	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	CC	(D)
Maryland	-	8	7	1.6	22.4	1.5	2.5	20.0	52.2	30.0	79.4	(D)	3.1	83.6
Massachusetts	-	19	15	4.9	78.0	4.2	7.8	58.7	292.3	222.1	506.0	(D)	6.3	189.0
Minnesota	E1	3	3	AA	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	BB	(D)
Mississippi	-	6	6	EE	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	EE	(D)
Missouri	-	7	5	EE	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	CC	(D)
New Jersey	E2	22	15	1.8	23.9	1.6	2.9	17.6	46.2	27.1	78.2	.5	3.1	57.5
New Mexico	-	3	1	AA	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
New York	E1	71	36	FF	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	9.3	211.5
North Carolina	-	5	5	CC	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	EE	(D)
Ohio	-	5	4	EE	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	2.9	119.2
Pennsylvania	E1	55	44	11.9	184.5	10.5	18.2	143.6	337.6	266.8	596.5	6.0	16.1	380.3
South Carolina	-	4	4	CC	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	EE	(D)
Tennessee	-	9	8	2.9	42.0	2.6	4.8	32.6	73.9	107.3	186.3	(D)	4.1	65.0
Texas	E2	10	9	3.1	39.2	2.7	5.3	32.8	72.3	29.9	100.5	(D)	3.6	53.5
Virginia	-	2	2	CC	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	CC	(D)
INDUSTRY 2325, MEN'S AND BOYS' TROUSERS AND SLACKS														
United States	-	481	416	94.2	1 112.7	83.2	147.5	887.0	3 314.8	2 731.2	6 050.2	74.8	(NA)	(NA)
Alabama	-	40	39	10.9	115.0	9.9	16.7	98.2	229.1	186.4	423.8	5.6	(NA)	(NA)
Arkansas	-	7	7	2.0	21.6	1.8	3.2	19.4	70.8	31.0	103.0	1.6	(NA)	(NA)
California	-	47	36	3.1	38.9	2.9	5.3	30.1	93.1	268.2	357.1	2.1	(NA)	(NA)
Florida	E1	7	4	EE	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Georgia	-	61	59	10.8	125.0	9.3	18.6	89.9	285.6	255.4	540.6	7.3	(NA)	(NA)
Illinois	-	2	2	BB	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Indiana	-	7	6	EE	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Iowa	-	5	5	CC	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Kansas	-	6	5	CC	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Kentucky	E2	18	18	4.4	44.2	4.1	7.0	39.5	86.3	60.5	148.6	1.6	(NA)	(NA)
Louisiana	-	7	7	2.1	20.9	2.0	3.9	18.5	27.7	11.8	41.7	.2	(NA)	(NA)
Maryland	-	5	3	BB	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Massachusetts	-	5	3	BB	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Michigan	-	2	2	AA	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Mississippi	E1	31	29	8.0	83.4	7.6	13.2	72.9	144.0	129.6	274.7	3.8	(NA)	(NA)
Missouri	-	23	23	4.2	48.4	3.7	7.0	37.2	99.6	30.7	130.5	1.4	(NA)	(NA)
New Mexico	-	2	2	EE	(D)	(D)	(D)	(D)	(D)	(D)	(D)	.1	(NA)	(NA)
New York	E1	37	14	1.3	25.7	1.0	1.8	14.9	112.9	143.6	257.0	1.3	(NA)	(NA)
North Carolina	E1	18	17	3.8	39.9	3.6	6.0	34.3	103.5	54.1	156.6	(D)	(NA)	(NA)
Ohio	-	2	2	(AA)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Oklahoma	-	14	14	FF	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
Pennsylvania	-	33	28	3.0	35.6	2.7	5.0	29.4	50.9	29.7	82.2	1.2	(NA)	(NA)
South Carolina	-	5	5	.9	9.7	.8	1.5	7.5	33.9	11.1	43.7	.5	(NA)	(NA)
Tennessee	-	39	36	11.1	132.9	9.8	17.7	107.3	438.6	266.6	699.0	14.7	(NA)	(NA)
Texas	-	40	34	14.1	184.5	12.1	21.2	134.8	649.7	559.9	1 222.3	(D)	(NA)	(NA)
Virginia	-	10	10	FF	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)
West Virginia	-	2	2	CC	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(D)	(NA)	(NA)

Table 14 Industry Statistics for Selected States: 1987 & 1982

3.5.9 The Demand for Industrial Goods and Services

Once markets have been defined by SIC designations, it is possible to determine many factors that relate to demand. Demand that is related to market characteristics may be viewed in terms of geographic distribution, size, and number of companies or organizations, and degree of concentrations in various regions of the country. (5)

Several factors are looked at by organizational customers when purchasing installations. The installation must be able to conform to specifications, the suppliers must have a dependable service plan, and the quality and quantity of technical assistance provided at the time of installment are all taken into consideration by the customer. Other customer considerations are: product warranties, ease of operation, difficultly or ease of integration of the new product into existing facilities, compatibility of the product with the other equipment used by the buyer, and the amount of training required for the buyer's personnel to operate the product properly. These factors must be addressed thoroughly in the marketing plan and implemented in early negotiations. All plus points of the product must be clearly stated and shown to the customer such as potential labor and time savings, and financial and leasing arrangements.

Since the Robotic Press has a joint demand characteristic, compatibility with a number of different types of makes and models of presses may be important or we may determine that the technology will be best suited as an integrated system. Joint demand states the products are demanded jointly or they may not be demanded at all. If the apparel factory has a press to which integration is not possible or too complex or costly, they will not purchase the robot. The development of a Retro-fit-kit forexisting presses is another possibility. Transportation to the press is also a key to the success of this installation which must be developed and tested during the second and third phases of this project.

3.5.10 Demand Estimation

When a primary target market is chosen, the potential size and market penetration objectives should be determined. The census of manufacturers will show the number of companies, establishments, and total employees in industry and some additional financial data in the market. (Table 11) A true target market has four factors that must be met:

1) The organizations must need the product or service.
2) The organizations must have the ability to purchase.
3) The organizations must be willing to purchase.
4) There must be authority to purchase. (6)

Having an industrial partner with considerable experience in this market will help us evaluate this phase more accurately prior to phase III.

3.5.11 Channels of Distribution

The industrial channels of distribution are different than those of the consumer market. Channels in the industrial market are typically shorter. Two types of middlemen that are common in the industrial market are the industrial distributor and the manufacturers representative. (Figure 56)

The primary market chosen for the robotic press was the mens and boys trouser market. This market shows promise for potential labor savings. The robotic press is a new product being introduced into a present market. The general steps that may be useful for the planning of this new product are using existing channels or creating new channels. The process must begin with the customer establishing the need for the product. The selling organization can aid the apparel factory (customer) in recognizing the problem. The robotic press can serve existing customers through new distribution channels.

During Phase II of the project we will begin examining which method of distribution will be most effective. At this time we would anticipate that the sophistication of the end product will dictate the channels. Other factors will be: ease of installation, technical features of the press, price and payback analysis that need to be presented to prospective users.

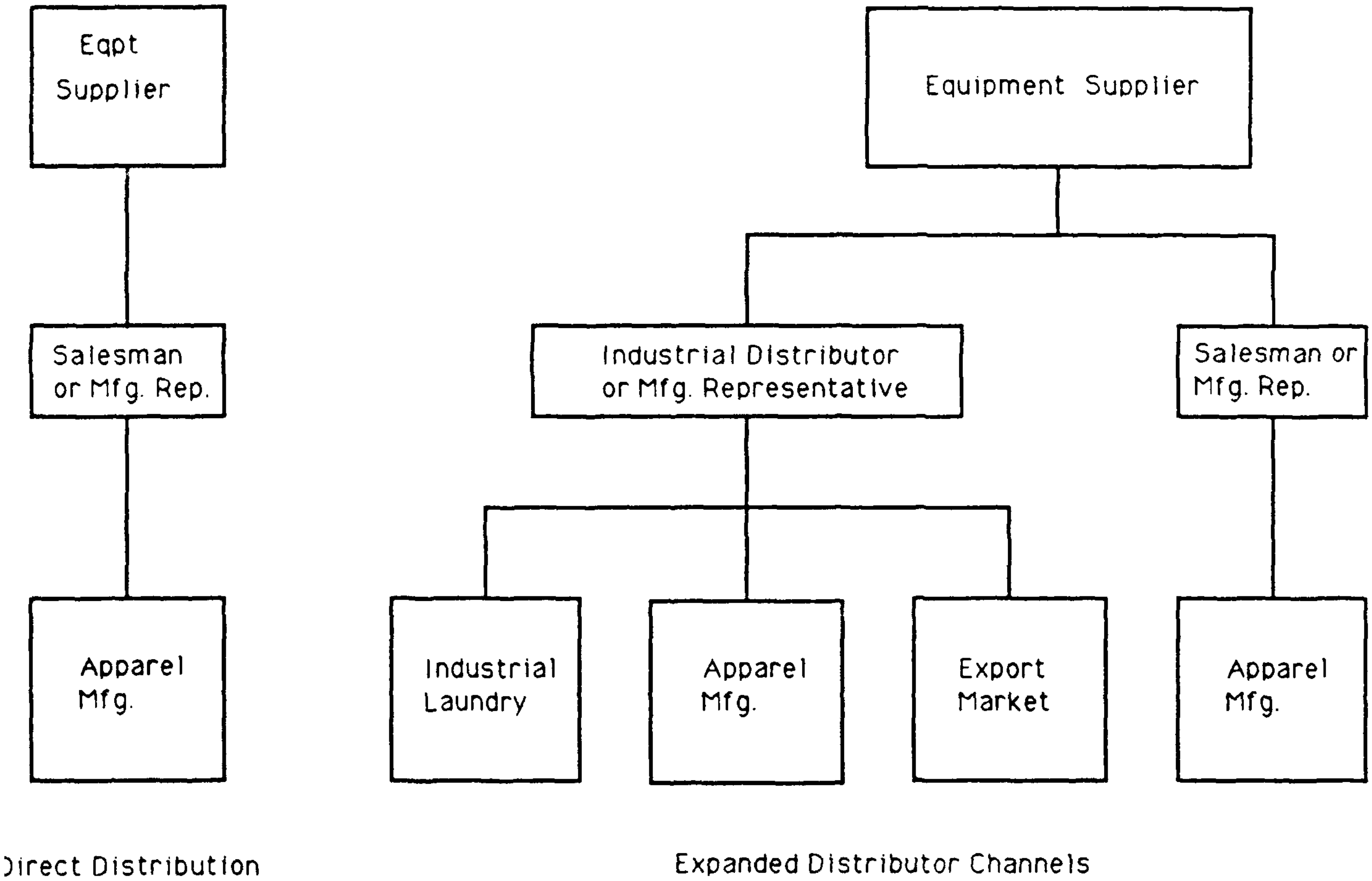


Figure 56: Channels of Distribution

3.5.12 Marketing Focus for Profits

In order for the supplier of the robotic press to obtain increased profits the higher added value for the customer must be proven. To relay this message to the customer it must be thoroughly addressed in the promotion. When promoting and pricing the robotic press, the technical risk must be compared to the potential payoff for the apparel factory. To evaluate that payoff a time limit must be chosen and a long term profit goal must be established by management. To complete the sale, the buyer and seller must negotiate for and enter agreement about selling terms. One of the prime considerations in phase III commercialization will therefore be affordability for the user.

3.5.13 The Organizational Buying Process

3.5.14 Marketing Implication in the Buying Process

Every organization has a different set of stages for its buying sequence. The industrial marketing manager must recognize that such a process exists in a different form for every organizational customer. The manager must also involve the company's marketing efforts in the process as early as possible if it is to be considered as a supplier of the product or service. (7)

If the marketing firm aids the factory in recognizing the problem of having an inefficient pressing operation they are more likely to secure the sale. When the involvement begins in the initial steps of recognizing the problem, it is easy to work the robot into the specifications of the factory's existing system. It will be more difficult for an outside company to match these needs without having been involved in establishing the specifications.

3.5.15 Decision Criteria in Industrial Buying

There are certain criteria which buyers take into consideration when deciding on a purchase. These should be a big part of the composition of the marketing mix. Ones to be addressed at this time are: performance, economic, integrative, adaptive, and legal.

3.5.15.1 Performance Criteria

Performance criteria is what should be initially addressed for the customer. This states how well the product will do the task. A good way of clarifying that point is to show a production rate comparison of the old method versus the new, as well as the consistency of finished quality.

3.5.15.2 Economic Criteria

The costs incurred by buying and using the product are economic criteria. This can be demonstrated by the more complex version of the same comparison used for performance criteria. Some specific information that is necessary to evaluate the profitability of the robot is the rate of production in a given time period as compared to the level of productivity of first quality garments in the same time frame utilizing the current method. The following chart outlines some of the critical areas that we will investigate and compare in order to prepare a thorough financial analysis at the end of Phase II of this project.

Traditional	Automated
1. Cost	1. Cost
2. Salary of operator.	2. Retro fit kit.
3. Employee benefits cost.	3. Prodution rate.

4. Production rate.	4. Percent of defects.
5. Downtime.	5. Downtime.
6. Sick time.	6. Spare parts.
7. Represses.	7. Represses.
8. Utilities.	8. Utilities.
9. Space.	9. Space.
10.Maintenance.	10. Maintenance.
11.Indirect overhead.	11. Indirect overhead.
12.Supplies.	12. Supplies.

Payback period, return on investment, and internal rate of return are computations that should be analyzed for the project. These are the most accepted forms of financial analysis used by the industry today. To be successful in this market we must be able to generate positive returns using all of these criteria.

We will also look at other non-traditional capital equipment analyses including the AMCIA approach developed by researchers from Clemson University under another DLA sponsored project. AMCIA is an example of a more sophisticated sensitivity analysis for capital equipment expenditures. Sensitivity analysis is usually recommended for cost justifying systems requiring expenditures of $500,000 or more.

3.5.15.3 Definitions and Cost Equations

Costs	Savings
Purchase price (P)	Direct, Indirect labor(L)
Accessories & equipment(A)	Material (M)
Engineering & install (I)	Tax Credit (C)
Maintenance & operation(O)	Annual Depreciation (D)

Payback Period: The payback period is a measure of the time required to recover the initial investment. Therefore, if the payback period is two years, a robot system will return net positive cash flow after that period for the remainder of the life of the system. This is a simple equation for determining the payback period:

$$\text{Payback Period} = \frac{(P+A+I)-C}{(L+M-O) \times H \times (1-TR) + D \times TR}$$

H= hours of use during the year.
TR= tax rate of the firm.

This equation may be adequate for financial analysis when the payback period is short, such as one or two years. When the period is longer, we include the time value of money by using discounted cash flows.

Return on Investment: Once a firm has ascertained its cost of capital for its new investment, it can then compare the return on investment with its cost of capital and determine if it meets or exceeds the cost. The return on investment is calculated as follows:

$$\text{Return on Investment} = \frac{S \times 100}{T} \text{ Percent}$$

S = Total annual savings ((L+M+O)x H - D)
T = Total investment (P+A+I-C)

Internal Rate of Return: The calculation of the internal rate of return (IRR) utilizes a calculation that accounts for the time value of savings realized in the future. The calculation for IRR is accomplished by calculating the present value (PV) of the stream of future cash income, annual savings (AS), or a candidate rate (R).

Total Cost Comparisons: It may be useful for a firm to compare the total costs of using a robot system with that of the normal labor procedure over a selected period, perhaps the life of the robot system or the planning horizon of the firm. For the normal labor procedure the total cost (TCL) is:

$$TCL = H \times L \times Y \qquad Y = \text{number of years.}$$

The total cost of the robot system (TCR) may be calculated as:

$$TCR = (P+A+I-C) + <(O-M) \times H - D \times TR> \times Y$$

A pertinent comprehensive discussion is presented by Graham R. Mitchell and William F. Hamilton in their May 1988 article "Managing R & D As A Strategic Option", which can be found in Research Technology Management.

3.5.16 Integrative and Adaptive Criteria

The supplier should state that the customer service will do more than just sell the product, but will become an integrated partner. The supplier will follow up on the sale with service and proper training on the operation and maintenance of the equipment. This can be a marketing strategy.

The seller and buyer should both be clear on the robot specifications, to insure they will be met. The certainty of the buying influence of this is called adaptive criteria.

3.5.17 Legal Criteria

There are many different legal criteria to be considered when buying and selling installations. These are generally classified as legal or policy considerations. The safety of workers around the machine or fire hazards are two prominent ones concerning the robot. All of these criteria including warranties, patents, copyrights, OSHA codes, Workers Compensation laws, must be carefully examined during phases II and III of this project.

3.5.18 Industrial Marketing Formulation

As we can not guarantee the commercialization of this technology we can not establish the marketing strategies to be used by our industrial partner or other parties in developing the market. We are assuming that we will be working with them in Phase III and that they will be looking at the industrial marketing model as outlined in the following section.

3.5.19 Marketing Intelligence

Marketing intelligence refers to information that can be used to enhance the competitive position of the company. This broad definition can include almost any information that can help the marketing manager make more effective decisions in marketing segmentation, overall marketing strategy formulation and marketing planning, and the marketing substrategies of product, promotion, channels, and price.(9) It is the basis of all sound marketing strategies and marketing plans. The information is obtained from internal and external sources.

3.5.19.1 Internal Marketing Intelligence

Internal marketing intelligence comes from within the company. Different departments throughout the company turn in reports to the marketing manager. For instance, a report from the production department will state the production capabilities of the facility, so that the marketing manager knows how large a demand can be filled within a certain period of time. This will also be a factor for sales forecasting. For pricing, a cost analysis can be put together by the company's accounting department and credit ratings of potential customers should be supplied by the credit department.

3.5.19.2 External Marketing Intelligence

Any type of useful information obtained from the outside sources of the company is called external marketing intelligence. This information is in the forms of feedback or research and analysis. It can include complaints of customers and middlemen on subjects ranging from the product itself, service, to packaging. Different sources of research are commonly used, i.e. trade publications, trade associations, and any form of market research survey. All of this information should lead to an adjustment in the marketing strategy and plan. Information gained from the Beta site field tests proposed in Phase III will be the biggest source of accurate data for marketing.

3.5.20 The Marketing Information System

The marketing information system is a frame work set up within the company to insure a steady flow of valid, desired information to plug into the company's strategies. This system is an organizational tool for the marketing intelligence information flowing in and through the company. We anticipate being part of that system as we transition from a research based project to possible commercialization.

Part of this analysis will be the development of the pricing strategy. A subset of pricing will be the presentation of purchase and/or lease considerations for prospective buyers which will depend upon the suggested selling price of the system and the buyer's financial structure.

3.5.21 Physical Distribution

3.5.22 Physical Distribution Activities

According to the Council of Logistics Management, physical distribution is defined as follows: A term employed in manufacturing and commerce that describes the broad range of activities concerned with the efficient movement of finished products from the end of the production line to the consumer, and in some cases including the movement of the raw materials from the source of supply to the beginning of the production line. These activities include freight transportation, warehousing, material handling, protective packaging, inventory control, plant and warehouse site selection, order processing, market forecasting, and customer service.(10)

These activities are taken over by different parts of the distribution channel. For instance, if industrial distributors are used they become an integral part of the distribution system. They typically take title of the goods involved, provide warehousing, maintain inventory stocks, and ship to customers. This leaves considerably less for the manufacturer or the industrial marketing manager to be in charge of, but the benefits and problems associated with the use of industrial distributors should be recognized.

A well thought out physical distribution plan will give the company a competitive advantage. We have seen from other technologies that were introduced in the past, that domestically the initial effort should be direct with a select group of distributors. After the product has been successfully introduced then additional distributors may be added as need requires. It is common for high technology products to be controlled by the vendor. The distributor then becomes the "door opener" and is compensated on a finders fee basis. Installation, service and spare parts may be the full responsibility of the vendor.

For export however, distributors are an absolute necessity. It is a requirement that you have full service outlets who are familiar with the local market, customs regulations, and other local economic conditions including method of payment, that are unknown to the American exporter.

3.5.23 Marketing Control and Evaluation

Once the strategies have been developed, then the path to success should be a measured one. Control systems that permit adjustments will be vital for this new technology. We will be sure to incorporate the knowledge gained in other DLA projects concerning the introduction of technology to attempt to avoid the failures uncovered in that research. This information will be passed along to the industrial partner to be used in developing control criteria.

3.5.24 Schedules and Charts

Schedules and control charts are useful for controlling functional activities, and when tied to responsibility centers, they can also be used to evaluate personnel. They are used in monitoring overall marketing plans and programs in which many separate activities must be coordinated for optimum impact. Finally, control charts can be used with quantitative standards to provide quantifiable performance standards.

3.5.25 Reports

Reports are other techniques used to evaluate and control the company. Reports are usually from within the company and the types requested are outlined by the Marketing Information System. Some common types for general company information are progress reports, activity reports, and informational reports. These are generated by each department.

3.5.26 Budgets, Sales and Costs Analyses

This financial data is used to control or discover strengths, weaknesses or unexpected patterns in marketing results or general company strategies. If there is a difference between actual and targeted costs and sales, a variance analysis must be done. This information is imperative to the company in order to make a profit. These reports may uncover product strengths or weaknesses that require adjusting the support program or product engineering.

FOOTNOTES (for section 3.5)

1. Haas, Robert W., Industrial Marketing Management, 4th edition, Boston: PWS-Kent Publishing Company, 1989, p. 204.
2. Ibid., p. 22.
3. Ibid., p. 35.
4. Ibid., p. 58.
5. Ibid., p. 64.
6. Ibid., p. 147.
7. Ibid., p. 92.
8. Ibid., p. 105.
9. Ibid., p. 177.
10. Definition used by the Council of Logistics Management, Chicago, Illinois.
11. Haas, Op. Cit., p. 410.

REFERENCES (for section 3.5)

Haas, Robert W., Industrial Marketing Management, 4th edition, Boston: PWS-Kent Publishing Company, 1989.

Hutt, Michael D. and Speh, Thomas W., Business Marketing Management, 3rd edition, New York: The Dryden Press, 1989.

Rachman, David J., Marketing Today, 2nd edition, New York: The Dryden Press, 1988.

References

[1] Amirbayat, J. and J.W.S. Hearle, "The Anatomy of Buckling of Textile Fabrics: Drape and Conformability," Journal of the Textile Institute, Vol. 80, pp. 51-69, 1989.

[2] Aono, M., "A Wrinkle Propagation Model for Cloth," Computer Graphics Around the World (CG International '90 Proceedings), eds. T.S. Chua and T.L. Kunii (Springer-Verlag, Tokyo, 1990) pp. 95-115.

[3] Bassett, R.J. and R. Postle, "Fabric Mechanical and Physical Properties, Part 4: The Fitting of Woven Fabrics to a Three-Dimensional Surface," International Journal of Clothing Science and Technology, Vol. 2, No. 1, pp. 26-31, 1990.

[4] Clapp, T.G. and H. Peng, "Buckling of Woven Fabrics, Part I: Effect of Fabric Weight," Textile Research Journal, Vol. 60, pp. 228-234, 1990.

[5] Clapp, T.G. and H. Peng, "Buckling of Woven Fabrics, Part II: Effect of Weight and Frictional Couple," Textile Research Journal, Vol. 60, pp. 285-292, 1990.

[6] Cusick, G.E., A Study of Fabric Drape, PhD Thesis, University of Manchester, 1962.

[7] de Jong, S. and R. Postle, "An Energy Analysis of Woven-Fabric Mechanics by Means of Optimal-Control Theory, Part I: Tensile Properties," Journal of the Textile Institute, Vol. 68, pp. 350-361, 1977.

[8] de Jong, S. and R. Postle, "An Energy Analysis of Woven-Fabric Mechanics by Means of Optimal-Control Theory, Part II: Pure-Bending Properties," Journal of the Textile Institute, Vol. 68, pp. 362-369, 1977.

[9] de Jong, S. and R. Postle, "A General Energy Analysis of Fabric Mechanics Using Optimal Control Theory," Textile Research Journal, Vol. 48, pp. 127-135, 1978.

[10] Feynman, C.R., Modeling the Appearance of Cloth, Masters Thesis, Massachusetts Institute of Technology, 1986.

[11] Getto, P. H., "Fast Ray Tracing of Unevaluated Constructive Solid Geometry Models," New Advances in Computer Graphics (CG International '89 Conference Proceedings), eds. R.A. Earnshaw and B. Wyvill (Springer-Verlag, Tokyo, June 1989) pp. 563-578.

[12] Getto, P.H. and D.E. Breen, "An Object-Oriented Architecture for a Computer Animation System," The Visual Computer, Vol. 6, No. 2, pp. 79-92, 1990.

[13] Haumann, D.R. and R.E. Parent, "The Behavioral Test-bed: Obtaining Complex Behavior From Simple Rules," The Visual Computer, Vol. 4, pp. 332-347, 1988.

[14] Hearle, J.W.S., P. Grosberg and S. Backer, Structural Mechanics of Fibers, Yarns, and Fabrics, Volume 1 (Wiley-Interscience, New York) 1969.

[15] Hearle, J.W.S. and W.J. Shanahan, "An Energy Method for Calculations in Fabric Mechanics, Part I: Principles of the Method," Journal of the Textile Institute, Vol. 69, pp. 81-91, 1978.

[16] Hillis, W.D., The Connection Machine (The MIT Press, Cambridge, MA) 1985.

[17] House, D.H. and D.E. Breen, "Particles As Modeling Primitives For Surgical Simulation," 11th Annual International IEEE Engineering in Medicine and Biology Conference Proceedings, pp. 831-832, 1989.

[18] House, D.H. and D.E. Breen, "Particles: A Naturally Parallel Approach to Modeling," 3rd Symposium on the Frontiers of Massively Parallel Computation Proceedings, pp. 150-153, 1990.

[19] Kilby, W.F., "Planar Stress-Strain Relationships in Woven Fabrics," Journal of the Textile Institute, Vol. 54, pp. T9-T27, 1963.

[20] Leaf, G.A.V. and R.D. Anandjiwala, "A Generalized Model of Plain Woven Fabric," Textile Research Journal, Vol. 55, pp. 92-99, 1985.

[21] Lloyd, D.W., W.J. Shanahan and M. Konopasek, "The Folding of Heavy Fabric Sheets," International Journal of Mechanical Science, Vol. 20, pp. 521-527, 1978.

[22] Ly, Nhan G., "A Model for Fabric Buckling in Shear," Textile Research Journal, Vol. 55, pp. 744-749, 1985.

[23] Mechanics of Flexible Fibre Assemblies, eds. J.W.S. Hearle, J.J. Thwaites and J. Amirbayat, (Sijthoff \& Noordhoff, Alphen aan den Rijn, The Netherlands) 1980.

[24] Metropolis, N., A.R. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, "Equation of State Calculations by Fast Computing Machines," Journal of Chemical Physics, Vol. 21, No. 6, pp. 1087-1092, 1953.

[25] Olofsson, B., "A General Model of Fabric as a Geometric-Mechanical Structure," Journal of the Textile Institute, Vol. 55, pp. T541-T557, 1964.

[26] Peirce, F.T., "The Geometry of Cloth Structure," Journal of the Textile Institute, Vol. 28, pp. T45-T97, 1937.

[27] Phan-Thien, N., "A Constitutive Equation for Fabrics," Textile Research Journal, Vol. 50, pp. 543-547, 1980.

[28] Shanahan, W.J., D.W. Lloyd and J.W.S. Hearle, "Characterizing the Elastic Behavior of Textile Fabrics in Complex Deformation," Textile Research Journal, Vol. 48, pp. 495-505, 1978.

[29] Shanahan, W.J. and J.W.S. Hearle, "An Energy Method for Calculations in Fabric Mechanics, Part II: Examples of Application of the Method to Woven Fabrics," Journal of the Textile Institute, Vol. 69, pp. 81-91, 1978.

[30] Terzopoulos, D. and K. Fleischer, "Deformable Models," The Visual Computer, Vol. 4, pp. 306-331, 1988.

[31] Weil, J., "The Synthesis of Cloth Objects," Computer Graphics (SIGGRAPH '86 Proceedings), Vol. 20, No. 4, pp. 359-376, 1986.

[32] Witkin, A., K. Fleischer and A. Barr, "Energy Constraints on Parameterized Models," Computer Graphics (SIGGRAPH '87 Proceedings), Vol. 21, No. 4, pp. 225-232, 1987.

[33] T. Lozano-Perez, "Task planning", in Robot Motion: Planning and Control, J. M. Brady et al., Eds. Cambridge, MA: MIT Press, 1982, pp. 473-498.

[34] J. L. Peterson, Petri net. Theory and the Modelling of Systems. Englewood Cliffs, NJ: Prentice-Hall, 1981.

[35] R. Fikes et al., "Learning and Executing Generalized Robot Plans", in Readings in Artificial Intelligence, N. Nilsson and B. Webber, Eds. Palo Alto, CA: Tioga, 1981, pp. 231-249.

[36] E. Sacerdoti, A Structure for Plans and Behavior. New York: North-Holland, 1977.

[37] D. E. Wilkins, "Domain-independent planning: Representation and plan generation", Artificial Intell, vol. 22, no. 3, pp. 269-301, 1984.

[38] R. Al-Jaar and A. Desrochers, "Petri Nets for Automation and Manufacturing", Advances in Automation and Robotics, vol. 2, Ed. G.N. Saridis, JAI Press, 1988.

[39] N. Vishwanadham and Y. Narahari, "Colored Petri Net Models for Automated Manufacturing Systems", Proceedings of the IEEE Robotics and Automation Conference, pp. 1985-1990, Raleigh, North Carolina, 1987.

[40] R. Valette, M. Courvoisier, H. Demmou, J. M. Bigou and C. Desclaux, "Putting Petri Net to Work for Controlling Flexible Manufacturing Systems", Proceedings of the International Symposium on Circuits and Systems, pp. 929-932, Kyoto, Japan, 1985.

[41] T. Agerwala, "Putting Petri Net to work", Computer, vol. 12, no. 12, pp. 85-94, 1979.

[42] P. Alanche et al, "PSI: A Petri Net Based Simulator for Flexible Manufacturing Systems", in Advances in Petri Net 1984, Lecture Notes in Computer Science 188, G. Rozenberg, Ed. New York: Springer-Verlag, pp. 1-14.

[43] K.-H. Lee and J. Favrel, "Hierarchical reduction method for analysis and decomposition of Petri nets", IEEE Trans. Syst. Man Cybern, vol. 15, no. 2, pp. 272-280, 1985.

[44] W. Zhang, "Representation of Assembly and Automatic Robot Planning by Petri Net", IEEE Trans. Syst. Man Cybern, vol. 19, no. 2, pp. 418-422, 1989.

[45] L. S. Homem de Mello and A. C. Sanderson, "AND/OR Graph Representation of Assembly Plans", AAAI-86 Proceedings of the Fifth National Conference on Artificial Intelligence, pp. 1113-1119, 1986.

[46] D. Parnas, "On the criteria to be used in decomposing systems into modules", Comm. ACM, 15 (2), pp. 1053-1058, 1972.

[47] B. Liskov and S. Zilles, "Programming with abstract data types", ACM Sigplan Notices, 9(4), pp. 50-59, 1974.

[48] A. Goldberg and D. Robson, Smalltalk-80: The Language and its Implementation, Reading, Mass., Addison-Wesley, 1983.

[49] B. Stroustrup, The C++ Programming Language, Addison-Wesley, Reading, Mass., 1986.

[50] E. Shapiro, Concurrent PROLOG: Collected Papers, Volumes 1 and 2. Cambridge, MA: MIT Press.

[51] I. Sommerville, Software Engineering, 3rd Edition, Addison-Wesley, 1989.

[52] G. F. Luger and W. A. Stubblefield, Artificial Intelligence and the Design of Expert Systems, The Benjamin/Cummings Publishing Company, Inc., 1989.

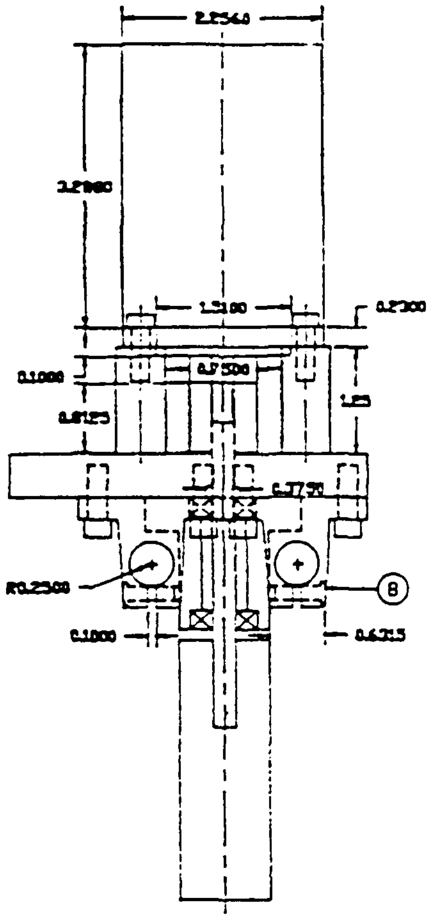
[53] B. Stroustrup, "What is Object-Oriented Programming?", IEEE Software, May 1988.

[54] M. Stefik and D. G. Bobrow, "Object-Oriented Programming: Themes and Variations", The AI Magazine, Vol. 6, No. 4, 1986.

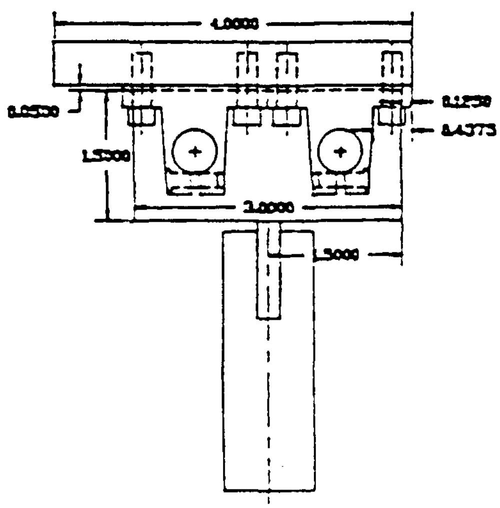
[55] P. Wegner, "Dimensions of Object-Based Language Design", OOPSLA '87 Proceedings, October 4-8, 1987.

[56] R. S. Wiener and L. J. Pinson, An Introduction to Object-Oriented Programming and C++, Addison-Wesley, 1988.

Appendix A
Internal Seam Alignment Devicc Details



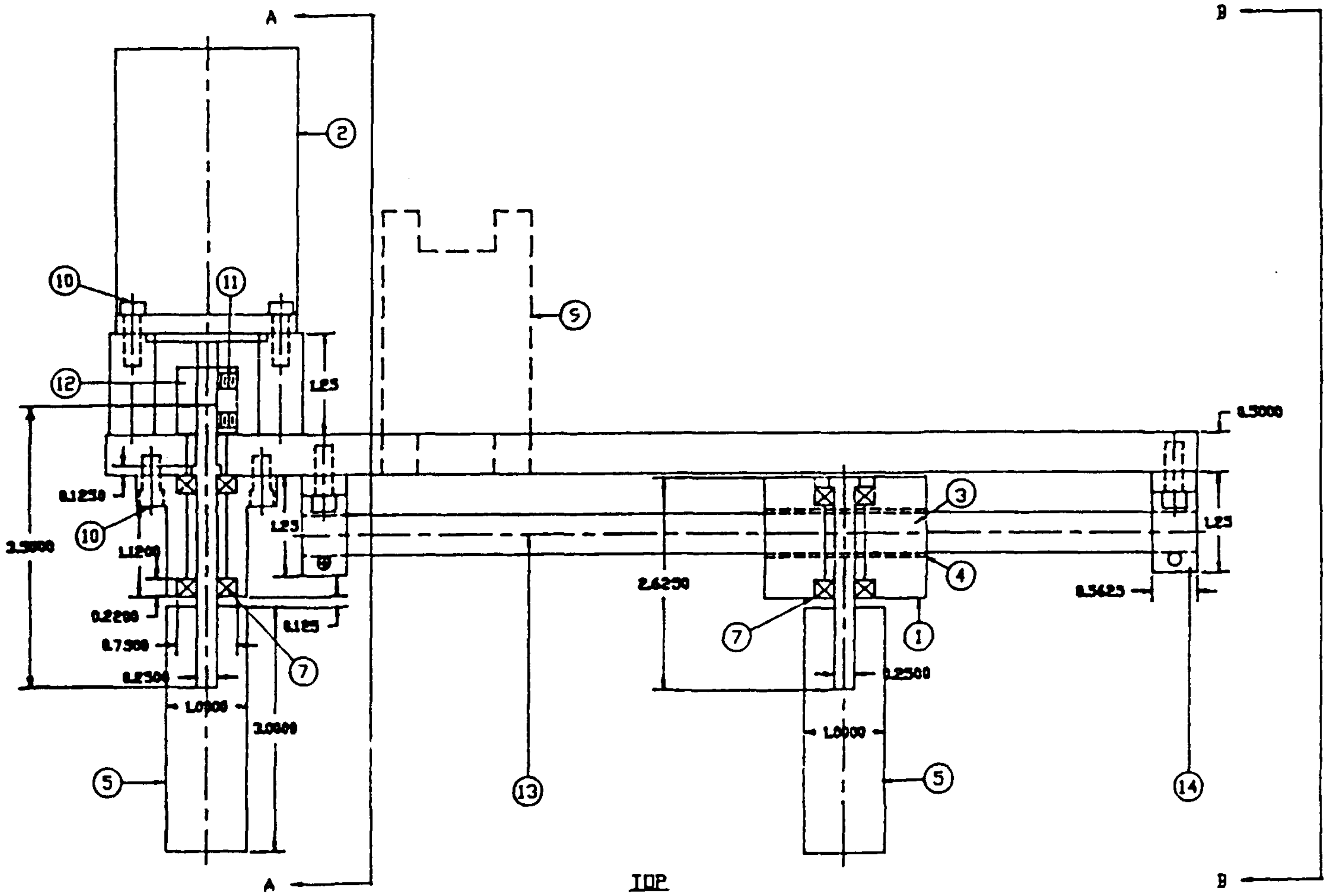
SECTION A-A



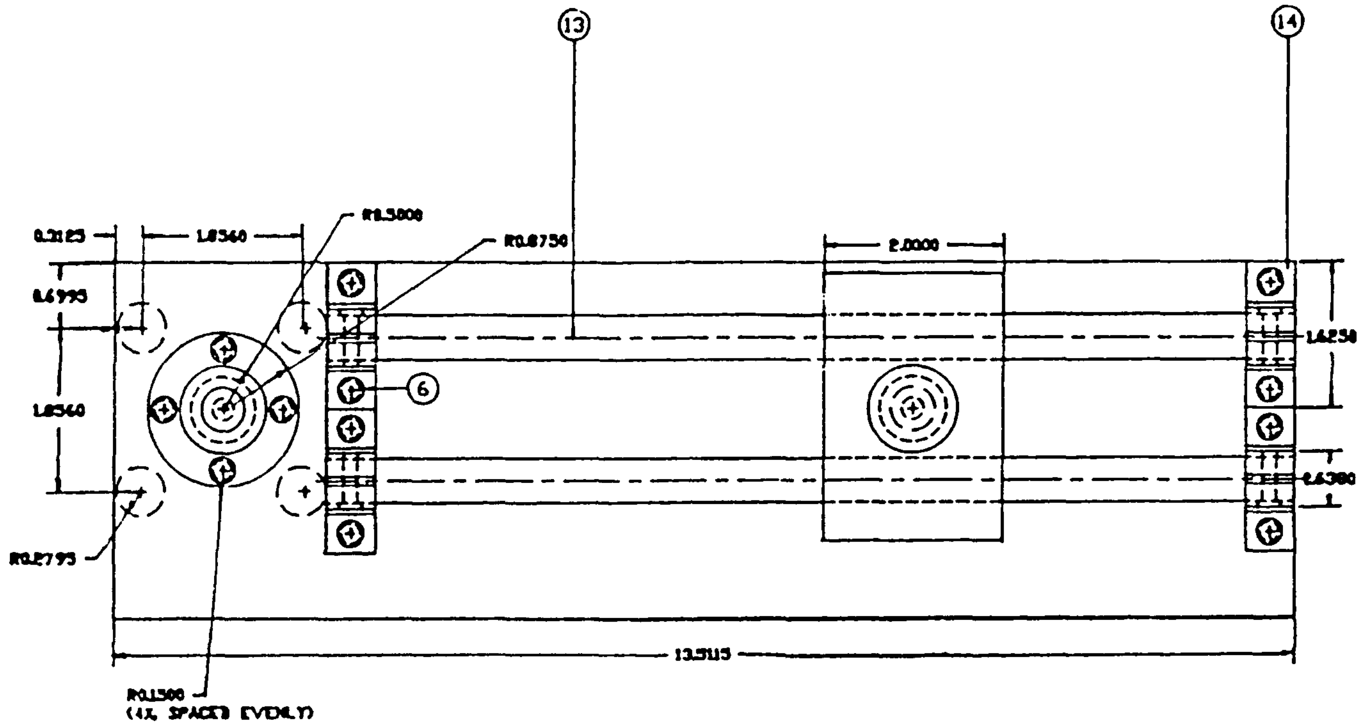
SECTION B-B

DETAIL	DESCRIPTION	QUANTITY
14	SHAFT SUPPORT BLOCK	4
13	GUIDE SHAFT 3/8 D X 12 L	2
12	MOTOR-SHAFT COUPLING	1
11	#6-32 UNC X 1/4 L SET SCREW	12
10	#10-24 UNC X 1/2 L SHCS	12
9	LOCATION OF AIR VALVE ASSEMBLY	1
	(SEE DWG # 3)	
8	#6-32 UNC X 1/2 L SHCS	4
7	BEARING, BALL 1/4ID X 3/4OD X .22W	4
6	#6-32 UNC X 1/2 L SHCS	8
5	ROLLER (COVERED BY RUBBER)	2
4	BALL BUSHING 3/8ID X 5/8OD X 7/8L	4
3	PUSH-IN RETAINING RING	4
2	STEPPER MOTOR	1
1	SLIDER BLOCK	1

DRAWN BY: TOBIAS HOPPE DATE: 07/31/90
APPROVED BY: DATE:
CMP
RPI
Troy, N.Y. 12180
SCALE: 0.45" = 1"
PART: ROLLER/SEAM ALIGNMENT DEVICE
TOLERANCE UNLESS SPEC'D:
USED ON:
PROJECT: F.I.T.
PROJECT #:
MATERIAL UNLESS SPEC'D: 6061-T6 ALUM.
DWG #: 12
REV:
SHEET 1 OF 1

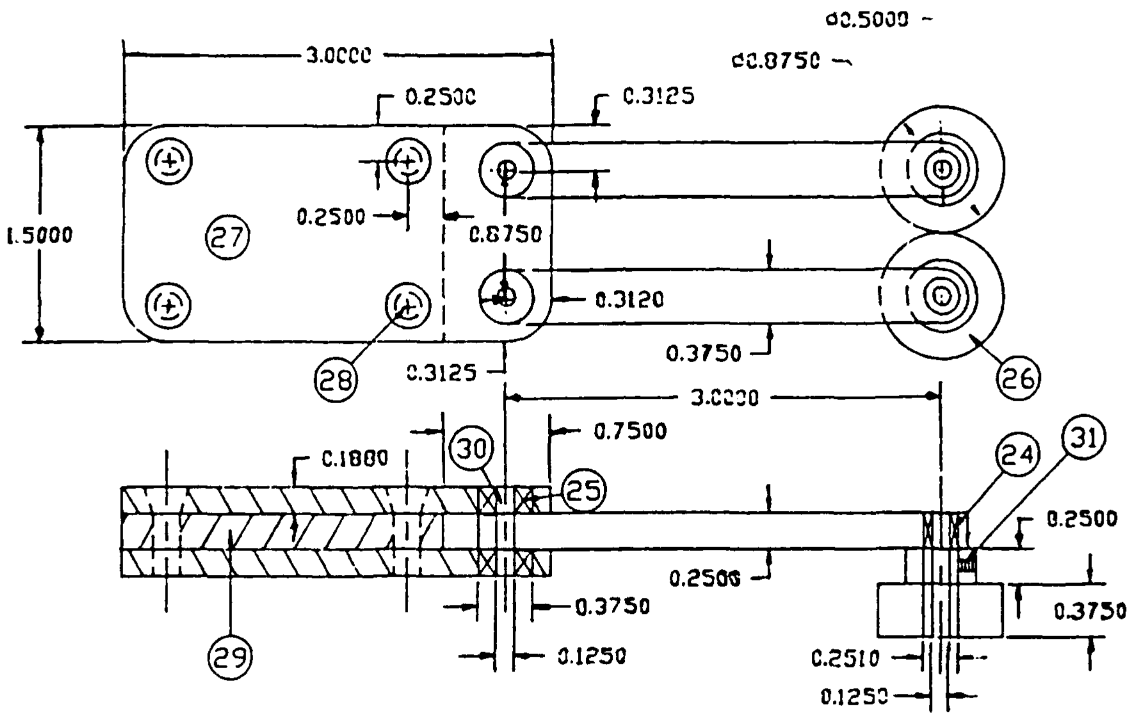


A
A
B
B
2
10
11
12
1.25
0.5000
3
4
1
7
5
13
14
9
0.1250
3.5000
1.1200
0.2200
0.7500
0.2500
1.0000
3.0000
0.125
2.6250
0.2500
1.0000
0.5625
1.25
TOP


13
14
0.3125
1.8560
R0.5000
R0.8750
2.0000
0.6995
1.6250
1.8560
6
0.6300
R0.2795
13.5115
R0.1500
(4X SPACED EVENLY)

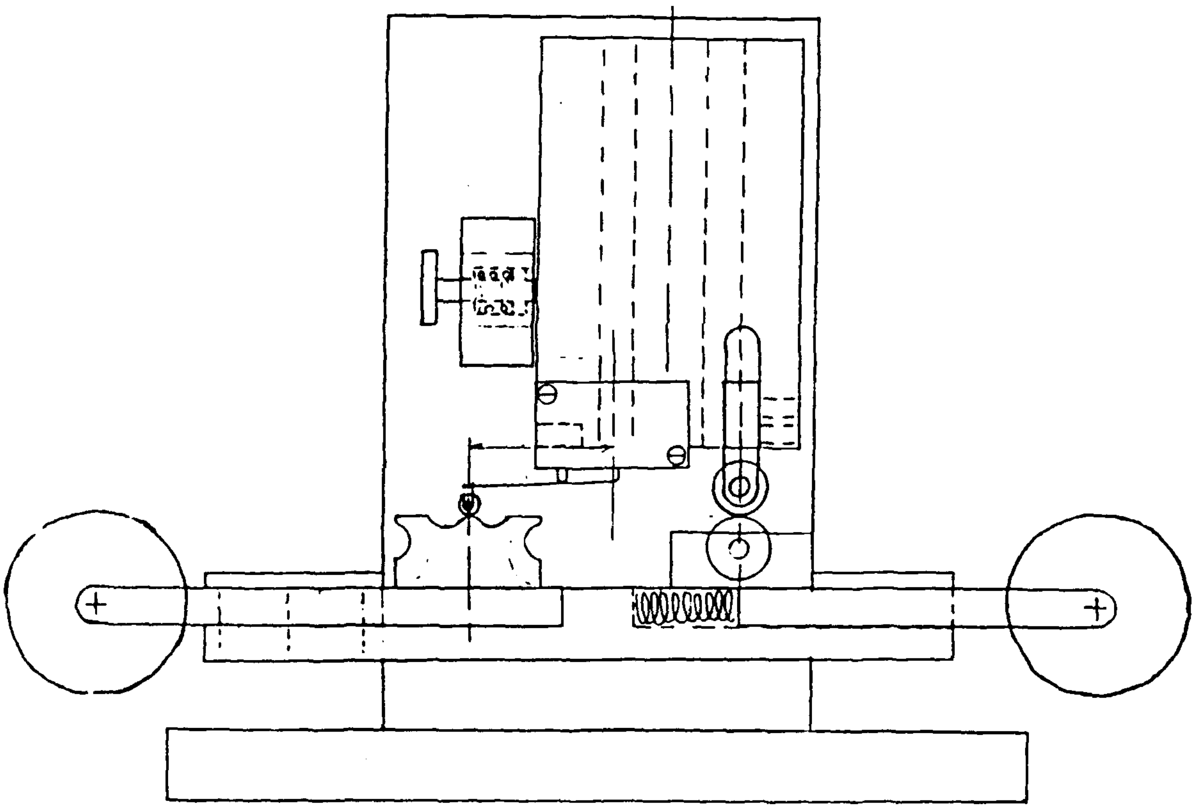
BOTTOM

Appendix B
External Seam Alignment Device Details


⌀0.5000
⌀0.8750
3.0000
0.2500
0.3125
1.5000
0.2500
0.8750
0.3120
0.3750
0.3125
3.0000
0.7500
0.1880
0.2500
0.2500
0.3750
0.3750
0.1250
0.2510
0.1250
24
25
26
27
28
29
30
31

Appendix C

Mechanical Seam Detection Mechanism Details



Appendix D
Robotics

A.1 HISTORY

Early automata started about 300 years ago. Inventors designed and fabricated a series of clockwork automata. Clockwork automata is based on the mechanics of a clock. The series were refined and became more intricate as time went by. The first mechanisms would be classified today as sequence machines. These machines could perform a series of activities stimulated by a main spring of a clockwork mechanism. The main spring drove gear trains which compressed air and turned cams. Push wires and push rods animated the mechanical object. Some of the inventions that classified as sequence machines were the Bird Organ, the Automatic Flute Player, and Malelardet's Automaton.

In operation, the Bird Organ moved its head and wings realistically using an arrangement of cams and levers. While whistling a tune, it opened and closed its beak. The assorted bird whistles were generated by valves and pistons.

The Automated Flute Player held a flute to its lips, blew air across the flute, and manipulated its fingers to cover the control ports of the flute. An air distributor inside the flute player controlled the pitch of the flute by releasing high, medium, or low pressure. The fingers were driven by a music box-type drum.

The most complicated and versatile of the clockwork automata was the incredible writing and drawing automaton, the Malelardet Automaton. It had an extremely large memory and very precise movements. It was operated by an unusually complicated series of cams, rods,levers, and shafts.

None of these automata had feedback or sensory capability. Complicated motions were carried out sequentially, but there was no response to anything in the environment.

A.2 MANIPULATORS, TELEOPERATORS, AND EXOSKELETONS

Manipulators, Teleoperators, and Exoskeletons use the same technologies as robots. Since their control comes from humans rather than from internal capabilities, they are not considered robots.

Initially, manipulators were driven and controlled by human operators by squeezing and twisting the control linkage. Later, power servo amplifiers were used to increase lifting and handling capabilities of the manipulators. Since it was possible to exert great force without the operator realizing it, it was necessary to equip the manipulator with force feedback. The feedback device supplied the operator with a sense of touch.

The next step after manipulators were teleoperators. These are remote hands that can be controlled by radio links, optical links, or even satellite communications.

Lastly, the exoskeleton is a control device used for the manipulator. The operator wears a skeleton framework that models the manipulators servo driven effectors. This technique can also be applied with the teleoperator.

A.3 EVALUATING AND DEFINING A ROBOT

There are five classifications for evaluating and defining robots and machines:

> Sequence Control Machines: Able to go through a specified sequence of actions according to present instructions.

Playback Machines: Can be taught to carry out a series of movements.

Controlled Path Robots: Can be programmed to follow a path between two points, no teaching is required.

Adaptive Robots: Have computer control and sensory feedback, so that they can react to their environment.

Intelligent Robots: Highest level of robot. Has an expert system which is a full set of sensors, a large memory, and a way to model its environment. This is intelligent by definition.

A.4 DEFINITION AND LAWS

The word "robot" was provided by a Czech playwrite who wrote a play called R.U.R. (Rossum's Universal Robots). The word "robota" in Czech means a worker providing compulsory service.

The three laws of robotics were written by a science fiction writer named Isaac Asimov, who also coined the term robotics. These laws are:

1. A robot may not injure a human being or, through inaction, allow a human to be harmed.

2. A robot must obey orders given by humans except when that conflicts with the first law.

3. A robot must protect its own existence unless that conflicts with the first or second laws. (1)

The "official" definition of robotics according to The Robotics Industries Association (RIA) is as follows:

> An industrial robot is a reprogrammable, multi-functional manipulator designed to move materials, parts, tools, or special devices through variable programmed motions for the performance of a variety of tasks. (2)

A.5 GENERAL DISCUSSION

A.5.1 SENSING DEVICES AND THEORIES

There are many sensing devices and theories existing and being developed. One is an eye coding theory. This theory is based on the human eye. The robot vision is divided into cells. Each cell is given a mathematical value which is later evaluated.

Another vision system which is modified after the human eye has depth perception. This system is applied to the Rochester Robot. It can perceive depth as it moves its head and can maintain focus on an approaching object. These skills may lead to an interactive robot which can deal with, adjust,and contribute to changing conditions.

Echovision is a sensor system that uses sound to enable a robot to distinguish between objects spaced only 0.1mm apart in depth and at a distance at about one half meter, which is about the working range of a typical robot arm. This system's positive attributes is that it is durable and a much simpler set up than optical systems. Acoustic transducers are not easily damaged and shrug off dirt or dust with ease. An optical system can be impaired by a simple strand of thread on its lens.

At JIAM '90, a robotic hand was shown that could pick up one ply of fabric by sensing its thickness. This was demonstrated by the Research Institute of Polymers andTextiles.

Researchers are beginning to realize that instilling a sense of touch may be a cheaper way of getting a blind robot to handle an object properly. "Artificial Skins" have been developed out of a polymer film which generates electric signals when compressed. There is also a possibility of using acombination of touch and slippage sensors. The slippage design is an embedded ball under the surface of the gripping part of the manipulator. This combination creates a highly versatile and efficient robot.

A.5.2 APPLICATIONS IN THE APPAREL INDUSTRY

Robotics have been applied in the apparel industry in two primary areas - production and warehousing.

The robot used for warehousing is a basic pick and place manipulator called a picker. The picker is run by a computer. It can perform the task of gathering needed goods and also transport the goods to the necessary departments. For example, the picker is instructed by the computer to get ten pink blouses is size 8 to be shipped. The picker finds all the blouses under the control of the computer, takes them to the packing department, and then to the shipping department.

In production, the applications are also basic pick and place manipulators and highly automated sewing machines. Groups of these "robots" create "islands of automation" which can be connected by a Unit Production System (UPS) or other suitable transporter.

The best production success to date has occurred in the cutting room. CAD/CAM devices have been integrated directly into the network so that a design can:

- Be graded.
- Set into a marker.
- Then be relayed to an automatic spreader.
- And finally be used to direct an automatic bundle labeler and cutter.

A new system being developed and marketed will cut garments one at a time and pick them off the cutting table onto a Unit Production System for transportation through the sewing process.

Communication standards have been proposed to attempt to move this technology to open protocol.

A.6 TECHNOLOGY AND RESEARCH

A.6.1 WHAT KEPT ROBOTICS LIMITED IN THE APPAREL INDUSTRY

Some of the reasons why robotics have been kept out of the apparel industry are that production facilities are not designed nor redesigned to accommodate robotics and the operators are not trained to use or trust the capabilities of the machinery. We have discovered in this and allied research that operators receive minimal instruction on automation. Time is set aside by management to learn how to feed, start, and stop the workstation. If thread is involved they also learn to fix needle and thread problems. Systems analysis is generally omitted so operators do not learn the true capabilities of the technology they are using.

The limited involvement of robotics in the apparel industry is also the result of the high cost of manufacturing or buying robotics, and the lack of research to fully develop this technology. The apparel industry typically spends less than 1/2 of 1% of gross sales on research and development. This is insufficient funding for this technology.

Manipulators are developed to the point where they can reliably grasp and sense rigid, solid surfaces, but not limp fabrics. Some research is now being done on improving the industry standard - the Clupicker. The Clupicker was developed by research engineers at Cluett, Peabody & Co. It is a single ply gripper that is normally used to separate plies from a stack.

A.7 COMPUTER INTEGRATED MANUFACTURING

A.7.1 DEFINITION

Computer Integrated Manufacturing (CIM) is the utilization of computer technology for management, control and operation of a manufacturing facility by interfacing with both the physical and the human resources of a company. Thus, CIM is intended to improve manufacturing performance and responsiveness and to facilitate equipment development in the apparel industry. (3)

A.7.2 CIM AND RPI

RPI is trying to integrate its series of computers and robot so that they can connect into a CIM system. The researchers will need to know what kind of information the data base(s) requires to define the pressing cycle. For example, the size, style, and fabric of the pants has to be known. The size and style affects the smoothing cycle and the fabric content affects the time duration and temperature of the pressing.

The cell control computer has to be fed this information from its sensors in order to instruct all the different components of the system. The controller can also obtain information from the sensors of the press to aid in further manipulation of the pants. The plant of the future will have the linkages to look to the style master for pressing information on a garment by garment basis. The cell controller will then regulate the pressing cycle and material transfer devices to achieve a quality press before handing the pants to the next operation.

A.7.3 OBSTACLES

There are many steps in computerizing and integrating a production facility. Some hurdles that need to be overcome are technical considerations such as integrating the computer software and hardware. In addition to these technical considerations, "people issues", such as building user confidence, establishing reliable preventive programs, maintenance, and training must be observed. (4)

Group dynamics between the workers and management must be addressed in order to break down the communication barriers and build positive interactions between technology and user. When such organizational changes take place, they also bring about changes in the building design. For example, color schemes are brighter, white and blue collar workers enter through the same door, and they eat in the same cafeteria. The engineering and office space should only be separated by a glass wall from the manufacturing floor for environmental reasons. (5)

Building design is of grave importance to achieve maximum flexibility for a computerized and/or integrated production facility. Factories should be designed from the inside out. Computer aided design can be applied to achieve the greatest flexibility in plant design. Flexible spaces can be created which can be moved in order to set up the floor for different production plans with great ease and low cost. With these new and optimal methods in floor planning and flexible technologies, future factories are likely to be 10 - 20% smaller. (6)

A.8 CONCLUSION

The intent therefore of the first phase of this project was to look at the viability of various technologies to solve our problem of pressing a pair of trousers. The robot used in our research has been a major asset in the first year in that it has permitted the team the opportunity to try various elements of designed hardware without the necessity of having to be concerned with transportation around the cell.

This flexibility has accelerated some aspects of the investigation and also allowed us to begin thinking about other matters of concern such as CIM. It is the intent of the research group that any system developed will be compatible with other work being sponsored within the industry. To that end we have established a liaison link with the AAMA sponsored CIM project at Southwest Louisiana University, Computer Design, Inc. and other firms and organizations working on this level of technology.

FOOTNOTES

1) Mikell P. Groover, Mitchell Weiss, Roger N. Nagel, and Nicholas G. Odrey. Industrial Robotics: Technology, Programming, and Applications., McGraw Hill Inc., New York, 1986, p. 7,8.
2) Mikell P. Groover, Mitchell Weiss, Roger N. Nagel, and Nicholas G. Odrey. Industrial Robotics: Technology, Programming, and Applications., McGraw Hill Inc., New York, 1986, p. 6.
3) The Apparel CIM Center, University of Southwestern Louisiana, September 1989.
4) Electronic Business, Cahners Publishing Co., Massachusettes, November 1, 1987, p. 124.
5) Engineering News Record, McGraw Hill Inc., New York, September 26, 1985, p. 33.
6) Engineering News Record, McGraw Hill Inc., New York, September 26, 1985, p. 30, 32.

REFERENCES

1) Foley, Mary Jo, "Expert Systems on the Factory Floor", Electronic Business, November 1, 1987, Cahners Publishing Co., Massachusettes, 1987.
2) Furukawa, Tsukasa, "Hi-Tech Takes Giant Step At JIAM", Daily News Record, Friday, June 22, 1990, Fairchild Publications, New York, 1990.
3) Groover, Mikell P., Weiss, Mitchell, Nagel, Roger N., and Odrey, Nicholas. Industrial Robotics: Technology, Programming, and Applications. McGraw Hill Inc., New York, 1986.
4) Kellso, James R., "CIM in Action: Microelectronics Manufacturer Charts Course Toward True Systems Integration", Industrial Engineering, July 1989, Institute of Industrial Engineers, Georgia, 1989.
5) Port, Otis and Shiller, Zachary, "Smart Factories: Americas Turn? A Few Advanced Plants Now - A Competitive Arsenal Soon", Business Week, May 8, 1989, McGraw Hill Inc., New York, 1989.
6) "An Expert System Primer", Electronic Business, November 1, 1987, Cahners Publishing Co., Massachusettes, 1987.
7) "Flexibility Key to Future Factories", Engineering News Record, September 26, 1985, McGraw Hill Inc., New York, 1985.
8) "Mechanical Arm Gives Robots a Helping Hand", Machine Design, January 12, 1989, Penton Publishing, Ohio, 1989.
9) "Research on Mechanical Hands on Innovative Software", Research and Development, January 1986, Cahners Publishing Co., Iowa, 1986.
10) "Robotic Vision Could Benefit from Eye Coding Theory", Research and Development, May 1988, Cahners Publishing Co., Iowa, 1988.
11) "Robots Get `Ears' for Better Depth Perception", Electronics, January 8, 1987, McGraw Hill Inc., New York, 1987.
12) "Scientists Give Robot Eyes With Depth Perception", Machine Design, April 6, 1989, Penton Publishing, Ohio, 1989.
13) "The Crucial Steps to Automation", Business Week, May 8, 1989, McGraw Hill Inc., New York, 1989.
14) "The Factory of The Future", The Economist, April 5, 1986, The Economist Newspaper Ltd., London, 1986.
15) "Touch and Go for Robots", Science News, Vol. 127, March 23, 1985, Science Service Inc., Washington D.C., 1985.
16) The Apparel CIM Center, University of Southwestern Louisiana, September 1986.

List of Figures

List of Tables

1.0 INTRODUCTION

This report contains additional information in the areas of Sensing and Integration. More detail is provided pertaining to the sensing of wrinkles and to the integration and control of the pressing system.

2.0 SENSING

2.1 WRINKLE DETECTION

2.1.1 Curve Approximation and Analysis

The difference curves as shown in Figure 21 of the original report are approximated using Cubic B-splines to allow extraction of derivative information for evaluation of the wrinkles, if any are present.

The cubic B-spline approximations are polynomial functions which when interpreted (i.e., evaluated for each point x), produce y-values which accurately represent the original values of the function. The polynomial functions can be manipulated to produce curvature and local peak and valley information of the curves. Those points which are used to extract wrinkle information will be referred to as *evaluation points.*

The first derivative of the polynomial function describes the slope of the curve at the evaluated point. The second derivative of the polynomial describes changes in the first derivatives. Inflection points are points at which the first derivative, or slope, changes sign, and thus indicate a change in curvature from convex to concave. Inflection points occur at points where the second derivative is equivalent to zero.

Several methods for producing evaluation points were explored. Inflection points were originally thought to be a good evaluation criteria, but it was found that the curvature of the material did not always change at the beginning and end of a wrinkle, and that the curvature at peaks and valleys was often subtle enough that the change in curvature was not obvious. Figure 1 is an example of a difference image evaluated using the inflection point technique. The vertical bars above the curve indicate locations at which inflection points were found.

Figure 1 Inflection Points of B-spline Approximation to Curve

The second technique explored was a method which evaluates the B-spline approximation for points at which the first derivative does not equal zero. Thus, the flat areas where no wrinkles exist or plateaus in a wrinkle exist would be disregarded. However, this method produced numerous evaluation points. Figure 2 is an example of a difference image evaluated using this method. The vertical bars above the curve indicate locations at which the first derivative does not equal zero.

Figure 2 B-spline Approximation First Derivative Points Not Equal to Zero

The third method explored was to evaluate the first derivatives for changes. Those points for which a neighboring point's first derivative differs from the local point's derivative, indicate changes in slope. Changes in slope occur at the beginning, end, peaks and valleys of the wrinkles, as well as other points. This approach has proven to be much more accurate than the inflection point approach, and produces less evaluation points than the second method. Figure 3 is an example of a difference image evaluated using the difference in first derivatives technique. The vertical bars above the curve indicate locations at which the first derivative differs between adjacent points.

Figure 3 B-spline Approximation Points of Changing Slope

Thus, the first derivatives of the cubic B-spline approximation are determined, and those points at which the first derivatives differ are labeled as evaluation points. The evaluation points are used to minimize the search for wrinkle information.

2.1.2 Wrinkle Data Extraction

The next step in the process of wrinkle detection involves evaluating the data for locating wrinkles and determining their characteristics. This is performed by searching through the evaluation points found in the above derivative evaluation process and tracking the height of the curve as well as whether a wrinkle has been encountered.

The wrinkle information obtained from the search is output to an image file. The location of the beginning, end, and peak of each wrinkle, as well as the maximum height and width of each wrinkle is formatted to an ascii file. Figure 4 is the wrinkle information file obtained from processing the difference contour of figure 5.

```
IMAGE 0
Wrinkle 0 :  width=53   height=16
  beg=21  peak=51  end=74

Wrinkle 1 :  width=71   height=30
  beg=104  peak=129  end=175

Wrinkle 2 :  width=2   height=6
  beg=215  peak=216  end=217
```

Figure 4 Wrinkle Information Output File

Figure 5 Difference Image of Buck Contour and Wrinkle Contour

The orientation of the wrinkles may be determined by performing a correlation between adjacent images. The correlation value is greatest for the shift which represents the maximum correlation between two images. By successively applying a correlation function to each pair of adjacent images of the garment on the buck, the shift of the wrinkles between each image can be determined and the overall orientation of the wrinkle produced.

The correlation is performed by multiplying the area under the curve representing the difference image at buck location one by the area under a shifted version of the curve representing the difference image at buck location two. Multiplication is performed by multiplying the y-value of each pixel location of difference image one with the y-value of the corresponding pixel locations in difference image two. Each of the 256 products from the 256 pixel locations are summed to produce the resultant correlation value.

The process is repeated for 512 shifted versions of image two, where image two is first shifted 256 to the left to align its right-most pixel with the left-most pixel of image one, and finally shifted to the right to align its left-most pixel with the right-most pixel of image one. The largest of the 512 correlation values indicates the shift between the two images, and thus the orientation of the wrinkles.

Testing was performed on the correlation algorithm for trousers with wrinkles placed by hand. The success of the testing was due to the consistent orientations of all the wrinkles on the trousers. The correlation algorithm would not work well for numerous wrinkles in various orientations. However, there is reason to believe that the trouser handling device will produce wrinkles in a particular orientation as it places the trousers on the press buck. Thorough testing will be required once the final handling device is in place.

Figure 6 illustrates the correlation performance on a set of six curves. A negative shift value indicates that image(i+1) should be shifted to the left by the value to correlate with image(i), and a positive value indicates that image(i+1) should be shifted to the right by the value to correlate with image(i). Excluding image four, which exhibits a dissimilar wrinkle, the correlation values are fairly similar, indicating a consistent orientation.

CORRELATION BETWEEN IMAGES 1 AND 2
shift = -8

CORRELATION BETWEEN IMAGES 2 AND 3
shift = -1

CORRELATION BETWEEN IMAGES 3 AND 4
shift = 9

CORRELATION BETWEEN IMAGES 4 AND 5
shift = -3

CORRELATION BETWEEN IMAGES 5 AND 6
shift = -2

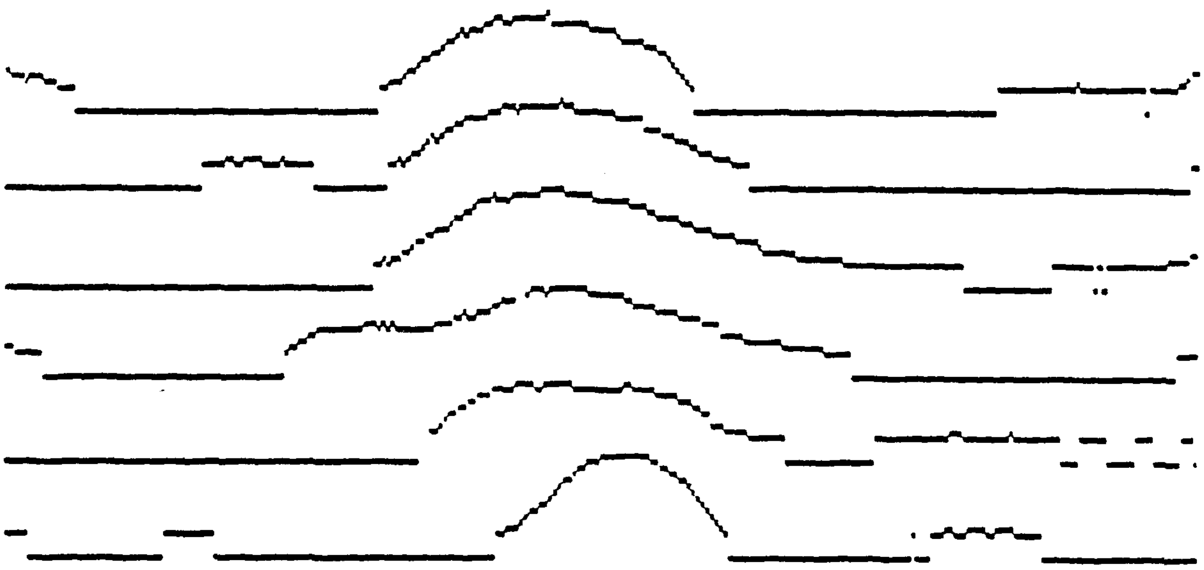


Figure 6 Correlation Values Between Six Images

2.1.3 Image Resolution

The original algorithms were developed for images of size 256 by 242 pixels. These images spanned an area of approximately 8 inches by 8 inches, thus providing a resolution of approximately 32 pixels/inch.

Evaluation of the wrinkle algorithms led to reducing the image resolution to decrease the amount of processing time required. An averaging algorithm was produced to create an image width of 64 pixels by averaging every four pixels. The averaging takes place following the contour extraction and operates on the 256 by 1 array of the curve locations, thus requiring only 64 averages (as opposed to the 64 x 64 = 4096 averaged required for an entire image). The new resolution of approximately 8 pixels/inch produced results similar to those of the higher resolution algorithm.

The averaging of the difference image array also aided in the removal of noise by averaging noisy pixels (i.e. pixels with difference values much greater or less than their neighbors) with neighboring pixels. Noise removal reduces the search space required for wrinkle information extraction; less noisy difference arrays produce fewer evaluation points from the change in slope process.

Figures 7 and 8 illustrate an averaged buck image and wrinkled garment on buck image, respectively. Figure 9 illustrates evaluation points along the difference array. In comparison to Figure 5 of the same difference array non-averaged, the averaged array is similar, but has less noise.

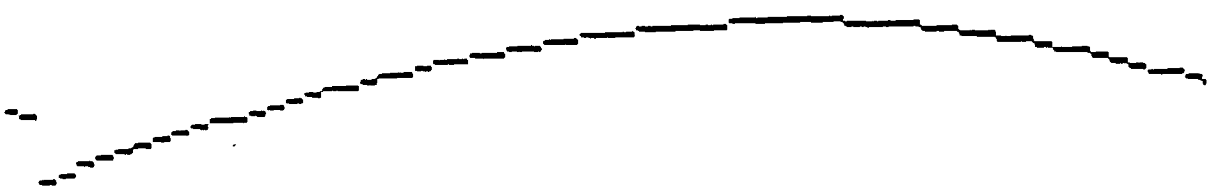


Figure 7 Averaged Buck Contour With 64 Points

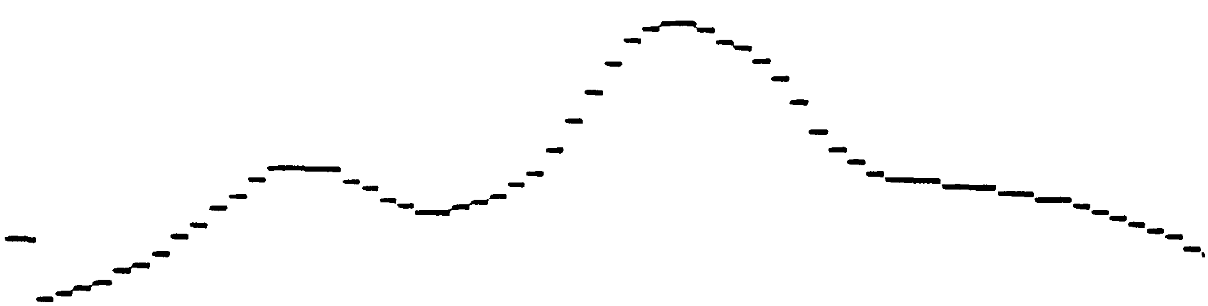


Figure 8 Averaged Wrinkled Garment Contour With 64 Points

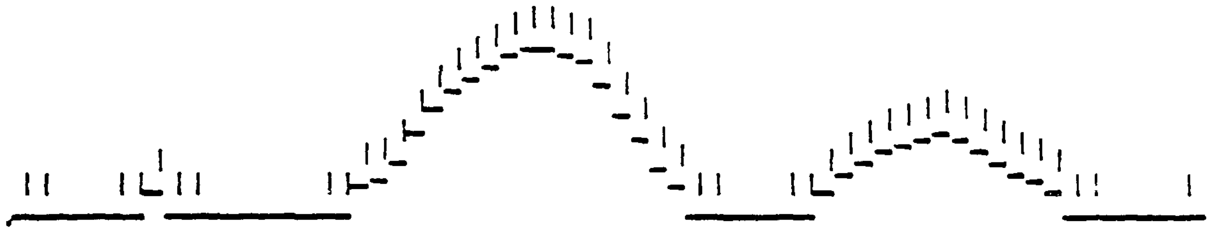


Figure 9 Averaged Difference Image With Evaluation Points

The final wrinkle detection system will require a field of view of twelve inches or more to handle the widest possible trouser leg width. Averaging algorithms can be modified based on the final system's camera CCD size and field of view to produce a similar resolution to that produced above.

2.1.4 Calibration

Camera calibration is required to determine the absolute wrinkle size information from the images. The calibration procedure relates the actual height of an object to the observed pixel height in the image to produce a conversion value.

For the camera and light source configuration pictured in Figure 10, the calibration is a simple operation. Placing an object of known height (preferably a block with a flat upper surface) beneath the lightline will produce distortions in the light. The image will contain a bright line with a height change at the location of the block.

The difference image of the buck surface and the image of the light across the block will contain the height information in pixels of the block from the camera viewpoint. The conversion factor to obtain the actual height information from an image is:

conversion factor = actual block height / image block height in pixels

The block used for calibration was .5 inches high. Evaluating the difference image array produced an average block height of 27.9 pixels. The conversion factor is then .018 inches/pixel for vertical measurements.

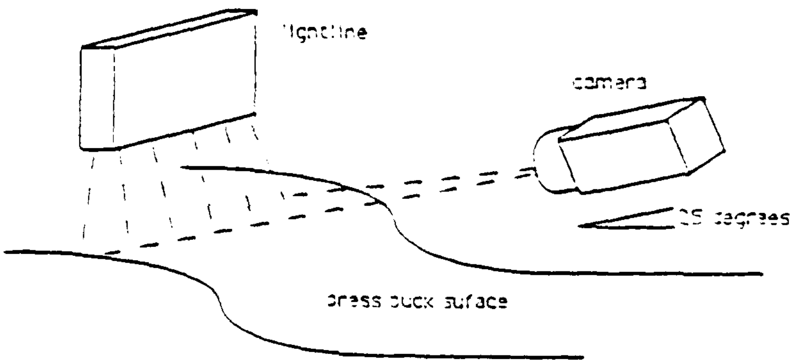


Figure 1 Light Line Scanning Configuration

2.1.5 Remarks

The wrinkle detection algorithms above produced highly accurate results for trousers with wrinkles placed by hand. Due to the difficulty in measuring and describing the actual wrinkles in the garment, the best indicators of the accuracy are the image resolution and the accuracy of the modeling algorithms. Testing of the grasping devices and wrinkle removal devices will aid in determining the accuracy required for wrinkle detection in the final system. In addition, the testing of the mechanical devices will demonstrate what type of wrinkles will tend to be placed in the trousers by the handling devices.

2.2 WRINKLE SETTING TESTS

Tests were performed on trousers of four different materials to determine the wrinkle heights and widths that would be pressed into the trousers. The results are shown in Table 1.

Material	Wrinkle Hgt. (in.)	Wrinkle Wid. (in.)	Pressed Hgt. (in.)*
# 7	1 / 2	11/16	1 / 4
	1 / 2	3 / 4	none
	1 / 2	3 / 4	1 / 8
	1 / 2	13/16	1 / 16
	1 / 2	7 / 8	none
	1 / 2	7 / 8	1 / 4
	5 / 8	3 / 4	3 / 8
	3 / 4	7 / 8	1 / 4
# 9	3 / 8	3 / 5	none
	1 / 2	3 / 5	none
	1 / 2	5 / 8	3 / 8
	1 / 2	7 / 8	none
	5 / 8	3 / 4	1 / 8
	5 / 8	3 / 4	3 / 8
	3 / 4	2 / 3	1 / 3
	1.0	2 / 5	3 / 4
# 10	1 / 4	7/16	none
	1 / 4	5/16	1 / 8
	3 / 8	3 / 8	1 / 8
	3 / 8	7/16	none
	3 / 8	7/16	none
	5 / 8	3 / 8	3 / 8
	1 / 2	7/16	1 / 4
	1 / 2	7/16	1 / 4
	1 / 2	1 / 2	1 / 8
	1 / 2	5 / 8	none
	1 / 2	11/16	none
	5 / 8	1 / 2	none
# 8	3 / 8	5 / 8	none
	1 / 2	5 / 8	1 / 8
	1 / 2	11/16	1 / 8
	1 / 2	13/16	none

* The width of the pressed wrinkle is negligible since a crease was produced with the height as indicated.

In general, as the wrinkle width gets larger than the wrinkle height, the wrinkle will not get pressed into the trousers.

It was found that the location of the wrinkle also affected whether or not the wrinkle was pressed into the trouser.

Table 1 Wrinkle Setting Test Data

3.0 INTEGRATION

As the work on the four technology areas has been proceeding, work has also been done on describing the configuration of a production pressing system as shown in Figure 11. A description of a proposed system sequence of operation follows.

Delivery System

Empty Carriages are queued in Empty Queue.

Operator brings Empty Carriage into Loading Station, either by physically grabbing the first Carriage in Empty Queue or by pushing a button to request that Control System do the same.

Operator loads Leg A, requests Leg A rollers to tension (probably by footswitch).

Operator loads Leg B, requests Leg B rollers to tension.

Operator inpsects pants on Carriage, then either releases the carriage into the Loaded Queue (again either physically or pushbutton) or takes corrective action, including Tensioning and Releasing rollers as many times as needed. If operator releases Carriage, system increments Loaded Queue counter.

When Loaded Queue counter is below a predetermined level, system notifies operator.

Alignment System (AS)

If Loaded Queue counter is equal to at least one, take a Carriage, place it in the AS, and decrement the counter.

Extend & Clamp Knee Aligners

Extend Cameras at knee height.

Get Seam Offset at knee height.

Align seams at knee as indicated.

Get Seam Offset again to verify within tolerance, align again if necessary.

Move Cameras to cuff.

Get Seam Offset at cuff.

Align Seams at cuff as indicated (engage and disengage clutch if necessary).

Get Seam Offset again to verity within tolerance, align again if necessary.

Retract cameras.

Signal Pants Ready to be removed from AS.

When Transfer System signals it has Grabbed Pants, then system releases both leg rollers, releases and retracts knee aligners then signals that Pants Released.

When Transfer System signals it has Cleared AS, system releases carriage into Empty Queue and re-starts its cycle from the beginning.

Transfer System (TS)

If TS is empty, attach to Robot if not already attached.

If TS is empty and we are attached to Robot and AS signals Pants Ready:
    Move to AS.
    Grab both legs.
    Wait for AS to signal Pants Released
    Tension both legs.
    Move away from AS and signal when we have Cleared AS.

Move to Open Position and Open Legs.

Place Leg A on Buck.

Open Cuff Gripper A and move off Buck.

Dock TS then Release Robot.

Wait for WS to signal that Leg A is Wrinkle Free and Wrinkle System (WS) is Clear of Press.

Start Press Cycle for Leg A

Request Robot, then Undock.

Once we are Undocked and Press Cycle is finished, mark Leg A as pressed, remove it from Buck, and place Leg B on Buck.

Open Cuff Gripper B and move off Buck.

Dock TS then Release Robot.

Wait for WS to signal that Leg B is Wrinkle Free and WS is Clear of Press.

Start Press Cycle for Leg B.

Request Robot, then Undock.

Once we are Undocked and Press Cycle is finished, mark Leg B as pressed and remove it from Buck.

Release pants from TS and re-start Cycle.

Wrinkle System (WS)

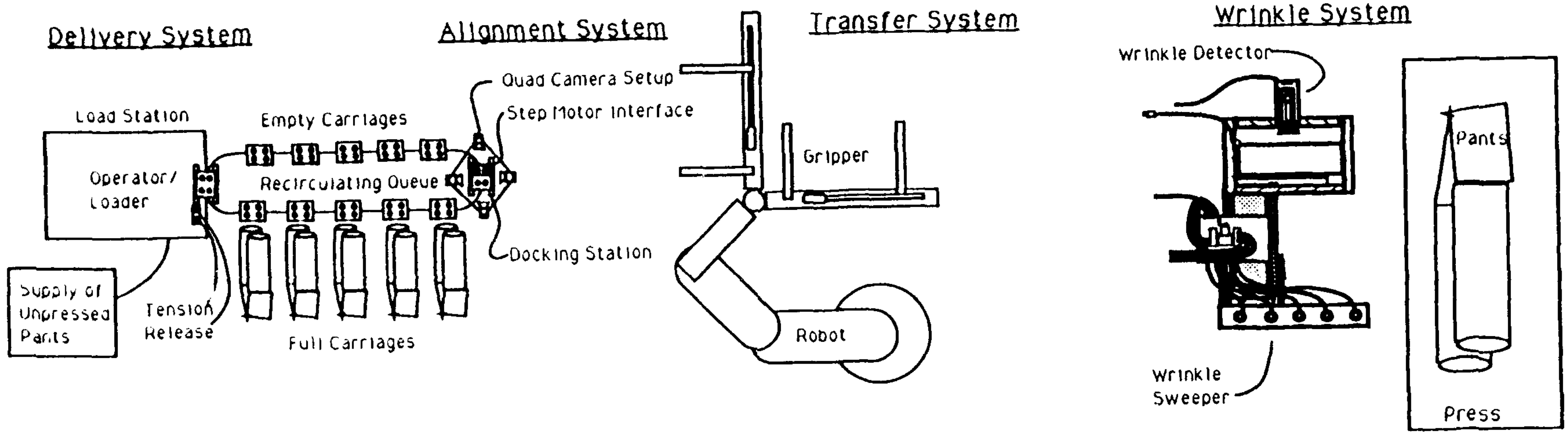
When there is a Leg on the Buck and it is not Wrinkle Free, request Robot.

Once Robot is attached, Sweep for wrinkles and smooth them until Leg is Wrinkle Free and marked as such.

Move away from press and signal when Clear of Press.

Release Robot.

Table 2 shows the output and input requirements for a simplified version of the pressing system.



Delivery System

- Raw Stock Supply
- Empty Carriages
- Docking Station
- Operator/Loader
- Tension/Release Buttons for rollers
- Queue Upcounter
- Release to Loaded Q

Alignment System

- Loaded Carriage Supply
- Docking Station
- Step motor drive interface
- Quad camera setup
- Camera extend & raise device
- External knee aligner device

Transfer System

- Robot Availability
- Gripper Docking Station
- Aligned Pants Available
- Transfer Gripper with leg longitudinal and lateral tensioning devices and dual leg pivoting action

Wrinkle System

- Robot Availability
- Sweeper Docking Station
- Press loaded with Pants
- Wrinkle Sweeper and Detector Apparatus

Figure 11 Integration System Diagram

OUTPUTS

Cuff Tension[1]	Tension/Release
Cuff Tension[2]	Tension/Release
Shuttle	Off/On
Shuttle	Slow/Fast
Shuttle	Alignment/Press
Shuttle Lock	Unlock/Lock
Cuff Gripper	Open/Closed
Fixed Gripper	Open/Closed
Head Steam	Off/On
Buck Steam	Off/On
Vacuum	Off/On
Head Hesitation	Off/On
Head Close	Off/On
High Pressure	Off/On
Jet[1] (or Sweep)	Off/On
Jet[2]	Off/On
Jet[3]	Off/On
Jet[4]	Off/On
Jet[5]	Off/On
Handchanger	Engage/Disengage
Transfer Rack	Clamp/Unclamp
Fan	Off/On

Note: Multi Chamber Buck will Require Additional Outputs

INPUTS

Shuttle Slow
Shuttle Stop
Transfer Rack Ready to Clamp
Transfer Rack Clamped
Handchanger Engaged

OTHER

Align Seam[1]	Stepper Motor
Align Seam[2]	Stepper Motor
Leg Tension	Servo Controller
Get Seam Offset[1]	Vision Routine
Get Seam Offset[2]	Vision Routine
Wrinkle Detect	Vision Routine

Table 2 Simplified System Interconnection

ADDENDUM

PROCEEDINGS

Frontiers'90

THE THIRD SYMPOSIUM ON THE

FRONTIERS OF MASSIVELY PARALLEL COMPUTATION

OCTOBER 8-10, 1990 • UNIVERSITY OF MARYLAND • CENTER OF ADULT EDUCATION • COLLEGE PARK, MARYLAND

IEEE COMPUTER SOCIETY

IEEE NATIONAL CAPITAL AREA COUNCIL

NASA
National Aeronautics and Space Administration
Goddard Space Flight Center

In cooperation with The University of Maryland

ADDITIONAL SUPPORT FROM:

ACTIVE MEMORY TECHNOLOGY, DIGITAL EQUIPMENT CORPORATION, MARTIN MARIETTA AEROSPACE, MASPAR COMPUTER CORPORATION, AND THINKING MACHINES CORPORATION

IEEE Computer Society Press

The Institute of Electrical and Electronics Engineers, Inc

Table of Contents

Applications

Particles: A Naturally Parallel Approach to Modeling

Donald H. House
Department of Computer Science
Williams College
Williamstown, Massachusetts USA

David E. Breen
Rensselaer Design Research Center
Rensselaer Polytechnic Institute
Troy, New York USA

Introduction

Motivated by both the goal of developing a real-time training simulator for arthroscopic surgery [8] and the goal of developing an accurate model of cloth for automatic garment handling, we are currently developing particle-based techniques for the mechanical and visual modeling of complex non-rigid materials. Using this approach, a material is represented as a large collection of microscopic particles interacting with each other according to simple physical laws operating on a microscopic level. Since modeled materials derive their macroscopic properties from these microscopic interactions, the technique closely parallels the actual discrete structures of nature. This contrasts with more conventional finite element techniques that use much smaller numbers of more complex elements, and that specify element properties directly in terms of physical laws operating at the macroscopic level.

In particle-based modeling one does not assume that the properties of the constituent elements used to model a material are a scaled down version of the properties of the material as a whole. Instead, one attempts to create the micro-primitives (particles) whose interactions, when aggregated, produce the desired macroscopic behavior. Thus, in developing one of these models, there is more to learn from scientific molecular models than there is from engineering models for materials.

In our conception, particles are hierarchically organized atomic, discrete entities containing fixed parameters governing how they interact with other particles, and state variables that are updated as the simulation proceeds. Each particle obeys laws that govern how its state variables change as functions of both time and inputs from other particles. In our current design, each particle receives direct inputs from a set of "neighboring" particles and inputs from more distant particles through a global information evaluator. A particle's set of "neighbors" may change during the course of a simulation as the particles change positions relative to each other.

The particle approach addresses many of the unique requirements of modeling complex non-rigid materials. One of the most important of these requirements is related to the large changes in both geometry and topology that may occur during a simulation. For example, during surgical simulation tissues may be cut, pulled, shredded and torn. Cloth is also cut and severely deformed during manufacturing and pressing. Since the underlying representation of a system of interacting particles is not affected by geometry or topology, particle modeling promises to be very robust in the face of these changes.

Particle-based modeling is also well suited to modeling the non-linearities inherent in such materials as soft biological tissue. For example, cartilage is a biphasic material consisting mostly of liquid supported by a fibrous mesh [13]. The effective porosity, and therefore the amount of internally generated viscous damping of this mesh, is greatly increased when the cartilage is placed under compression. This material nonlinearity can be handled naturally in a particle model, since the model can be constructed to contain a representation of the two components of the material's microstructure, allowing the interactions of chains of fiber particles, with freely moving fluid particles to be modeled directly.

Having a representation of the microstructure of a material is essential for the modeling of certain responses. For instance, the behavior of cloth cannot be adequately understood without representing the weave of the material, because most of the significant stresses occur at the crossing points of the threads in the weave.

The particle approach, utilizing simple microscopic primitives, and deriving complexity from large numbers of these primitives, should map naturally onto massively parallel computing architectures. Since this technique will rely on several millions of particles to represent a material, the computational requirements on a serial machine would be prohibitive. However, the approach is tailor-made to take advantage of the processors on a massively parallel machine and there it should be not only practical but quite fast. A goal and expectation of our work is to produce near constant time (at most $O(log\ n)$) algorithms for the right computing architectures.

Experiments and Results

Our initial investigations of particle-based modeling have concentrated upon the development of software tools for use in experimentation. These tools have been integrated into The Clockworks [2][4], our computer animation and simulation environment. Work so far has centered around facilities for geometric modeling, simulation, and image rendering.

For geometric modeling, we have implemented several particle based modeling tools [6]. The tool intended for small systems of particles gives the user interactive control over the placement of particles in a 3D environment. For larger systems, we have developed a kind of "injection molding" capability, which works by evaluating a conventional constructive solid geometry (CSG) model using particle sampling. This latter capability allows us to construct particle models conforming to any desired geometry.

For the "injection molding" work we have experimented with two methods for determining a final equilibrium configuration for a set of particles. One of these is deterministic and the other is stochastic. The deterministic method calculates the forces and torques generated by the inter-particle potentials, and simply applies the laws of motion to move the particles to some final equilibrium state, if one exists. In the non-deterministic approach, particles are randomly perturbed from their current position, with new positions probabilistically accepted or rejected based on a Boltzmann distribution of particle energies. This approach was first suggested by Metropolis et al. [12] and was proposed as a practical optimization technique by Kirkpatrick et al. [10].

We have completed a general simulation tool for defining and evaluating an arbitrary collection of interacting objects [3]. This provides a message-based object-oriented environment within which to experiment with various interaction rules. This tool is optimized for generality rather than for speed. Thus, its main use is for simulations of small numbers of particles, but it is allowing us to develop the experience necessary to construct successful large-scale simulations. Dynamic simulations of small particle systems have been run, studying such systems as balls bouncing in an enclosed container, and particles flowing through an externally applied vector field. It has also been used to produce an animated film showing 27 particles using a "potential well" model of the force of attraction between the particles. All of these simulations were done using the assumption that inter-particle potential drops to zero at some small distance from the particles.

In order to handle the more realistic situation of particles acting across large distances, we have implemented and are experimenting with the Fast Multipole algorithm due to Greengard [5]. This algorithm has been implemented in CLOS (Common Lisp Object System) [9][14], and should be readily ported to certain massively parallel machines. On a serial machine it provides for $O(n)$ computation of the inter-particle potentials or forces. On a massively parallel machine, the algorithm promises $O(log\ n)$ performance.

Our initial simulations looked only at particles that were spherically symmetric, producing approximately spherical objects, but more recently we have conducted a series of experiments with "structured" particles. Structured particles have a microstructure that may produce arbitrarily shaped objects. We have turned to physical chemistry and studies of the water molecule for insight into modeling on a molecular level. There have been numerous models developed that are capable of simulating the structure and dynamic behavior of water. In particular, we have focused on studying the ST2 water molecule model [16]. To date, we have created an animation showing the dynamics of the ST2 model using conventional numerical integration techniques, and we have created simulations where large numbers of ST2 particles are effectively "poured" into a container defined by a CSG model. A stochastic approach based on the Metropolis algorithm is used to determine the particle configuration.

There are several methods for generating computer graphic images from a particle-based model. Simple images may be attained directly from the particles themselves by simply projecting each particle's three-dimensional coordinates to two-dimensional image space [15]. Another approach, first suggested by Blinn [1], is to generate an iso-potential surface enclosing the cloud of particles. This can be constructed by defining field functions centered around each particle, with each particle's field contributing to a net field for the system. Any convenient iso-surface through this field can then be used as a surface for image making.

We have developed two iso-surface rendering techniques that may be used to create computer generated images of a collection of particles. The first is appropriate when using polygon-based graphics hardware to do the rendering, and is based on algorithms by Wyvill et al. [18] and Lorensen [11]. Here, the system's iso-surface is approximated by a polygon mesh. For high quality images, a ray-tracing algorithm was developed in our laboratory [17] that renders the system's iso-surface directly.

Our particle-based model of cloth attempts to capture the dynamics of the draping behavior of cloth through the definition of energy functions defined at node points (particles) that represent thread crossings. The evaluation of the dynamics is achieved in a two step process. The first step calculates, for one time step, the velocities and positions of the particles by solving force-based differential equations with inter-particle constraints removed. The second step enforces the inter-particle constraints using stochastic energy-based methods. Since the central operation of our stochastic evaluation method minimizes energy functions using the Metropolis algorithm, our initial investigations into a parallel implementation of particle-based models focused on a parallel version of the this algorithm.

A simplified version of our cloth model has been implemented in Thinking Machine's *Lisp for the CM-2. The cloth is represented by a topologically 2-D mesh of points moving in 3-D Euclidean space. Each particle is mapped to a processor and is connected to its four nearest neighboring particles, except along the edges of the mesh. An energy can be calculated for each particle based on its location and the locations of its neighbors. It is this energy that we minimize with the Metropolis algorithm.

Once the model was implemented in *Lisp, tests were conducted to determine if it was necessary to impose a "coloring" scheme on the 2-D mesh of processors. A coloring scheme for the parallel Metropolis algorithm would break the processors into a number of independent "colored" groups. In order to properly implement the Metropolis algorithm as originally defined, the energy function associated with each member in a particular group should be independent of all the other members of that group. Each group of processors and their associated data must be modified serially, one group at a time. A four-connected 2-D mesh of processors, where each processor only depends on its nearest neighbors, can be split into two independent groups in a checkerboard fashion.

We tested the minimization of our model using a parallel Metropolis algorithm in *Lisp both with the coloring scheme and without. Our initial investigations showed that there was no significant difference in the results of these two methods. Both methods brought the model down to approximately the same minimum configuration in approximately the same number of iterations. Since each iteration in the colored approach takes twice as many operations as the "uncolored" approach, it appears that perturbing and modifying every particle in parallel is the preferable method for implementing the Metropolis algorithm for our application.

Discussion

Our efforts to date have provided us with a conceptual framework, and software tools for the study of particle systems. Our current efforts are focused upon developing the theory, methodologies, and algorithms necessary to make the technique both practical and useful. Ideas for this effort are being drawn from work in cellular automata, neural networks, parallel programming, statistical physics, and traditional mechanics.

There are many problems to be solved and open issues to be addressed before particle modeling becomes a practical simulation and engineering tool. The most important of these problems is that of developing a methodology for the establishment of microscopic particle parameters that will give rise to the desired macroscopic behavior. The problem is clearly NP-Complete, but near optimal solutions to the problem should be attainable. Other important open problems are the development of fast algorithms for updating particle neighbor lists, and high-speed global information exchange. We are also concerned with elegant approaches to introducing orientation sensitivity into particle-particle interactions, tradeoffs between using a discrete or continuous representation of particle states, and developing fast particle-particle and particle-object collision detection algorithms.

A large effort is being devoted to studying and applying the Fast Multipole algorithm [5] for evaluating particle systems. We are particularly interested in investigating how to exploit the hierarchical structures inherent in the algorithm for rendering. These structures provide spatial information that may be used to rapidly create iso-surfaces of the potential field. Direct ray – iso-surface intersections may also be based on the hierarchical spatial subdivisions of the Fast Multipole algorithm. The algorithm is not only providing a technique for the rapid evaluation of inter-particle potentials, but also promises to be the foundation for a new class of rendering algorithms.

We are currently designing an implementation of our modeling approach for the Connection Machine [7]. On the Connection Machine each particle will be mapped into a virtual processor. Each virtual processor will communicate directly with a certain number of 'adjacent' virtual processors in order to compute near field effects. Far field effects will be calculated using the Fast Multipole algorithm requiring an additional hierarchical organization on all of the virtual processors. Once near and far effects have been calculated for each particle, every processor would then respond uniformly and simultaneously to the potentials acting upon it.

Acknowledgements

This work was conducted at the Rensselaer Design Research Center, Michael J. Wozny, director. This work is partially supported by the RDRC's Industrial Associates Program, DLA contract # DLA900-87-D-0016 NSF grant # CDR-8818826 and DARPA contract # DACA76-88-C-0012.

REFERENCES

[1] Blinn, J.F., "A Generalization of Algebraic Surface Drawing," *ACM Transactions on Graphics*, Vol. 1, No. 4, pp. 235-236, 1982.

[2] Breen, D.E., P.H. Getto, A.A. Apodaca, D.G. Schmidt and B.D. Sarachan, "The Clockworks: An Object-Oriented Computer Animation System," *Eurographics '87 Proceedings*, pp. 275-282, 1987.

[3] Breen, D.E. and V. Kühn, "Message-Based Object-Oriented Interaction Modeling," *Eurographics '89 Conference Proceedings*, pp. 489-503, 1989.

[4] Getto, P.H. and D.E. Breen, "An Object-Oriented Architecture for a Computer Animation System," *The Visual Computer*, Vol. 6, No. 2, pp. 79-92, 1990.

[5] Greengard, L.F., *The Rapid Evaluation of Potential Fields in Particle Systems* (The MIT Press, Cambridge, MA, 1987).

[6] Hersh, J.S., "Tools for Particle Based Geometric Modeling," *TR-88050, Rensselaer Design Research Center*, Rensselaer Polytechnic Institute, 1988.

[7] Hillis, W.D., *The Connection Machine* (The MIT Press, Cambridge, MA, 1985).

[8] House, D.H. and D.E. Breen, "Particles As Modeling Primitives For Surgical Simulation," *11th Annual International IEEE Engineering in Medicine and Biology Conference Proceedings*, pp. 831-832, 1989.

[9] Keene, S.E., *Object-Oriented Programming in Common Lisp (A Programmer's Guide to CLOS)* (Addison-Wesley, Reading, MA, 1989).

[10] Kirkpatrick, S., C.D. Gelatt and M.P. Vecchi, "Optimization by Simulated Annealing," *Science*, Vol. 220, No. 4598, pp. 671–680, 1983.

[11] Lorensen, W.E and H.E. Cline, "Marching Cubes: A High Resolution 3D Surface Construction Algorithm," *SIGGRAPH '87 Proceedings*, pp. 163-169, 1987.

[12] Metropolis, N., A.R. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, Vol. 21, No. 6, pp. 1087-1092, 1953.

[13] Mow, V.C., M.K Kwan, W.M. Lai and M.H. Holmes, "A Finite Deformation Theory for Nonlinearly Permeable Soft Hydrated Tissues," In *Frontiers in Biomechanics* (G.W. Schmid-Schonbein, S.L.Y. Woo, B.W. Zweifach, eds.), 1986.

[14] Müller, L. and W. Müller, "An Object-Oriented Implementation of The Fast Multipole Method," *TR-90030, Rensselaer Design Research Center*, Rensselaer Polytechnic Institute, 1990.

[15] Reeves, W.T., "Particle Systems - A Technique for Modeling a Class of Fuzzy Objects," *SIGGRAPH '83 Proceedings*, pp. 359-376, 1983.

[16] Stillinger, F.H. and A. Rahman, "Improved Simulation of Liquid Water by Molecular Dynamics," *Journal of Chemical Physics*, Vol. 60, No. 4, pp. 1545-1557, 1974.

[17] Tonnesen, D., "Ray Tracing Implicit Surfaces Resulting From the Summation of Polynomial Functions," *TR-89003, Rensselaer Design Research Center*, Rensselaer Polytechnic Institute, 1989.

[18] Wyvill, G., C. McPheeters and B. Wyvil, "Data Structures for Soft Objects," *The Visual Computer*, Vol. 2, No. 4, pp. 227-234, 1986.

SENSING FOR AUTOMATED GARMENT HANDLING

By

Julianne Boyd

A Thesis Submitted to the Graduate
Faculty of Rensselaer Polytechnic Institute
in Partial Fulfillment of the
Requirements for the Degree of
MASTER OF SCIENCE

Approved:

Robert B. Kelley

Robert B. Kelley
Thesis Adviser

Rensselaer Polytechnic Institute
Troy, New York

May 1991

CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENT

I wish to express my sincere appreciation to Dr. Robert B. Kelley for the time and effort he invested in me over the period of this project.

I would like to acknowledge the CIRSSE and CMP students who have helped to make my experience at RPI an enjoyable one.

My thanks to Tom for helping me to keep the appropriate perspective on life and for his endless support.

And finally, thanks to my parents for their love and patience throughout the years.

Support for this research is provided through grant 900-87-D-0016-0006 from the Defense Logistics Agency.

ABSTRACT

The Sensing for Automated Garment Handling project was conceived as part of a larger project to automate the process of pressing trousers at garment manufacturing sites. The automated pressing system is to replace the current method of manually pressing the trousers with a fully integrated, automated trouser pressing system. This paper describes the efforts put forth to produce sensing solutions for the automated system.

The sensing requirements include detection of trouser seams for seam alignment and detection of wrinkles for removal of wrinkles. The seam sensing utilizes a lighting scheme which casts a shadow across the seam that is detectable by the camera. Applying convolutions, histograms and thresholding functions, the seam contour can be extracted from the image. A Hough Transform approximation to the seam line produces the location of the seam. Determination of the in-seam and out-seam locations and the camera locations and resolutions produces the difference distance for driving the roller and stepper motor alignment device.

The wrinkle detection technique utilizes structured light to extract three dimensional information about the wrinkles. The wrinkle detection is performed on the trousers which are placed on a curved press surface. The curvature contour of the press surface is extracted from the structured light image of the press surface and the curvature contour of the wrinkled garment is extracted from the structured light image of the garment on the same curved press surface. The difference of the two contours represents the wrinkles present in the garment. The difference image can be processed to obtain wrinkle height, width and location information, as well as wrinkle orientation information.

CHAPTER 1
INTRODUCTION

1.1 Justification

The Sensing for Automated Garment Handling project was conceived as part of a larger project to automate the process of pressing trousers at garment manufacturing sites. The automated pressing system is to replace the current method of manually pressing the trousers which requires a worker be exposed to extreme heat and humidity and be well trained to operate the pressing equipment. The automated pressing project was proposed to alleviate labor problems due to the adverse working environment and the quick turnover of laborers.

1.2 Goals and Project Groups

The automated pressing project goal is to produce a fully integrated, automated trouser pressing system for installation at garment manufacturing sites. To produce such a system requires research and development efforts in several different areas: sensing, modeling of materials and garments, grasping and manipulation of garments, and integration of these three areas. This paper describes the efforts put forth to produce sensing solutions for the automated system.

1.3 Background

1.3.1 Automation in the Garment Industry

Automation in the garment industry is not a new topic. Units to cut and sew cloth pieces are commonly utilized at many manufacturing sites. However, automatic handling of garment pieces which are fully assembled is an area which received very little attention prior to The Automated Handling of Garments for

Pressing project.

The concerns involved in automatically handling fully assembled garments are quite different from those involved in the automatic cutting and sewing operations. Cutting operations are typically performed on flat, vacuum tables, simplifying transport of the cloth and the sensing needs of the system. Automated sewing operations often utilize simple point detectors to locate the edge of the cloth.

Fully assembled garments require dramatic increases in the complexity of the automated system handling and sensing requirements. Manipulation of the garment becomes difficult and requirements for wrinkle detection, force sensing and more sophisticated garment parts location sensing become mandatory.

It is well known that garment manufacturers prefer low-cost, simple solutions to problems. In addition, the manufacturing sites usually have very restricted physical space. Thus, in designing automated garment manufacturing systems, it is necessary to keep costs low and refrain from implementing large, complex solutions.

1.3.2 Material

The difficulty in automating a garment manufacturing process lies in the material. Slight variations in the thread content of a material will cause the material to react quite differently, requiring different applied tension for manipulation, producing different reflectance properties when viewed by a camera and having different wrinkling properties. Assembling a garment increases the complexity of the manipulation and sensing of the material by applying new tensions to the material.

Various properties of flexible materials have been studied, but no concrete method of accurately characterizing materials has been developed. Thus, the task of automating the manipulation of fully assembled garments is quite difficult.

1.3.3 The Pressing Process

The pressing of trousers is performed to place permanent creases in the center of the front and back panels of each trouser leg. The process is also performed to remove any puckers present in the garment. Due to the high temperature of the steam from the press, any creases placed in the trousers due to the pressing process are permanent. It is therefore imperative to remove any wrinkles that may be pressed into the trousers as creases during the pressing process.

The pressing unit consists of a horizontal, curved, vacuum surface (referred to as the buck) for placing the trousers on, a similar curved top closure to distribute the steam, and a programmable unit for varying the temperature, amount of steam, amount of vacuum and length of the pressing cycle.

CHAPTER 2
BACKGROUND

2.1 Images

Charge Coupled Devices (CCDs) are typically used for camera image planes. These CCDs are 2-dimensional square arrays of evenly spaced picture elements (pixels) which are light sensitive. The amount of charge at each pixel site is directly related to the amount of light incident on that pixel over the period allotted for charging the pixels.

A 2-dimensional image obtained from a CCD sensor is represented by a function $f(x, y)$ where x and y denote the horizontal and vertical location of a pixel in the image, and $f(x, y)$ denotes the image intensity value of the same pixel. A typical CCD camera will contain a sensor with a square array of pixels of size 256 by 256 or 512 by 512. The image origin, $x = 0, y = 0$, is designated as the upper left hand corner of the image.

The image intensity value refers to the amount of light incident upon the pixel and is generally quantized to $2^8 = 256$ grey levels. $f(x, y) = 255$ denotes the brightest (maximum) amount of light at a location x,y, and $f(x, y) = 0$ indicates the darkest (minimum) amount of light at location x, y.

2.2 Image Operations

2.2.1 Threshold

The thresholding operation creates a binary image from image points $f(x, y)$. The operation is performed by setting all pixel intensity values equal to or above a given threshold to a given value, A, and all those below the threshold to a given value, B.[1]

$$
\begin{aligned}
g(x,y) &= A \text{ for } f(x,y) \geq threshold \\
&= B \text{ otherwise} \qquad (2.1)
\end{aligned}
$$

The use of double thresholding to create an intensity band is also useful.

$$
\begin{aligned}
g(x,y) &= A \text{ for } f(x,y) \geq threshold1 \text{ and } f(x,y) \leq threshold2 \\
&= B \text{ otherwise} \qquad (2.2)
\end{aligned}
$$

2.2.2 Histogram

The histogram function produces the frequency of occurrence of each grey level in the image. For an image with n grey levels, the histogram is denoted $h[j]$, $0 \leq j \leq (n-1)$, where j denotes a particular grey level and $h[j]$ denotes the number of occurrences of grey level j in a particular image.[1]

2.2.3 Convolution

The discrete convolution operation is a method of filtering the image to produce, for example, lower or higher spatial frequency images, or to extract particular features from the image. The operation involves overlaying a mask of a desired size on an image and summing the products of the values of the mask and the image grey level values. The new pixel value is the sum of the products. The operation is repeated as the mask is shifted pixel by pixel over the entire image.

$$
g(x,y) = \sum_{i=\frac{-(M-1)}{2}}^{\frac{M-1}{2}} \sum_{j=\frac{-(N-1)}{2}}^{\frac{N-1}{2}} h(x-i, y-j) f(i,j) \qquad (2.3)
$$

where $h(i,k)$ is the convolution mask, $f(x,y)$ is the original image and $g(x,y)$ is the convolved image. The mask size is given by the summation values, M and N, and is generally square with an odd number of values.

A uniform mask with each mask value equal to 1/(total number of mask values) will perform an averaging over an M by N region of the image. This averaging will appear as a smoothed version of the image.

Extracting edges (enhancing high frequency image information) is performed by weighting certain mask values more than others. Vertical edges can be enhanced through the use of a mask with weights to the right of the center that differ from those to the left of the center. Horizontal edges can be enhanced through the use of a mask with weights above the center of the mask that differ from the weights below the center of the mask.

Lines can be enhanced through the use of a mask with a band of weights of the width of the line and at the anticipated orientation of the line.

Numerous masks have been developed. including the Sobel operator and the Roberts operator, which utilize discrete approximations to the gradients and laplacian operators to extract edge information.[2]

2.2.4 Correlation

The correlation operation is performed to determine the amount of similarity between two functions. The operation integrates the product of the two functions over a given area and produces a correlation value which is directly proportional to the amount of similarity between the two functions.

In image processing, the functions are represented by images or templates and the correlation operation is discretized. If an image is correlated with a template, the correlation output will be maximized for those portions of the image which are most similar to the template. Correlation is performed by shifting the template across the image and determining the correlation for each location.

$$g(x,y) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} h(x+i, y+j) f(i,j), \tag{2.4}$$

where $f(i,j)$ and $h(i,j)$ denote the original image and template, respectively. M and N define the size of the template $h(i,j)$, and $g(x,y)$ denotes the resultant correlation value for each (x,y) pair.[2]

2.3 Data Interpolation

2.3.1 Hough Transform

The Hough transform is a method for producing a best-fit line from data that may be noisy or corrupted. For each data point, the transform searches through its 2-dimensional parameter space to determine which slope-intercept combinations (or radius-angle combinations) are valid for creating a line containing the given point. The accuracy of the solution is proportional to the discretization of the parameters; thus the solution is a best-fit line for the given discretization and range of the parameters.

A line may be represented in cartesian coordinates as:

$$y = mx + b \tag{2.5}$$

where x,y denote the data point, m denotes the slope and b denotes the y-axis intercept.

A line may be represented in polar coordinates as:

$$\rho = x\cos\theta + y\sin\theta \tag{2.6}$$

where x,y denote the data point, ρ denotes the radius, and θ denotes the angle.

An accumulator for each parameter pair tracks the number of valid data points that solve the line equation. After completing the search, the largest accumulator value indicates the best-fit slope-intercept combination (or radius-angle combination).

The polar coordinate representation is valid for all possible lines, while the cartesian representation is not valid for vertical lines with slopes approaching infinity.[1]

2.3.2 Cubic B-Splines

B-splines are interpolation techniques which produce approximations to sets of data points. B-splines have an advantage over other interpolation techniques in that B-splines are simple to formulate. and they produce curves that are easy to manipulate and have properties that reflect those found in natural curved surfaces.

An nth order B-spline uses a local neighborhood of n points to produce a polynomial approximation of the $(n-1)th$ order. Cubic B-splines, the most commonly used splines, produce functions whose first and second derivatives are continuous.

The B-spline is computed according to the following formulation:

$$X(s) = \sum_{i=0}^{n+1} v_i B_i(s) \tag{2.7}$$

$X(s)$ is given by the expression where the $n+2$ v_i are the n coefficients determined from the n data points, plus an additional two v_i obtained from boundary conditions. $B_i(s)$ represents the cubic polynomial basis functions that are non-zero only over the interval $(i-2)$ to $(i+2)$. The basis functions are identical, but shifted, functions.

The basis functions can be divided into four spans such that each span contains $\frac{1}{4}$ of each of the basis functions and is designated $C_{i,j}(s)$, $0 \leq j \leq 3$. Then:

$$x(s) = C_{i-1,3}(s)v_{i-1} + C_{i,2}(s)v_i + C_{i+1,1}(s)v_{i+1} + C_{i+2,0}(s)v_{i+2} \tag{2.8}$$

Figure 2.1 illustrates one span of the basis functions.[3]

2.4 3-D Sensing Methods

Numerous methods have been developed to extract 3-dimensional information from 2-dimensional images. Three methods commonly implemented are range finding, stereo vision and structured lighting.

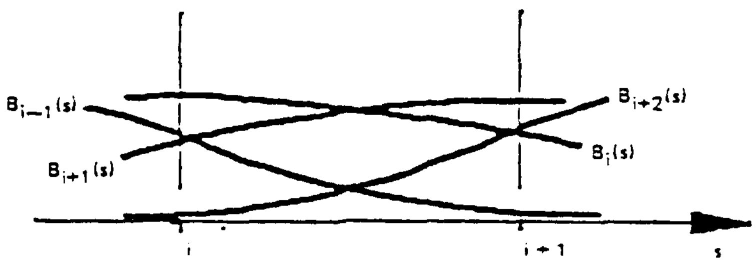


Figure 2.1: Four non-zero basis functions over single span

2.4.1 Laser Rangefinder

Laser rangefinders are based on laser technology and time-of-flight methods. The first time-of-flight method utilizes phase-shift measurements of the laser pulse between leaving the source, bouncing off the object and returning to the detector. The second method utilizes total-time measurements from the time the pulse leaves the source until it arrives at the detector.

The range-finders produce high density range maps with high quality spatial and range resolutions. However, the units are extremely expensive due to the requirement for quality scanning equipment. The laser scanning process requires fast and accurate movement of mirrors to project the beam across the scene, which results in costly mechanical parts.

2.4.2 Stereo Vision

Stereo vision utilizes two cameras viewing the same scene from different locations. Given the location of the two cameras, and the correspondence between the points in the two images. the depth of the objects in the images can be determined from the geometry. The difficulty lies in determining the correspondence between each point in the first image with the corresponding point in the second image.

Figure 2.2 illustrates the stereo vision configuration. Ideally, the two cameras' optical axes should converge near the scene object of interest to maximize the number of points in common between the two images. This method, known as convergent stereo, reduces, but does not alleviate, the correspondence problem between the two images.

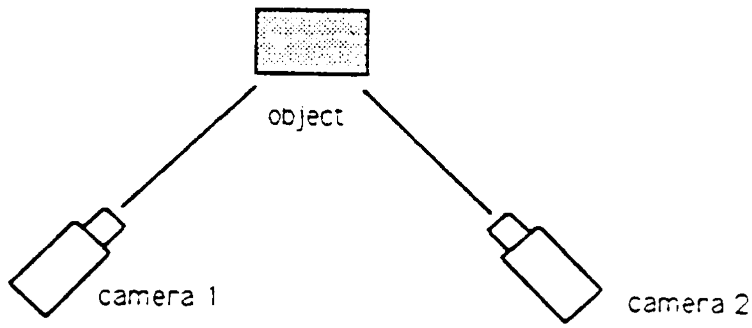


Figure 2.2: Stereo vision configuration

2.4.3 Structured Lighting

Structured lighting is a technique similar to stereo vision, but the second camera is replaced by a light source which projects a known light line or pattern onto the scene being viewed by the first camera. Again, given the location of the camera and light source, and the correspondence between the points in the two images, the depth of the objects in the images can be determined. However, the correspondence problem is simplified because the light pattern is known and can be tailored to the application.

The projection of a line of light onto a non-flat surface will result in geometric distortions of the light pattern at changes in the relative depth of the surface being illuminated. A surface with a constant first derivative in the direction of the line will exhibit smoothly varying changes in the line of light projected on that same surface. Points on the object surface at which the first derivative is not constant, i.e., non-smooth changes in the surface, will produce discontinuities in the projected line of light. In general, the projected line of light will follow the contour of the surface on which it is projected.

Figure 2.3 illustrates the projection of a single line onto two objects. The depth of the sphere's surface changes at a constant rate and thus the lightline distortion is smoothly varying. The lightline displays a discontinuity at the edge of the right and front surface of the cube. For an alternative view of the cube with just the front surface visible, as illustrated in figure 2.4, a jump in the lightline would appear at the edge of the cube.

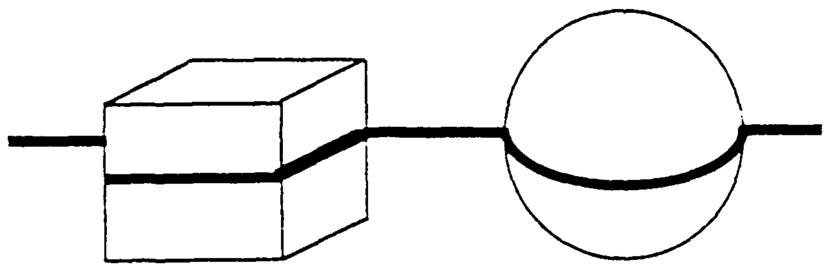


Figure 2.3: Structured Lighting - Side Perspective

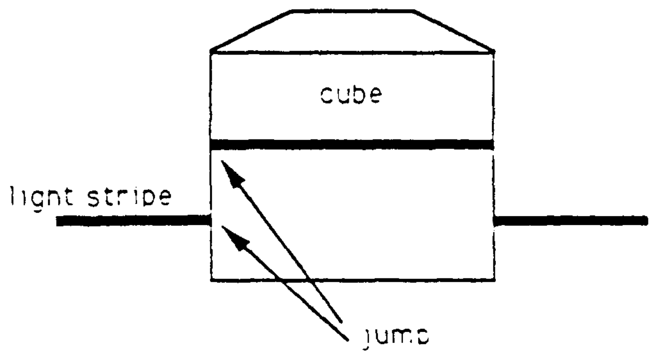


Figure 2.4: Structured Lighting - Front Perspective

The light source utilized to project the light may be one of several devices: a laser, a projector, or any other high intensity light source. A laser and scanner provide high intensity light patterns of varying complexity and offer re-programmability.

A projector, a less costly option, may utilize high quality cut-glass slides, but will not produce as high intensity light patterns as the laser. The third option, high intensity light sources, produce lower intensity lighting than the laser and less complex patterns than the laser or projector.

CHAPTER 3
DEFINITION OF REQUIREMENTS

3.1 System Requirements

The fully automated pressing system is required to operate at rates equal to or greater than manual operators currently work at and produce pressed and creased pants of equal or better quality than those produced manually. The system will initially handle only wool and wool-blend dress trousers with busted seams as illustrated in figure A.1 in Appendix A.

The pressing process consists of six stages: alignment of trouser leg seams, transfer of trousers to the pressing buck, detection of wrinkles on the trousers, removal of wrinkles, pressing of trousers and removal of trousers from the buck. Each of these stages will be automated.

3.2 Sensing Requirements

Evaluation of the above pressing process resulted in the proposal of the following sensing requirements: seam sensing, wrinkle sensing and force sensing.

3.2.1 Seam Sensing

The seam sensor is utilized for locating the inner and outer seam of each pant leg for proper alignment prior to pressing the garment. Proper seam alignment requires aligning the inner and outer seam at the cuff of the trouser leg and at or above the knee of the trouser leg. The proper alignment of the seams is crucial for producing front and back creases that lie in the center of the trouser legs. The seam sensing and alignment take place prior to transfer of the trousers onto the press buck.

3.2.2 Wrinkle Sensing

Wrinkle sensing occurs after placement of the trousers onto the press buck and is utilized to ensure that no significant wrinkles are present prior to pressing the garment. If wrinkles of significant size are present, the height, width, orientation and location of the wrinkles are determined.

3.2.3 Force Sensing

Force sensing is utilized to ensure that the proper amount of force is applied to the garment to reduce the risk of damaging the garment and to ensure that enough tension is applied to minimize the number of wrinkles. Force sensing was not developed in this project, but will be developed once the grasping/manipulation devices have been fully designed and implemented and the force sensing requirements fully defined.

CHAPTER 4
SENSING METHOD PROPOSALS

Several methods were initially proposed and pursued for each of the two sensing requirements. Investigation of the different methods led to a single method for the wrinkle sensing and a standard method for the seam sensing with a backup method to handle a unique and infrequently encountered material.

Definitions and illustrations of garment-related terms mentioned throughout this chapter may be found in Appendix A.

4.1 Seam Sensing

Seam sensing requires accurate location (+/- 0.125 inches) of the seams on both sides of the trouser leg for alignment. Many methods have been proposed for seam sensing of which most can be eliminated due to measurement inaccuracy, inconsistent performance or implementation complexity. The proposed methods include mechanical location of the seam. mechanical detection of multiple plies. transmitted light through seam detection. one ply versus two plies detection. transmitted light detection. laser light technique. and exterior seam shadow detection.

4.1.1 Mechanical Location of the Seam

The location of the seam can be determined by determining where the two pieces of fabric meet for a seam. Mechanically this can be performed by sliding a knife-edge along the surface of the fabric until the resistance of the folded seam is sensed. This same action could be performed to locate the seam from the opposite direction as a confidence check. The opposite seam would be located in the same manner. Some experiments have been performed that indicate this technique could be used for heavier weight fabric which have well-formed seams. Such a device might be incorporated as an integral part of the rotating seam alignment mechanism which is described in the Grasping and Manipulation section. However, improperly busted seams (not an uncommon problem) would prevent the apparatus from operating properly.

4.1.2 Mechanical Detection of Multiple Plies

The detection of the differential thickness of multiple plies formed by the seam might be detected by mechanical means. An instrumented roller could be used to measure the thickness of the trouser leg material as it is being rotated by the seam alignment mechanism. Some experiments have been performed that indicate this technique could be used for the heavier weight materials to locate the multiple ply selvage areas. However, improperly busted seams would prevent the apparatus from properly working. In addition, seam location would be inaccurate for seams with uneven amounts of selvage on either side of the seam.

4.1.3 One Ply versus Two Plies Interior Light Method

The One Ply Versus Two Plies method is similar to the method described above, but utilizes an interior light in the trouser leg to detect the location of the selvage edges. The light is a high intensity gas lamp with an aluminum housing and two plastic diffusion plates to permit simultaneous illumination of both the in-seam and out-seam with the same light. The overall dimension of the light source is approximately $3x4x0.75$ inches for easy placement within the interior of the trouser leg. The light intensity is great enough for detection by an exterior camera: the light passes easily through a single ply of fabric, but is more attenuated by the two plies of fabric located on either side of the trouser seam. The camera image of the transmitted light then contains a dark central strip, indicating the two non-transparent plies, with lighter areas to either side of the strip, indicating the one transparent ply. The seam is estimated to be at the mid-point of the two-ply vertical strip, as illustrated in Figure 4.1.

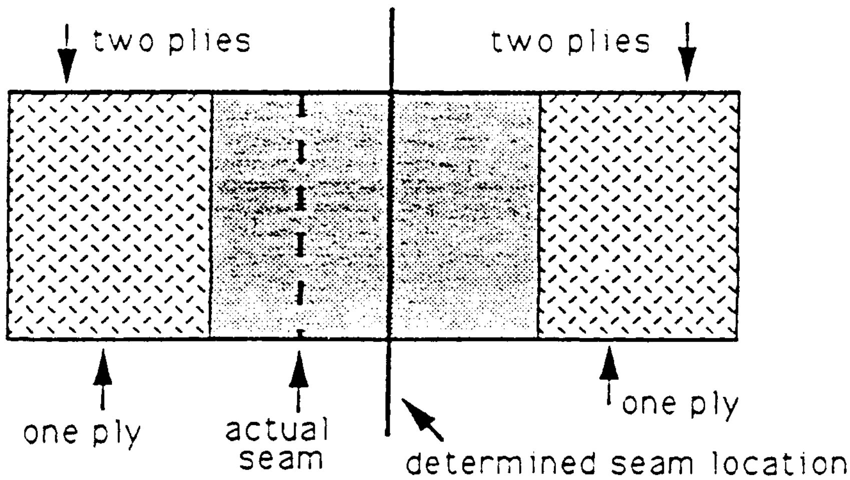


Figure 4.1: One Ply versus Two Plies Interior Light Method

Testing of this method on a variety of trousers produced very inaccurate seam location results, due to improperly busted seams and an uneven amount of selvage on either side of the seam in the majority of trousers tested.

4.1.4 Transmitted Light Through Seam Detection

The Transmitted Light method utilizes the same light as the One Ply Versus Two Plies method. The light fixture is placed on the inside of the trouser leg to transmit light through the seam gaps for location of the seam. The application of an appropriate tension force across the seam of the trouser leg will cause the seam to separate slightly and create gaps between the threads in the seam stitching. The light fixture on the inside of the leg will transmit light through the seam gaps that will appear on the exterior as a broken line of light in the darker two ply area, as illustrated in Figure 4.2. A camera can be used to acquire the image of the seam, and appropriate algorithms applied to extract the broken line of light from the image. This method requires the insertion of the light fixture into the trouser leg and more extensive computer processing of the image data.

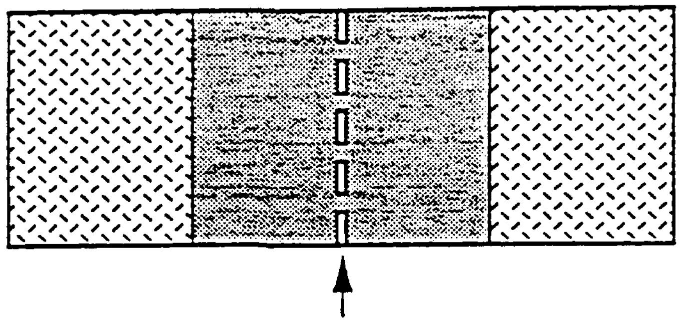


Figure 4.2: Transmitted Light Through Seam

4.1.5 Laser Light Method

The laser technique utilizes a single laser line scanned across the seam. The line of light is distorted at the seam due to the change in the trouser surface at the seam, producing an extremely accurate seam location value. This method was tested on many pairs of trousers. It was observed that, in some instances, the surface change produced by the presence of the seam is not significant enough to produce a detectable distortion in the light. This is most probably due to either the light weight of the fabric or the high sewing tension of the thread binding the seam.

4.1.6 Seam-Shadow Detection

The seam shadow technique utilizes an exterior light at an angle to the trouser leg to cast a shadow along the length of the trouser seam. An infrared filter is added to the camera lens to aid in distinguishing the cast shadow from any stripes or other fabric pattern markings that are parallel to the seam direction. The infrared filter blocks light below 850 nanometers, which includes the majority of color variations in materials. The external light used to create the shadow contains enough light energy above 850 nanometers (light not blocked by the filter) to be effective in illuminating the trousers. The camera is positioned normal to the trouser leg with a field of view of approximately 8 by 8 inches and the lighting is positioned at an angle to the trousers to cast a shadow across the seam, as illustrated in Figure 4.3.

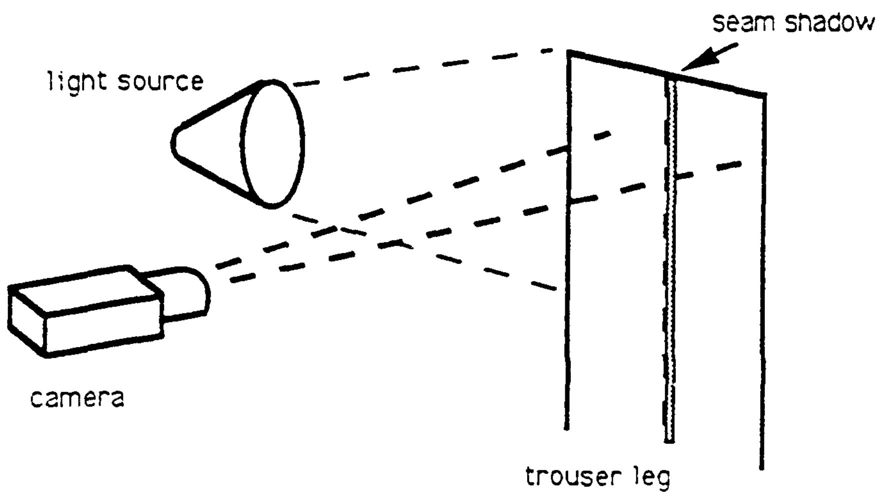


Figure 4.3: Seam shadow technique configuration

Testing of this method proves it is fast, simple and reliable in all but one situation. The exception is caused by the presence of vertical red threads in trousers. The infrared filter passes red light above 850 nm through to the camera; the red threads reflect enough energy above 850 nm to appear on the image as vertical lines parallel to the seam shadow.

Despite this one anomaly, the seam shadow technique appears to be the most accurate and reliable of all techniques proposed and was the chosen technique for further research. The problem of handling fabric with vertical red stripes has been handled as an anomaly and will most likely be solved utilizing other light source-filter combinations or the Transmitted Light Through Seam Technique as a backup method when trousers with red vertical threads are encountered.

4.2 Wrinkle Sensing

The sensing of wrinkles requires recovering three-dimensional information from a two-dimensional image. The problem is a significant one, but can be simplified through the use of structured lighting and stereo vision, both well-documented methods for recovering three-dimensional surface information.

Wrinkles in a flexible material can be viewed as surface height changes across the material, whereas a flat material will display no surface height changes. The projection of a high-intensity light onto a non-flat surface will result in geometric distortions of the light pattern at points of surface height change of the material. By locating the camera and light projector at different points (commonly referred to as triangulation stereo vision), the light pattern distortions on the surface of the object are detectable in the two-dimensional image acquired by the camera. The light pattern distortions seen by the camera are proportional to the actual surface height changes of the material. The three-dimensional surface height data is recovered through processing of the two-dimensional image.

4.2.1 Laser Scanner Method

A laser scanner was originally proposed for use as the reflected light sensor for projecting the light pattern onto the fabric. The benefits inherent in using a laser are the ability to reprogram the scanner to produce a variety of light patterns, the ability to produce complex patterns, and the high intensity of the light. The created laser light patterns are designed to cover wrinkles at all angles and of all sizes. Computer programs were successfully written to produce spiral, Lissajous (figure-eight-like) and snake-like patterns. The patterns are fairly complex and require a large minimum number of points to be correctly displayed. This results in a significant amount of required time to display a single pattern, which produces a a significant amount of flicker in the viewed pattern. Such flicker is caused by the entire light pattern not being fully captured by the camera in a single image. The problem is not resolvable without utilizing more expensive scanning equipment.

4.2.2 Light Line Method

The second approach for the light source is a high intensity light source and lens which produce a high intensity line of light. When projected across a trouser leg, the ten inch wide line of light will be displaced at points of change in the fabric height. Processing of the image produces information about the height and width of the wrinkles at each image sampling point. By scanning the light line down the length of the trouser leg, wrinkle information may be extracted at the appropriate sampling frequency. Interpolation of the data between the sampling points will provide wrinkle orientation information. The configuration and scanning technique are illustrated in figure 4.4. The camera is located at a 25 degree angle to the surface and the lightline is located at a 90 degree angle for simple calibration of distances.

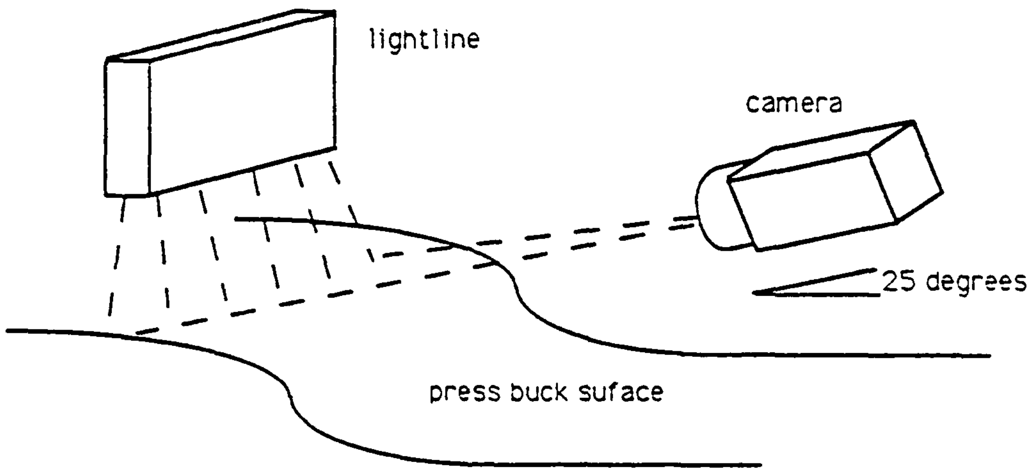


Figure 4.4: Light line scanning configuration

CHAPTER 5
ALGORITHM DEVELOPMENT

The sensing algorithms may be grouped into three operations: seam detection, wrinkle detection, and modeling of the press buck and wrinkles. The modeling algorithms were developed to prove the reliability of the methods for acquiring surface information for wrinkle detection.

The seam sensing and wrinkle sensing algorithms were originally developed using the GMF Karel^{TM} and Insight system for convenience. For processing speed, the ability to handle a wider range of operations, the ability to store a larger number of images, and future compatibility, the algorithms were re-written in C language and compiled and processed on the IBM host computer. The original Karel^{TM} algorithms proved the processing techniques operated properly and produced appropriate results. In conversion to C, several algorithms were modified and many algorithms enhanced.

The actual C-code modules for seam detection and integration and for wrinkle detection and description are described in detail in Appendices B and C, respectively. Conversion algorithms for converting images from one format and resolution to a second format and resolution are described in Appendix E. Descriptions of the development equipment may be found in Appendix D.

CHAPTER 6
ALGORITHMS FOR SEAM SENSING

6.1 Karel^{TM} Algorithms

The Karel^{TM} algorithms for seam sensing were produced to prove that the seam line of the trouser leg could be detected using the seam shadow technique. The Karel^{TM} algorithms do not attempt to locate or model the seam but do perform image processing to extract the seam from the image.

6.1.1 Image Filtering

The original image is first filtered to remove noise and enhance the vertical edges in the image (in this case, the vertical edge is the seam). The optimum filter kernel size for an image with a field of view of approximately 8 inches was determined to be 3 by 3 pixels, as illustrated in figure 6.1. Larger kernel sizes (5 by 5 pixels, 7 by 7 pixels, etc.) did not produce as sharp seam contours as did the 3 by 3 kernel. The filtered image has a gray background with a bright contour for the seam.

6.1.2 Extraction of Seam Contour

The histogram of the filtered image is then determined. The histogram produces the frequency of occurrence of each gray level intensity in the image. The

1	-2	1
1	-2	1
1	-2	1

Figure 6.1: 3 x 3 Convolution Kernel for Smoothing

histogram is used to determine where to set the threshold level in the third step of processing. The goal is to produce a binary image with the seam contour at gray level 255 on a background of gray level 0. The seam contour contains the pixels with the brightest intensity levels, and thus are found in the upper end of the histogram. Starting from the upper end of the histogram, the number of pixels per gray level is summed until a given number of pixels has been encountered. This number is determined as the approximate number of pixels that may be found in the seam contour. A given line across the image will contain approximately 256 pixels; thus the sum is determined to be complete when 200 pixels have been summed. The gray-level at which the 200th pixel is found is set as the threshold.

The third step in the processing entails thresholding the image with the gray-level found in step two. Thresholding produces a binary image of the bright seam contour against a dark background.

6.1.3 Remarks

All images processed using the above algorithms produced images with well-defined seam contours on dark backgrounds. Some noise was evident in the images, but the noise was not significant enough to hamper extraction of the contours.

6.2 C Algorithms

The C algorithms for seam sensing are similar to the Karel^{TM} algorithms. A fourth step was added to allow extraction of the seam contour location, and several steps added for integration of the vision algorithms with the stepper motor device used to align the seams.

6.2.1 Approximation to Seam Contour

To locate the seam contour following the filtering, histogramming and thresholding of the image, a Hough transform is performed on the image. The Hough transform determines the best line fit for pixels of high intensity, in this case, pixels of intensity 255. The transform works by scanning through possible slope and intercept combinations for each bright pixel. The best fit line is determined by the most common slope-intercept pair. The seam is thus modeled by the best fit line, and the slope and intercept of that line are known.

Although the seam appears as a near vertical line, the cartesian representation for a line is used for the Hough transform, where:

$$x = \frac{1}{m}y + a \tag{6.1}$$

The slope, $\frac{1}{m}$, will approach zero for a vertical line. The x-axis intercept, a, is allowed to vary along the width of the image, from 0 to 255, and designates the image point at which the vertical seam line intersects the top row of the image. The slope is allowed to vary ± 45 deg from the 90 deg ideal, in 1 degree increments.

The discretization resolution and parameter space range for the slope and intercept of the line approximating the seam can be modified in the final integrated system to optimize the accuracy of the seam location while minimizing the required processing time.

6.2.2 Remarks

Many images were successfully processed utilizing the above algorithms. The following figures are of images at each stage of processing. Figure 6.2 is the original seam shadow image. Figure 6.3 is the filtered image. Figure 6.4 is the thresholded image. Figure 6.5 is the best-fit line approximation to the curve found in figure 6.4. From comparison of figures 6.2 and 6.5 it is evident that the Hough transform produces lines which accurately model the original seam contours.

Figure 6.2: Trouser with Seam Shadow

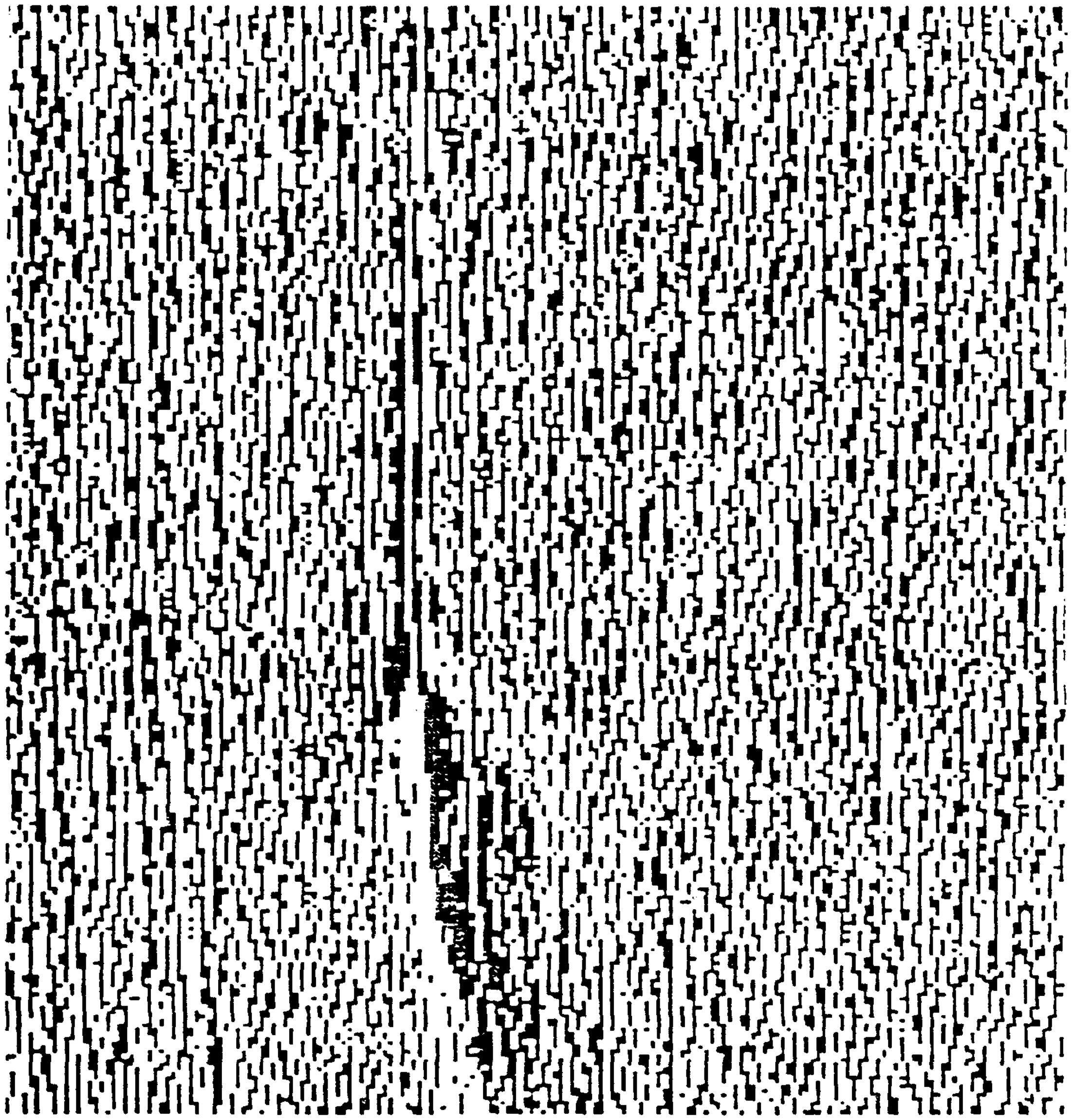


Figure 6.3: Filtered Seam Image

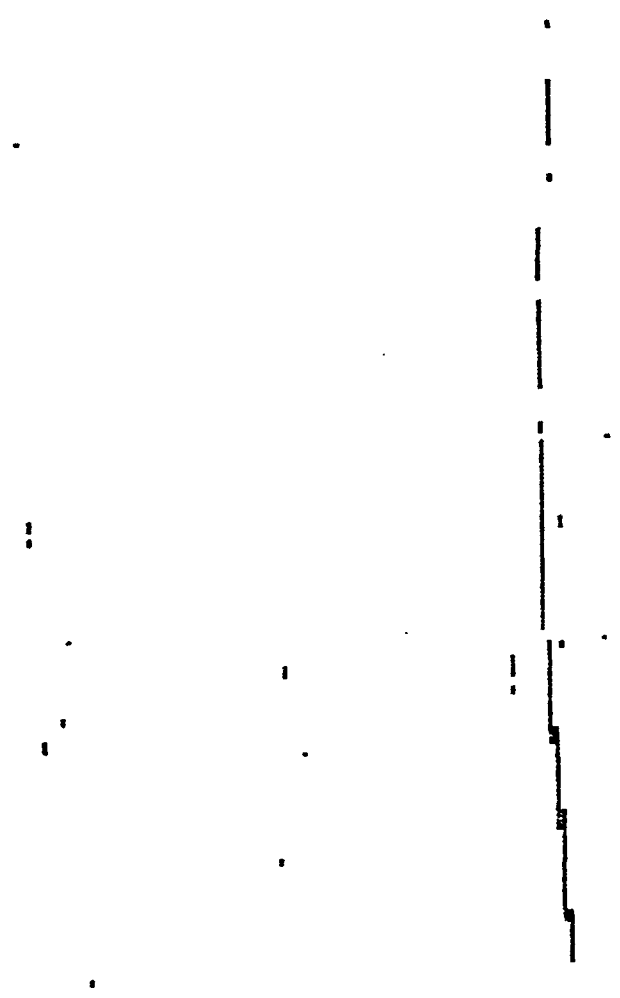


Figure 6.4: Thresholded Seam Image

Figure 6.5: Hough Transform of Thresholded Image

6.3 Seam Integration

The integrated seam alignment algorithm utilizes two images, one of the in-seam and one of the out-seam. Both images are snapped while the leg of the trouser hangs by the cuff from the seam alignment roller and stepper motor device. A quick calibration module locates a stationary calibration point in the upper band of each of the images. The two calibration locations are used to determine the image resolution and as reference points to determine the distance at which the two seams are apart.

The calibration routine performs a filtering operation which locates the right and left edge of the calibration reflective strip for each side of the alignment device. The reflective strip is a bright mark against a dark background across the top horizontal margin of the image. The filtering thus looks for intensity changes across the top rows of the image. The column value of the mid-point of the strip and the width of the strip are determined for each image.

The resolution for each image is equal to the actual width of the strip divided by the pixel width of the strip determined from the image.

$$resolution = actual\ \ strip\ \ width/determined\ \ width\ \ in\ \ pixels \tag{6.2}$$

The seam detection algorithms described above are implemented to determine the top margin intercept of each seam image. The actual offset of each seam from the desired location is determined as the difference of the strip and seam intercept locations.

$$offset1 = (strip\ \ location1 - intercept\ \ location1) * resolution1 \tag{6.3}$$

The reflectance strip locations and the intercept locations are then used to determine the actual required distance to move to align the in-seam and out-seam of the trouser leg.

$$difference\ \ distance = (offset1 + offset2)/2 \tag{6.4}$$

The drive distance is determined to be half the difference distance between the in-seam and out-seam. The sign of the distance determines the direction, clockwise or counter-clockwise, in which to drive the stepper motor.

CHAPTER 7
ALGORITHMS FOR WRINKLE SENSING

7.1 Karel^{TM} Algorithms

The Karel^{TM} algorithms for wrinkle sensing were produced to verify that the light stripe technique was appropriate for wrinkle detection. Due to the varying curvature of the buck surface, wrinkle detection requires determining the difference image of the image of the empty buck contour at a particular location and the image of the garment contour on the buck at the same location. The difference image thus contains the same wrinkle height information as would be obtained if the garment had been placed on a flat surface.

7.1.1 Reference Images

The light stripe and camera are initially scanned along the length of the empty buck such that the light stripe falls across the width of the buck surface. Figure 4.4 illustrates the scanning technique and camera and light source configuration. Reference images are obtained at particular locations (the appropriate lengthwise sampling frequency has yet to be determined) and the buck curvature information is extracted at these locations. The buck curvature is seen as a bright contour across the width of the image against a dark background.

7.1.2 Extracting the Contour

The images are convolved to enhance the horizontal curvature of the buck contour. This is performed utilizing the 3 by 3 kernel depicted in 7.1. The convolution enhances the intensity of bright horizontal lines.

The algorithm then utilizes moments to determine points of change along the

1	1	1
-2	-2	-2
1	1	1

Figure 7.1: 3 x 3 Convolution Kernel for Extracting Horizontal Contours

curvature of the image. The image is divided into 256 vertical windows (1 pixel wide by the height of the image) and moments are calculated for each window. The Karel^{TM} moment function calculates moments for a bright object on a dark background; each of the 256 windows in the image will contain a section of the bright curvature against a dark background. The moment function produces the vertical component for each of the 256 windows. Each vertical component is relative to the height of the curvature at a particular pixel location.

7.1.3 Garment Images

The above process of scanning the buck, enhancing the horizontal curvatures of the image and calculating the vertical component for each of the 256 one pixel wide windows is repeated with a garment placed on the buck surface. The images are acquired at the same locations utilized for the reference images.

7.1.4 Difference Image

The difference values are determined by finding the difference in the vertical components of the reference image and garment image for each window. These values indicate the height of the garment from the underlying buck.

7.1.5 Wrinkle Information

The difference values are then processed to obtain the desired wrinkle information. Changes in height indicate points where the wrinkles begin and end, and the height and width of the wrinkles. These values are determined by scanning through and tracking the 256 vertical difference values.

7.1.6 Remarks

The algorithm works fairly well, producing the appropriate wrinkles, their start locations, end locations, maximum height and width. However, the procedure of determining the vertical value by calculating the moments for a series of windows is not an optimal method. The sensitivity of the moment calculations is highly dependent upon the width of the window across which the moments are determined and the moment calculations are fairly time consuming. The Karel^{TM} function, MAXPT, for extracting the maximum (brightest) pixel in a window would appear to be a quicker and simpler method; but the MAXPT function requires calibration of the camera (using the Karel^{TM} calibration routine) which would not function at the angle at which the camera was located.

7.2 C Algorithms

The C algorithms for wrinkle sensing are similar to the Karel^{TM} algorithms and utilize the same light stripe and camera scanning methods as the Karel^{TM} algorithms. The steps for obtaining the difference images and the method for extracting the wrinkle information use different approaches than those developed using the Karel^{TM} language. In addition to the wrinkle size, height and location information, the orientation of the wrinkles is determined.

Figure 7.2 is an image of the light stripe projected on the empty buck.

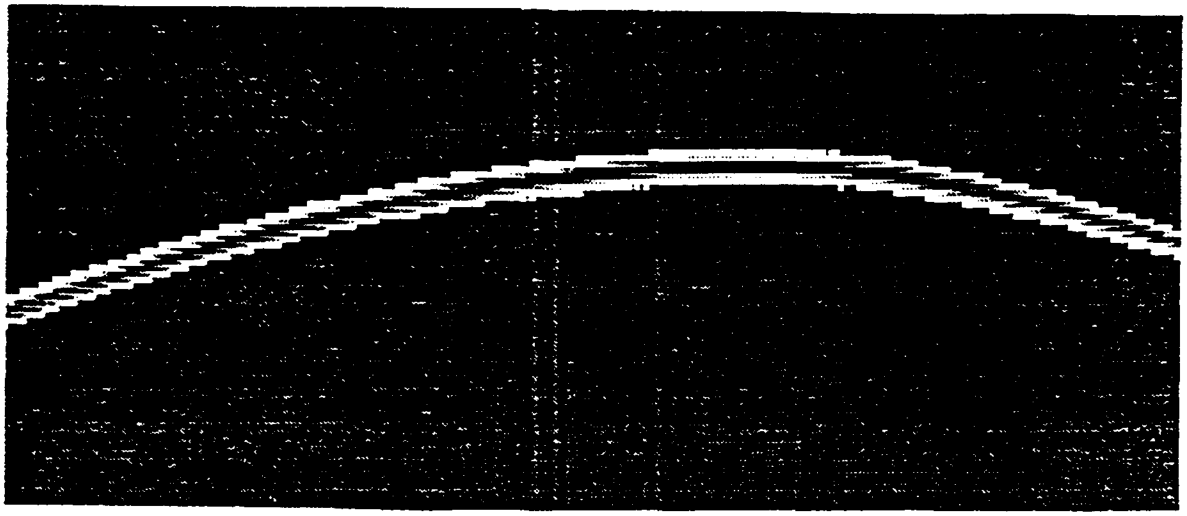


Figure 7.2: Light Stripe Projected onto Empty Buck Surface

7.2.1 Image Smoothing

The smoothing algorithm performed on the images prior to other processing is a simple 3 by 1 convolution kernel with each of the three weights equal to $\frac{1}{3}$, and is performed to remove single pixels of noise in the image. This convolution is equivalent to taking the average of every three adjacent pixels in each column and produces a smoothed version of the input image. The smaller 3 by 1 convolution kernel requires approximately $\frac{1}{3}$ the processing time of the larger 3 by 3 kernel.

7.2.2 Extraction of Contour

The buck contour is then extracted from the image by scanning each vertical column for the brightest pixel, the light stripe location approximation for that particular column. The search for the brightest pixel guarantees the center location of the light stripe due to the Gaussian shaped cross section of the light intensity. The light is brightest at the center of the light line width and decreases in intensity to each side of the center, as observed in the numerous images analyzed. The bright

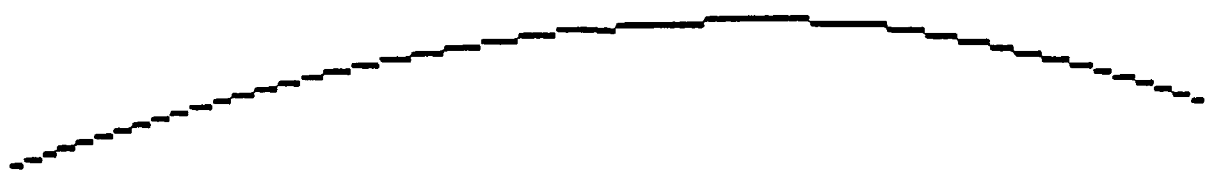


Figure 7.3: Contour extracted from light stripe projected onto empty buck

pixels' y-locations (one per column) are stored in a 256x1 reference array.

Figure 7.3 is the curve obtained from processing the image of figure 7.2; the curve is an extremely accurate estimation of the projected light stripe.

7.2.3 Garment Images

The above process of scanning the buck, filtering the image and locating the brightest pixel in each column is repeated with a garment on the buck. The images are acquired at the same locations utilized for the reference images. Figure 7.4 is and image of the light stripe projected onto a wrinkled garment on the buck, and figure 7.5 is the contour obtained from processing the image of figure 7.4; the curve is an extremely accurate estimation of the projected light stripe.

7.2.4 Difference Images

The difference information between each reference image and the corresponding image with trousers is then computed. This is performed by calculating the

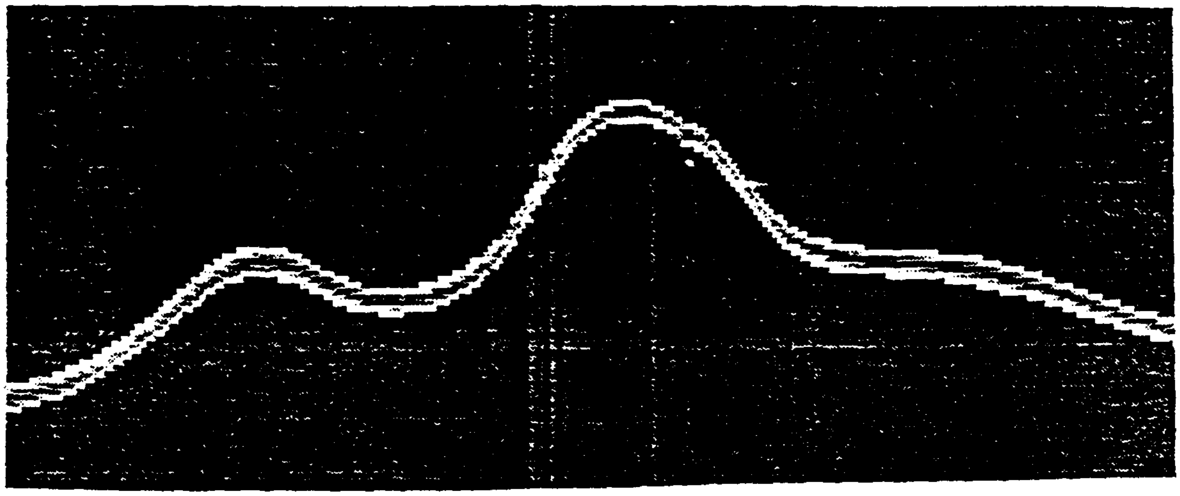


Figure 7.4: Light Stripe Projected onto Trouser Leg Placed on Buck

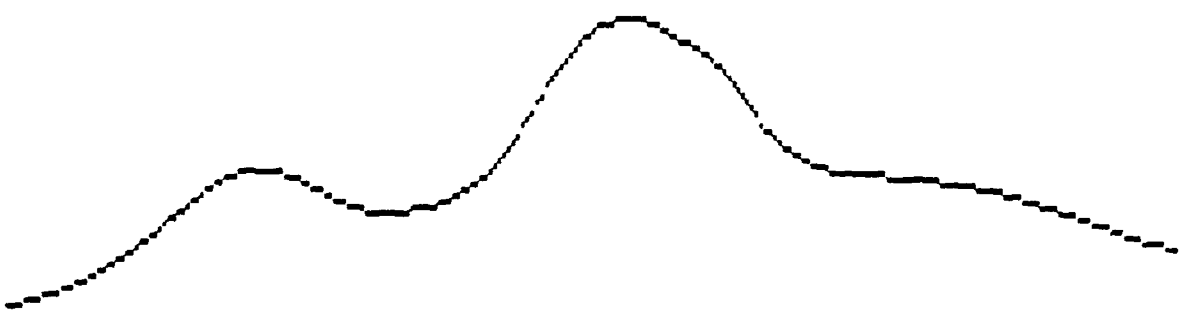


Figure 7.5: Contour extracted from light stripe projected onto trouser leg placed on buck

Figure 7.6: Difference image of buck contour and wrinkle contour

difference between the y-location of the reference image bright pixel and the y-location of the trouser image pixel location for each column. The 256 resultant differences are then stored. This yields the wrinkle height information, as displayed in figure 7.6. A flat reference value is produced which gauges the smallest difference between the empty buck and trousers on buck for all the columns.

7.2.5 Curve Approximation and Analysis

The difference curves are then approximated using Cubic B-splines to allow extraction of derivative information for evaluation of the wrinkles, if any are present.

The cubic B-spline approximations are polynomial functions which when interpreted (i.e., evaluated for each point x), produce y-values which accurately represent the original values of the function. The polynomial functions can be manipulated to produce curvature and local peak and valley information of the curves. Those points which are used to extract wrinkle information will be referred to as *evaluation points*.

The first derivative of the polynomial function describes the slope of the curve at the evaluated point. The second derivative of the polynomial describes changes

Figure 7.7: Inflection points of B-spline approximation to curve

in the first derivatives. Inflection points are points at which the first derivative, or slope, changes sign, and thus indicate a change in curvature from convex to concave. Inflection points occur at points where the second derivative is equivalent to zero.

Several methods for producing evaluation points were explored. Inflection points were originally thought to be a good evaluation criteria, but it was found that the curvature of the material did not always change at the beginning and end of a wrinkle, and that the curvature at peaks and valleys was often subtle enough that the change in curvature was not obvious. Figure 7.7 is an example of a difference image evaluated using the inflection point technique. The vertical bars above the curve indicate locations at which inflection points were found.

The second technique explored was a method which evaluates the B-spline approximation for points at which the first derivative does not equal zero. Thus, the flat areas where no wrinkles exist or plateaus in a wrinkle exist would be disregarded. However, this method produced numerous evaluation points. Figure 7.8 is an example of a difference image evaluated using this method. The vertical bars above the curve indicate locations at which the first derivative does not equal zero.

The third method explored was to evaluate the first derivatives for changes.

Figure 7.8: B-spline approximation first derivative points not equal to zero

Those points for which a neighboring points' first derivative differs from the local point's derivative, indicate changes in slope. Changes in slope occur at the beginning, end, peaks and valleys of the wrinkles, as well as other points. This approach has proven to be much more accurate than the inflection point approach, and produces less evaluation points than the second method. Figure 7.9 is an example of a difference image evaluated using the difference in first derivatives technique. The vertical bars above the curve indicate locations at which the first derivative differs between adjacent points.

Thus, the first derivatives of the cubic B-spline approximation are determined, and those points at which the first derivatives differ are labeled as evaluation points. The evaluation points are used to minimize the search for wrinkle information.

7.2.6 Wrinkle Data Extraction

The next step in the process of wrinkle detection involves evaluating the data for locating wrinkles and determining their characteristics. This is performed by searching through the evaluation points found in the above derivative evaluation process and tracking the height of the curve as well as whether a wrinkle has been

Figure 7.9: B-spline approximation points of changing slope

encountered.

The wrinkle information obtained from the search is output to an image file. The location of the beginning, end and peak of each wrinkle, as well as the maximum height and width of each wrinkle is formatted to an ascii file. Figure 7.10 is the wrinkle information file obtained from processing the difference contour of figure 7.6.

7.2.7 Wrinkle Orientation

The orientation of the wrinkles may be determined by performing a correlation between adjacent images. The correlation value is greatest for the shift which represents the maximum correlation between two images. By successively applying a correlation function to each pair of adjacent images of the garment on the buck, the shift of the wrinkles between each image can be determined and the overall orientation of the wrinkle produced.

The correlation is performed by multiplying the area under the curve representing the difference image at buck location one by the area under a shifted version of the curve representing the difference image at buck location two. Multiplication

```
IMAGE 0
Wrinkle 0 :  width=53   height=16
  beg=21  peak=51  end=74

Wrinkle 1 :  width=71   height=30
  beg=104  peak=129  end=175

Wrinkle 2 :  width=2   height=6
  beg=215  peak=216  end=217
```

Figure 7.10: Wrinkle information output file

is performed by multiplying the y-value of each pixel location of difference image one with the y-value of the corresponding pixel locations in difference image two. Each of the 256 products from the 256 pixel locations are summed to produce the resultant correlation value.

The process is repeated for 512 shifted versions of image two, where image two is first shifted 256 to the left to align its right-most pixel with the left-most pixel of image one, and finally shifted to the right to align its left-most pixel with the right-most pixel of image one. The largest of the 512 correlation values indicates the shift between the two images, and thus the orientation of the wrinkles.

Testing was performed on the correlation algorithm for trousers with wrinkles placed by hand. The success of the testing was due to the consistent orientations of all the wrinkles on the trousers. The correlation algorithm would not work well for numerous wrinkles in various orientations. However, there is reason to believe that the trouser handling device will produce wrinkles in a particular orientation as it places the trousers on the press buck. Thorough testing will be required once the final handling device is in place.

Figure 7.11 illustrates the correlation performance on a set of six curves. A negative shift value indicates that $image_{i+1}$ should be shifted to the left by the value to correlate with $image_i$, and a positive value indicates that $image_{i+1}$ should be shifted to the right by the value to correlate with $image_i$. Excluding image four, which exhibits a dissimilar wrinkle, the correlation values are fairly similar, indicating a consistent orientation.

7.2.8 Image Resolution

The original algorithms were developed for images of size 256 by 242 pixels. These images spanned an area of approximately 8 inches by 8 inches, thus providing a resolution of approximately 32 pixels/inch.

Evaluation of the wrinkle algorithms led to reducing the image resolution to decrease the amount of processing time required. An averaging algorithm was produced to create an image width of 64 pixels by averaging every four pixels. The averaging takes place following the contour extraction and operates on the 256 by 1 array of the curve locations, thus requiring only 64 averages (as opposed to the $64x64 = 4096$ averaged required for an entire image). The new resolution of approximately 8 pixels/inch produced results similar to those of the higher resolution algorithm.

The averaging of the difference image array also aided in the removal of noise by averaging noisy pixels (i.e. pixels with difference values much greater or less than their neighbors) with neighboring pixels. Noise removal reduces the search space required for wrinkle information extraction; less noisy difference arrays produce fewer evaluation points from the change in slope process.

Figures 7.12 and 7.13 illustrate an averaged buck image and wrinkled garment on buck image, respectively. Figure 7.14 illustrates evaluation points along the difference array. In comparison to figure 7.9 of the same difference array non-averaged, the averaged array is similar, but has less noise.

```
CORRELATION BETWEEN IMAGES 1 AND 2
     shift =   -8

CORRELATION BETWEEN IMAGES 2 AND 3
     shift =   -1

CORRELATION BETWEEN IMAGES 3 AND 4
     shift =   9

CORRELATION BETWEEN IMAGES 4 AND 5
     shift =   -3

CORRELATION BETWEEN IMAGES 5 AND 6
     shift =   -2
```

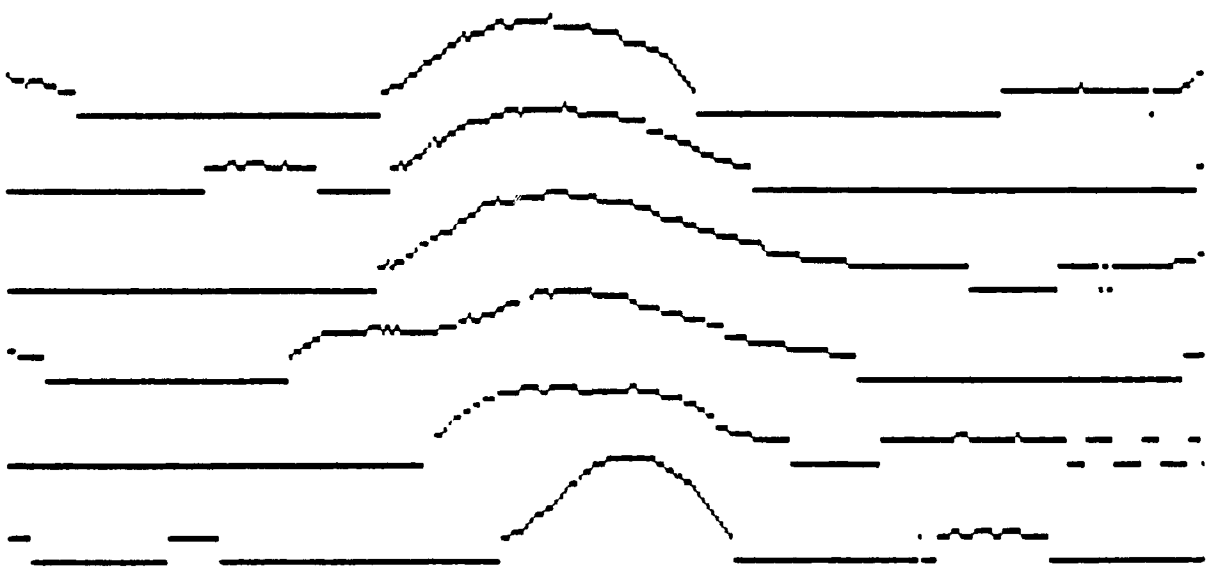


Figure 7.11: Correlation values between six images

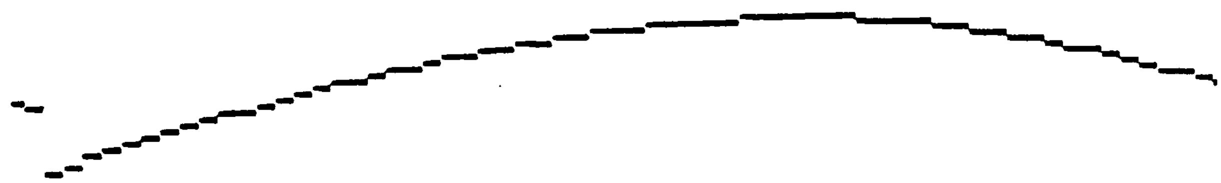


Figure 7.12: Averaged buck contour with 64 points

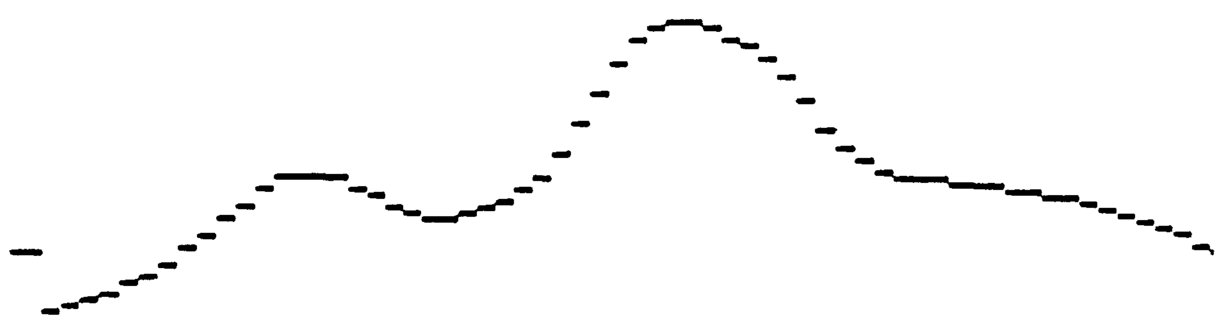


Figure 7.13: Averaged wrinkled garment contour with 64 points

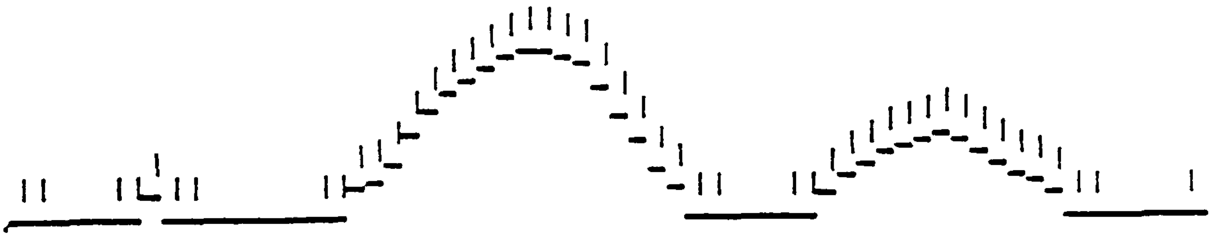


Figure 7.14: Averaged difference image with evaluation points

The final wrinkle detection system will require a field of view of twelve inches or more to handle the widest possible trouser leg width. Averaging algorithms can be modified based on the final system's camera CCD size and field of view to produce a similar resolution to that produced above.

7.2.9 Calibration

Camera calibration is required to determine the absolute wrinkle size information from the images. The calibration procedure relates the actual height of an object to the observed pixel height in the image to produce a conversion value.

For the camera and light source configuration pictured in figure 4.4, the calibration is a simple operation. Placing an object of known height (preferably a block with a flat upper surface) beneath the lightline will produce distortions in the light. The image will contain a bright line with a height change at the location of the block.

The difference image of the buck surface and the image of the light across the block will contain the height information in pixels of the block from the camera viewpoint. The conversion factor to obtain the actual height information from an image is:

$$conversion\ factor = actual\ block\ height/image\ block\ height\ in\ pixels \quad (7.1)$$

The block used for calibration was .5 inches high. Evaluating the difference image array produced an average block height of 27.9 pixels. The conversion factor is then .018 inches/pixel for vertical measurements.

7.2.10 Remarks

The wrinkle detection algorithms above produced highly accurate results for trousers with wrinkles placed by hand. Due to the difficulty in measuring and describing the actual wrinkles in the garment, the best indicators of the accuracy are the image resolution and the accuracy of the modeling algorithms (see Chapter 8). Testing of the grasping devices and wrinkle removal devices will aid in determining the accuracy required for wrinkle detection in the final system. In addition, the testing of the mechanical devices will demonstrate what type of wrinkles will tend to be placed in the trousers by the handling devices.

CHAPTER 8
ALGORITHMS FOR MODELING

The modeling algorithms were developed as part of the wrinkle sensing algorithms to ensure the reliability of the wrinkle sensing algorithms. The modeling approximates the curvature in an image and evaluates the approximating cubic B-splines using interpolation. The modeled curvature that is obtained is nearly identical to the curvature in the original image.

8.1 Wrinkle Modeling with Cubic B-Splines

The modeling algorithms utilize several of the modules developed for wrinkle detection. The image with the curvature to be modeled is filtered to remove noise. The curvature is then approximated with cubic B-splines. The B-splines are cubic functions which, when evaluated at a point, produce the appropriate curve for that point. A function is obtained for every four neighboring points to ensure that only accurate local information is utilized in determining the appropriate function. Through interpolation, the B-splines are evaluated to produce an approximation, or model, of the original curvature.

8.2 Remarks

The approximations are very accurate models of the original curvatures, thus verifying the validity of the B-spline algorithm that was written. The successful modeling, utilizing the same B-spline algorithm utilized in wrinkle detection, suggests that the B-splines are accurate approximations, and thereby, so are the evaluation points obtained from the B-splines.

The modeling has been tested utilizing a series of images of the empty buck and images with a garment on the buck. The following figures are of images created using

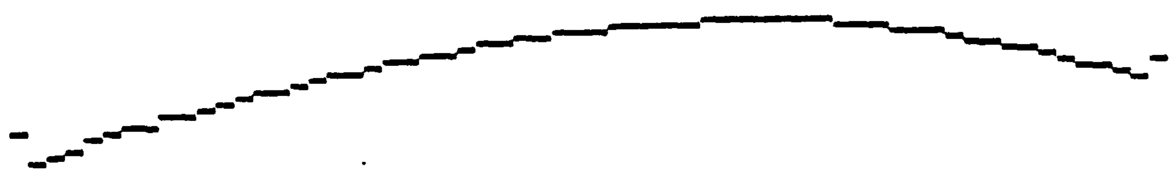


Figure 8.1: Interpolation of buck contour

the modeling algorithms. Figure 8.1 is an 'nterpolation of the press buck surface, 8.2 an interpolation of the wrinkled garment contour and 8.3 an interpolation of the difference contour. These interpolations are extremely accurate representations of the contours in figures 7.3, 7.5 and 7.6.

A model of the press buck surface can be obtained by acquiring images along the length of the buck surface and approximating the curvatures utilizing the modeling algorithm.

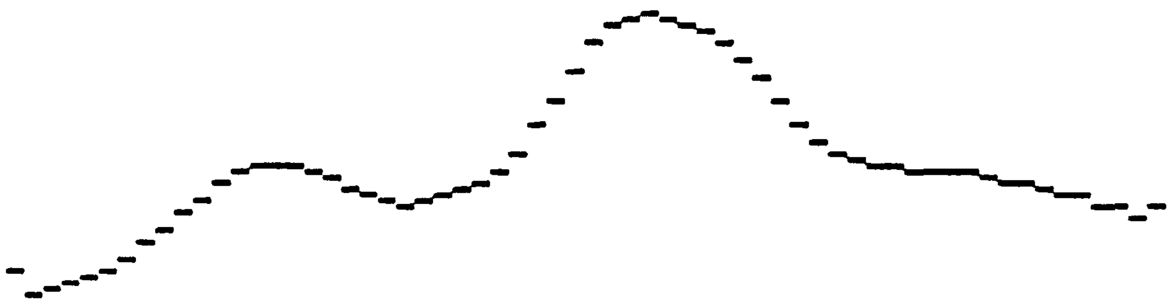


Figure 8.2: Interpolation of wrinkled garment contour

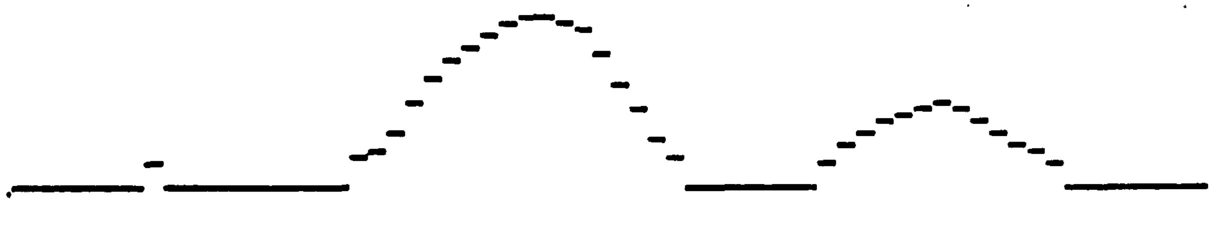


Figure 8.3: Interpolation of difference image

CHAPTER 9
SUMMARY

9.1 Sensing Solutions

The Sensing for Automated Garment Handling project was created to produce sensing solutions for The Automated Handling of Garments for Pressing project. The algorithms developed provide solutions for the seam detection and wrinkle detection requirements.

The seam detection algorithms utilize a Hough transform technique to produce extremely accurate seam location results. The algorithms are efficient and provide a real-time solution for integration with the stepper motor seam-alignment device.

The wrinkle detection algorithms utilize structured lighting and B-spline approximations to provide accurate wrinkle descriptions. The efficiency of the algorithms has yet to be determined, as it will depend heavily upon the accuracy required by the wrinkle removing device and the method by which the algorithms will be integrated into the rest of the system.

9.2 Additional Research

Several additional sensing tasks will be required prior to completion of The Automated Handling of Garment for Pressing project.

A force sensor needs to be designed to ensure that the proper force is applied to the garment to reduce the risk of damaging the garment and to ensure that enough tension is applied to minimize the number of wrinkles. Force sensing was not developed on this project. but should be developed once the grasping/manipulation devices have been fully designed and implemented and the force sensing requirements fully defined.

Another area of research deserving more attention is that of lighting for wrinkle detection. Several methods were pursued for creating a structured light pattern for wrinkle detection. The current method utilizes a single light stripe scanned across the garment and requires multiple images for processing. Computer generated holograms might provide the resolution required and pattern complexity desired to alleviate the need for scanning a light source across the garment and obtaining multiple images.

Sensing integration software has been partially developed, but will require further work when the other system components are ready for integration. In addition, the required image resolution should be determined in conjunction with testing of the accuracy of the garment handling devices. The image resolution will drastically effect the processing time required for seam detection and wrinkle detection.

9.3 Alternate Applications

The solutions developed for seam detection and wrinkle detection are by no means limited to garment applications. Many applications involve the handling of flexible materials and may benefit from applying the algorithms presented in this paper.

APPENDIX A
GLOSSARY OF GARMENT TERMS

buck - the curved vacuum surface of the pressing unit on which the trousers are placed.

busted seam - a seam whose selvage has been pressed flat against the trouser leg.

crease - a line produced in a fabric by pressing the fabric. An intentional crease is commonly placed in the center along the length of each trouser leg.

ply - a layer of material, as illustrated in figure A.1.

press - a machine that presses or smooths fabrics or garments.

pucker - a small wrinkle produced during the sewing process by applying too much, too little or uneven tension in the stitching.

seam - the edge formed by sewing two pieces of fabric together. Various types of seams exist. The type addressed in this project is illustrated in figure A.1.

selvage - the excess material formed when creating a seam, as illustrated in figure A.1.

wrinkle - a ridge of excess material formed in the fabric.

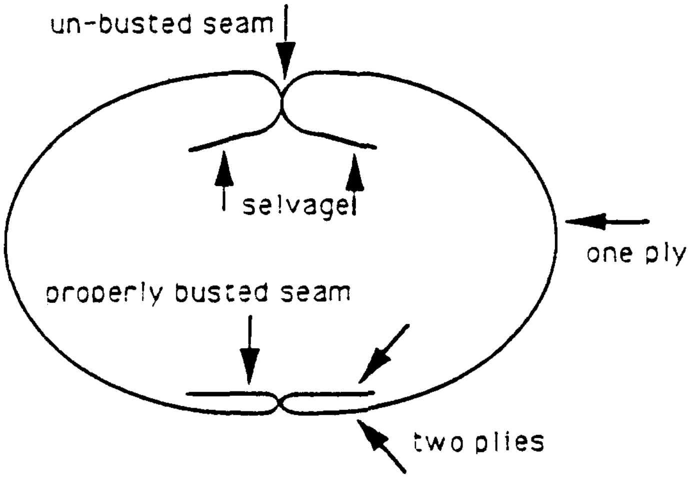


Figure A.1: View of Cuff Opening on Trouser Leg

LITERATURE CITED

[1] Gonzalez, Rafael C. and Wintz, Paul. *Digital Image Processing.* Massachusetts: Addison-Wesley Publishing Company, 1988.

[2] Schalkoff,R. *Digital Image Processing and Computer Vision.* New York: John Wiley and Sons, Inc., 1989.

[3] Ballard, Dana H. and Brown, Christopher M. *Computer Vision.* Englewood Cliffs, New Jersey: Prentice-Hall, 1982.

[4] Press, William H. et al. *Numerical Recipes in C.* Cambridge: Cambridge University Press, 1990.

[5] Kernighan, Brian W. and Ritchie, Dennis M. *The C Programming Language.* Englewood Cliffs, New Jersey: Prentice Hall, 1988.

[6] Schildt, Herbert. *C, The Complete Reference.* Berkeley: Osborne McGraw-Hill, 1987.

[7] Farin, Gerald. *Curves and Surfaces for Computer Aided Geometric Design.* Boston: Academic Press, Inc., 1988.

APPENDIX B

SEAM DETECTION AND INTEGRATION MODULES

The following modules were created for seam detection and integration and must be linked and compiled to create the output file for running the algorithms. The algorithm input is a 256 by 242 image file with 256 grey levels per pixel. The input is an image of the vertical seam with a shadow cast across the seam. The shadow is created by external lighting at an angle to the trouser leg, with the camera normal to the trouser leg. The equipment configuration is illustrated in FIGURE of 4.1.6.

SEAM DETECTION

rdimg.c - A module which queries the user for the input image filename and reads the image into a $256x242 = 61952byte$ buffer. If the input image is not accessible the module replies with "Can't open file". INPUT: 61942 image buffer. OUTPUT: modified 61952 byte image buffer.

rdhd.c - A module which reads image header information into a buffer whose size is dependent upon the image header size. The image header size varies from system to system. INPUT: image header buffer. OUTPUT: modified image header buffer.

wrimg.c - A module which writes the input buffer to a SUN compatible 256 by 256 output file for viewing. The user is queried for the output file name. INPUT: 61952 byte image buffer, image header buffer. OUTPUT: SUN^{TM} x-window compatible image file.

wrline.c - A module which writes the line created by the hough transform slope-intercept pair to an 256 by 256 image file. The line is created as a bright contour against a dark background. The user is queried for the filename of the image file created. INPUT: image header buffer, slope, intercept. OUTPUT: SUN^{TM} x-window compatible image file.

3conv_vert.c - A module which computes the convolution of a image buffer of size 61952 byte. utilizing the convolution kernel displayed in FIG of 6.1.1. The outer row and edge of the image which are not convolved are filled with pixel intensities equal to the nearest convolved neighbors' intensity. INPUT: original 61952 byte image buffer, buffer for convolved image. OUTPUT: modified convolved 61952 byte image buffer.

histo_thr.c - A module which determines the histogram of an image buffer of size 61952 byte. The module also determines the appropriate threshold value by summing the number of pixels per grey level, starting at the upper end of the histogram. The grey level at which the pixel sum is 200 indicates the proper threshold. The image is then thresholded and stored in the input image buffer. INPUT: convolved image buffer. OUTPUT: thresholded image buffer.

hough_vert.c - A module which determines the best slope-intercept pair to describe the seam contour in the thresholded image. Using the equation: $x = \frac{1}{m}y + a$, a determines the image top row intercept and m determines the slope. INPUT: thresholded image buffer. OUTPUT: slope, x-intercept.

seam.c - A module which incorporates all of the above seam detection modules.

SEAM INTEGRATION

calibrate.c - A module which calibrates the camera location based on the input image. The module determines the resolution and the location of the reflective strip calibration point. IN: 61952 byte image buffer, iteration number. OUTPUT: image resolution, reflective strip mid-point location.

integr_seam.c - A module which incorporates the calibrate.c module and all the seam detection modules, as well as a function to determine the proper stepper motor driving distance to align the seams. The seam detection modules are passed an additional offset value which is determined from the number of rows in the top part of the image which are used for the calibration device. The seam extraction and approximation modules process image data below the top rows utilized for calibration. The modules are each called twice, once for the in-seam image and once for the out-seam image.

APPENDIX C
WRINKLE DETECTION MODULES

The following modules were created for wrinkle detection and must be linked and compiled to create the output file for running the detection algorithm.

`readimg180.c` - a module which queries the user for the input image and reads the image into a 61952 byte image buffer. The camera for wrinkle detection is positioned upside down such that the images obtained are rotated 180 degrees. The module stores the image file into the image buffer in a reversed order, so that the values will appear as if the camera was correctly positioned. If the input image is not accessible the module replied with "Can't open file". INPUT: 61952 byte image buffer. OUTPUT: modified 61952 byte image buffer.

`smooth.c` - a module which performs a smoothing operation on the input image buffer by averaging every three adjacent vertical pixels in each of the columns. The smoothing in each column starts at $location = ref + offset$, where ref is the reference or dummy values and $offset$ is a constant column offset value. ref is set to the bottom of the column for smoothing the reference image, and to the reference image 256 column y-location values for the garment image. The garment image y-location values are guaranteed to be lower (closer to the top of the image) than the reference values, so there is no need to smooth the entire image. INPUT: 61952 byte image buffer, column location to start smoothing from, offset from . OUTPUT: smoothed image buffer.

`findbright.c` - a module which searches each vertical column for the brightest pixel. Each column's brightest pixel's y location is stored in a 256 by 1 array. The search in each column starts at $location = ref2 + offset2$, where $ref2$ is the reference or dummy values and $offset2$ is a constant column offset value. $ref2$ is set to the bottom of the column for smoothing the reference image, and to the reference image 256 column y-location values for the garment image. The garment image y-location values are guaranteed to be lower (closer to the top of the image) than the reference values, so there is no need to search the entire image for the bright location. INPUT: image buffer, 256 by 1 reference/dummy array, offset value. OUTPUT: 256 by 1 array of column brightest pixel y-location values.

`differ_hipass.c` - a module which determines the difference values and the flat value of the two 256 by 1 input arrays. The input arrays are the brightest pixel locations array for the reference image and the array for the corresponding garment image. The difference is a 256 by 1 array. The flat value is the minimum difference value. INPUT: 256 by 1 reference image array brightest pixel locations, 256 by 1 garment image array brightest pixel locations. OUTPUT: 256 by 1 difference array, flat value.

`spline.c` - a module which determines the cubic B-spline approximation to the input array. The module calculates the first and second derivatives of the approximation and determines points at which the first derivatives (slopes) are different. These points are flagged as points of change. INPUT: image buffer. OUTPUT: first derivatives, array of points of change.

`wrmultcur.c` - a module which writes multiple input array curves to a single output file. For easier viewing, each curve is given a vertical offset based on the iteration number of the particular curve. INPUT: multiple 256 by 1 input array curves. OUTPUT: SUN^{TM} x-window compatible image file.

`locate.c` - a module which locates the wrinkles and determines their height and width. This is performed by scanning through only those points of the 256 by 1 difference array which have been flagged by the spline.c module as points of change. By tracking changes in the heignt. the height, width and locations of the wrinkles are determined. The user is queried for the output filename to which to write the wrinkle information. INPUT: 256 by 1 difference array, flat value, inflection points. OUTPUT: wrinkle location, height and width information written to file.

`correl.c` - a module which determines the correlation between two input images represented by their 256 by 1 difference arrays. Image one is shifted across image two to produce correlation values for each of the 512 shifted versions. The maximum correlation value determines the amount of shift. The user is queried for the output filename to which to write the correlation information. INPUT: 256 by1 difference image array one, 256 by 1 difference image array two. OUTPUT: correlation value.

`findwrin.c` - a module which incorporates all of the above wrinkle detection modules. Each image is fully processed to extract wrinkle data. Following the processing of the first two images, correl.c is called for every two images.

APPENDIX D
EQUIPMENT

Several systems were utilized in the development of the sensing algorithms. These included: the GMF Robotics and Vision system, the Host PC System and a SUN^{TM} System.

1. GMF

The GMF Robot and Vision System was used extensively for original development of the algorithms. Algorithms were created using the Karel^{TM} language with the Insight language extension for processing images. The Insight vision system provided basic image acquisition and processing operations. The images were 256 by 242 pixels in size with $2^8 = 256$ grey levels with a 24 byte image header.

2. PC

The Host PC system was used for some algorithm development, communication with the GMF and SUN^{TM} systems and acquisition of images. Algorithm development was done in C using the UNIX operating system. The images obtained using the MV-Series$_{TM}$ image buffer were of size 512 by 480 pixels with 2^8 grey level intensities and no image header.

3. SUN^{TM} System

Due to the difficulty in maintaining the PC and the frequent unavailability of the unit, a SUN^{TMTM} system was utilized for the majority of the algorithm development. The algorithms developed on the SUN^{TM} system were developed in C using the UNIX operating and X windows for viewing images.

The SUN^{TM} images were of size 256 by 256 and contained 2^8 grey level intensities with a 32 byte size image header.

APPENDIX E
IMAGE CONVERSION ALGORITHMS

Due to the variety of image sizes and image header types produced by the three systems utilized for development, several routines were written to allow conversion from one image type and size to another image type and size. The following conversions are supported: MVSeries^{TM} to GMF, MVSeries^{TM} to SUN^{TM}, SUN^{TM} to GMF and GMF to SUN^{TM}. The SUN^{TM} to MVSeries^{TM} and GMF to MVSeries^{TM} conversion is not supported; this project does not require images of as high resolution as the 512 by 480 pixel images obtained from the MVSeries^{TM} image buffer.

1. MVSeries^{TM} To GMF,SUN^{TM}

To convert the 512 by 480 pixel MVSeries^{TM} image to the smaller images, algorithms were written to average adjacent pixels and to select alternate pixels on alternate lines. The algorithms produced almost identical reduced images. Thus, the routine to select every other pixel was chosen to maintain as low a processing time as possible.

The MV-Series^{TM} image files are stored in a different format than the GMF and SUNquandrants and each quadrant is stored by row. Also, the grey levels are not in the standard format and thus are converted to be compatible with the GMF and SUN

2. GMF To SUN^{TM}, GMF To SUN^{TM}

The SUN^{TM} system handles images of varying size. Thus the conversion from GMF images merely requires a change in the image header for display purposes on the SUN^{TM} system.

In converting from SUN^{TM} to GMF images, the image header must be modified and the image size changed if it is not 256 by 240 pixels. 256 by 256 pixel SUN^{TM}

images were often utilized which required subtracting the last 14 rows of the image in conversion to the GMF format.

MODULES

`headergmf.rd` - an image header file which is read by rdhdgmf.c.

`headersun.rd` - an image header file which is read by rdhdsun.c.

`mvtosun.c` - a module which converts the MVSeries^{TM} compatible input image file into a SUN^{TM}-compatible output image buffer.

`mvtogmf.c` - a module which converts the MVSeries^{TM} compatible input image file into a GMF size-compatible output image buffer.

`gmftosun.c` - a module which converts the GMF compatible input image file into a SUN^{TM} size-compatible output image buffer.

`suntogmf.c` - a module which converts the SUN^{TM} compatible input image file into a GMF size-compatible output image buffer.

`wrimggmf.c` - a module which writes the GMF size-compatible image to buffer to an output file.

`wrimgsun.c` - a module which writes the SUN^{TM} size-compatible image buffer to an output file.

`rdhdgmf.c` - a module which reads in the image header file rdhdgmf.c for use in writing the converted GMF compatible image to a file.

`rdhdsun.c` - a module which reads in the image header file rdhdsun.c for use in writing the converted SUN^{TM} compatible image to a file.

`convert.c` - a module which incorporates the above modules to convert from one image format to another. This module calls one of each of the following modules: rdhdsys2.c, sys1tosys2.c and wrimgsys2.c.

GRASPING AND MANIPULATION DEVICES

FOR FLEXIBLE MATERIAL HANDLING

by

John P. Peiffer

A Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF ENGINEERING

Approved:

Stephen J. Derby

Dr. Stephen J. Derby
Thesis Advisor

Rensselaer Polytechnic Institute
Troy, New York

December 1990

This Page Shall Not be Duplicated

TABLE OF CONTENTS

CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(continued)

ACKNOWLEGEMENTS

The author wishes to express his gratitude to his wife, Rebekah, for her unwavering support, encouragement, and patience during her years as a "students widow".

My appreciation must also be expressed to Dr. Stephen Derby. His enlightening explanations and insight into robotics and grasping and manipulation concepts have been very instructive and have formed the basis for this thesis.

Larry Ruff must also be acknowledged for his extremely practical advice in the areas of mechanical design and process design. His suggestions have had a strong influence on the conceptual designs and details of the devices discussed in this thesis.

Also, I could not miss this opportunity to thank Claudio Caballero and Cliff Lansil for their suggestions and general help with the "hands-on" part of my work.

I must also acknowledge all the other members of the research team who have been helpful in some way.

And for those of you who find this to be a pressing issue: It only seams sew.

FOREWORD

This research project is a three year effort being funded by the Defense Logistics Agency with the goal of making American apparel manufacturers more competitive in the global marketplace.

The project is currently at the beginning of its second year, having its start in January 1990. As originally proposed the first year was spent in fundamental conceptualization, research, and experimentation. The second year will be the integration phase, where the results of each area of research will be integrated into a working prototype system. The third phase of the project will be coordinated with apparel manufacturers to field test the prototype system. The project will then be transferred to a manufacturer to be commercialized.

Twelve researchers from the the Center for Manufacturing Productivity of the Rensselaer Polytechnic Institute are doing basic research in the areas of: grasping and manipulation, sensing, modelling, and integration. Two professors from the Fashion Institute of Technology are providing technical support in the areas of textile engineering and garment manufacturing.

ABSTRACT

The final pressing operation for tailored garments is a labor intensive task in an environment unsafe for humans. Also, the quality and consistency of final pressing is critical to the customer's judgement of the garment's value. A project that addresses these problems by attempting to automate the entire pressing operation is being conducted.

An overview of the development of automation in the apparel industry and the state of the art is presented to show the need for and justification of automation in pressing. The automated handling of finished garments has largely been avoided in the past because of the properties of fabric which make it incompatible with existing grasping and manipulation techniques.

The design of four prototype garment handling devices is presented. Experiments were performed to determine the effectiveness and reliability of each of the devices. The experimental results are used to indicate a direction for the further development of the prototypes.

CHAPTER 1

INTRODUCTION

The Center for Manufacturing Productivity and Technology Transfer of the Rensselaer Polytechnic Institute embarked on an automated garment handling project in January 1990. The purpose of the project is to address the need of automation in the garment and apparel industries, and specifically, the final steam pressing operation for high quality pants.

Although many other manufacturing areas have seen the successful introduction and use of robotics and automation, the garment industry has been largely avoided because of the inherent qualities of fabric which make it incompatible with conventional automated handling techniques.

There are several reasons why the pressing operation was chosen to be automated. First, there has been little work done to reduce the labor intensity of the loading and unloading portion of the pressing cycle, even though it occupies a significant portion of the pressing cycle time.

Secondly, the pressing operation uses high temperature steam and a large clamping force that creates an unsafe environment for human operation. Many steps have been taken to reduce human risk. However, the most effective safety measure is to remove the human from direct interaction with the press.

Thirdly, the final pressing of a garment has a great influence on the customer's perception of its quality. An automated pressing system has the potential to greatly increase the quality and consistency of the finished garment by eliminating the natural variations of human performance during a highly repetitive job, such as the pressing operation.

The synergistic effect of enhancing the quality of the finished garment, while reducing the direct and indirect manufacturing costs of the pressing operation, provides an opportunity for American apparel companies to increase their international competitiveness.

A project team was formed consisting of four technical groups: grasping and manipulation, sensing, integration, and modelling.

The focus of the grasping and manipulation group is to identify the necessary operations to successfully prepare the pants for loading and the sequence of manipulation required to load the pants into the press. Finally, the mechanical devices and grippers needed to achieve these functions are being designed and prototyped.

The sensing group is developing the sensors and algorithms needed to provide the information to successfully manipulate the pants throughout the pressing operation and to identify the existence of error conditions.

The modelling group is developing a mathematical model to simulate the dynamic interaction of fabric with its environment. An interactive module of this program will convert sensory information into useful manipulation parameters.

The integration of the mechanisms and sensors into a working system, and the development of software needed to control the hardware are being done by the integration group.

CHAPTER 2

HISTORICAL BACKGROUND

There are three steps in the automation process that have been identified by Carroli.[1] The first step is the automation of individual tasks or sub-processes. This can involve several types of automation such as mechanization and automatic control. In mechanization the human labor content of the task is reduced or eliminated and only the parameters of the sub-process remain in human control. In automatic control, sensors provide feedback to a control system that adjusts the parameters of the sub-process as needed, thus, the need for human intelligence to successfully operate the sub-process is reduced. This step has already been accomplished in many manufacturing areas. For example, the spinning operation in textile production is automated and requires minimal human interaction.

The second step in the automation process is systems automation. In this case each automated sub-process is integrated into a system. If this step were accomplished on a large scale, an entire textile plant could run automatically. However, it is usually practical to integrate sub-processes on a smaller scale. For example, the combing and spinning processes could be integrated to work as a system.

The third step in the automation process is control systems automation. Here the control parameters of the system are not restricted to preprogrammed settings, they are adjusted as new process information becomes available. In other words, the control system "learns" from its past experience and adapts to changes in the process. This is one promising application for artificial intelligence.

It is useful, at this point, to briefly review the development of automation in the textile industry. The accomplishments of engineering in textile manufacturing are numerous and well known. Some of the earliest mechanical inventions about the time of the Bronze Age were weaving looms built to increase the efficiency of the weaving process. As the art progressed, whorls and spindles were invented to assist with the forming of yarn. The state of the art changed little until the advent of the Industrial Revolution. Until then the human operator was the prime mover of the machinery. This was the major factor which limited the production of textiles to subsistence household operations. When large centralized sources of power became available, textile factories came into existence and household production began to decline. Eventually, looms developed to the point where a highly-patterned tapestry could be made entirely by machine, with only a human supervisor. These looms followed the patterns on pre-punched cards. High speed knitting machines were developed in the 1940's that increased the production of one operator by several orders of magnitude. In the last forty years, textile processes such as filament production, spinning, combing, weaving, and materials handling have been developed to the point where human labor is not only being automated, but human intelligence is being replaced by sensors, feedback, and complex self-correcting control systems.[2] Although much has been accomplished to automate and control individual steps, there remains a great deal of work to be done in integrating every step in the overall process into an automated system (process automation).[1]

The focus of automation in the apparel industry has traditionally been on joining processes, specifically sewing. The forerunner of the modern sewing machine was patented in 1846 by Elias Howe. The labor intensity was reduced dramatically but a skilled operator was still required. Templates and other devices have been added to sewing machines to further de-skill and speed up their operation. More recently, programmable sewing and embroidery machines have been introduced. The operator must only load and unload the machine, a program automatically moves the fabric or the sewing head to guide the needle in the correct path. These machines detect broken thread and automatically re-thread the needle or they can change thread color at any time.

Several automated sewing systems have been developed by Draper Laboratories. These systems accept cut fabric and manipulate the pieces to various stations for sewing, inspection, or unloading. These systems are currently in a prototype stage and are being developed for commercialization. An automated garment manufacturing system is being developed that uses an ultrasonic joining method and a robotic manipulator. The goal is to have a fully automated system requiring minimal human interaction through the entire process, from the bolt of fabric to the finished garment.[3]

Recent advances have been made in fabric layout and cutting. Computer-aided design (CAD) systems have been integrated with automatic fabric cutters. The CAD system increases the efficiency of the design process, generates the numerical control data needed to control the cutter, and transmits it to the cutter. The use of CAD to position the individual pieces in the optimum layout can reduce by half the amount of fabric wastage compared to an unaided human.[4]

The final operation on a finished garment is steam pressing. Since it is the last operation before inspection and shipping, it is critical that sewing and assembly errors are minimized or hidden by the pressing operation. Presses have been automated to reduce operator skill and speed up pressing. Since the development of microprocessor technology, the control of the press parameters has been optimized and stored as a program. When a change of garment material occurs, the operator only has to load a new program. Some pressing operations require little operator skill. For example, fully automated steam tunnels exist that only require loading and unloading. But, this type of operation does not make a crease in the garment and therefore wouldn't be sufficient for tailored clothing. Clamshell presses develop a large clamping force and are used to make a crease in garments. However, to ensure that the crease is located properly, a skilled operator is needed to properly load the garment into the press.[5]

It should be noted that most of the apparel industry is only at stage one in the automation process. Many of the labor intensive sub-processes have been mechanized and a few have been automated, like programmable sewing, automatic cutting, and steam tunnel finishing. Other processes like crease pressing and finished garment handling have only been partially automated. It is important to realize that in order for the apparel industry to reach the second stage of automation, process automation, all of the individual sub-processes must be automated.[1]

The Rensselaer research effort is focusing on automating the pressing operation. There are several reasons why this process was chosen. First, the pressing operation creates an environment that is unsafe for humans. A clamshell press uses steam in excess of 300°F and a clamping force over 1000 lbs. Many safety measures have been taken to reduce the risk of operator injury. Modern presses have emergency release buttons and use two-handed cycle start buttons or sensors to ensure that the operators hands are clear of the press buck before the cycle begins. These efforts have undoubtedly reduced the number of injuries caused by pressing equipment. However, the most effective safety measure is to remove the human entirely from direct interaction with the press. Even if operator safety was perfected, the high temperature and humidity in the immediate vicinity of the press can cause an untenable climate, even in climate controlled factories.

Secondly, the rapid rise in apparel imports during the 1980's is forcing American apparel manufacturers to find new ways to reduce their costs while maintaining or increasing their level of quality. Imports now hold a 40% share of the U.S. apparel market.[6] Direct labor accounts for 30% to 40% of the final cost of a finished garment.

Automation is suggested as a solution to this problem by increasing productivity and quality, while reducing the direct cost of labor, and eliminating the indirect costs of medical insurance and injury leave. The large-scale application of technological change will cause social consequences that should be dealt with.[7]

Third, the quality of the final pressing operation is critical to the customer's judgement of the garment's value. By automating the handling of the garment and inspection after pressing, a large number of errors can be elimi .ted that normally occur during a highly repetitive operation.[8]

A review of the state of the art in the apparel industry revealed that little work is being done with the automated handling of finished garments, both during and after the pressing operation.[4,9]

Finally, the number of skilled workers in the garment industry has been declining and is expected to continue declining in the foreseeable future.[10] Therefore, it is appropriate to find new ways to de-skill processes and increase their productivity in order for apparel companies to maintain the same level of output with less skilled workers.

CHAPTER 3

THE RESEARCH EFFORT

3.1 Introduction

The research team at Rensselaer is composed of 12 individuals separated into 5 groups. An organizational chart of the research team is shown in Figure 1.

This chapter describes the efforts of the technical research groups (modelling, sensing, and system integration) and their relationships to the grasping and manipulation effort.

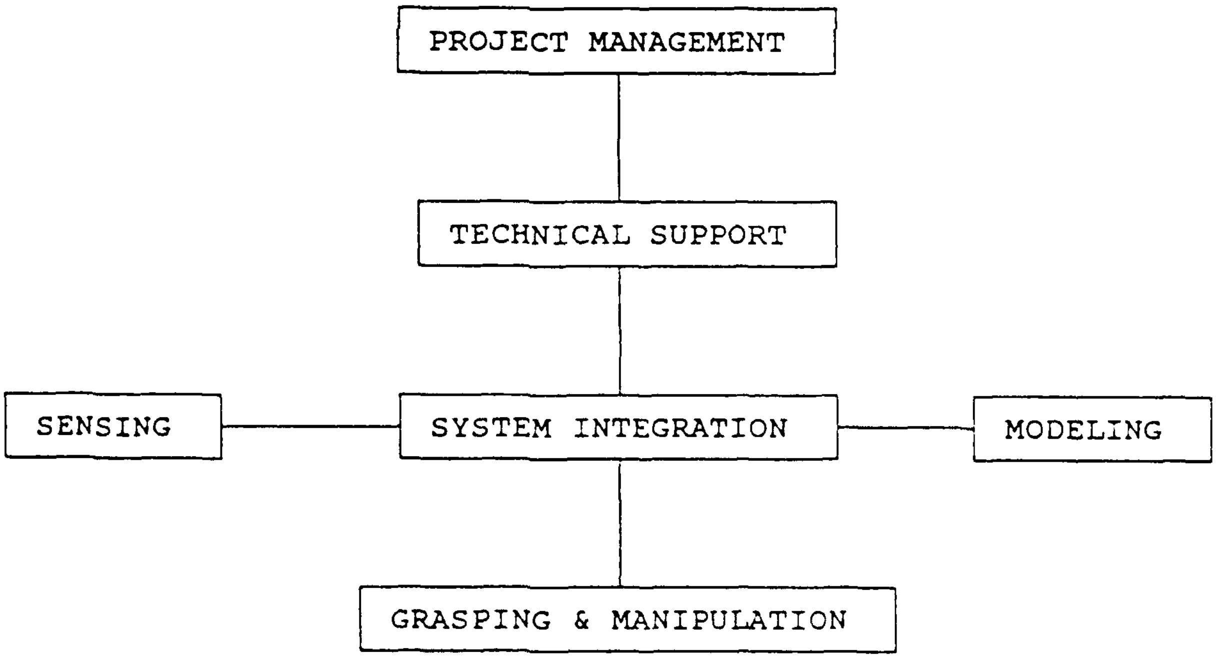


Figure 1: Rensselaer Research Team Organizational Chart

3.2 The Modelling Effort and its Impact on Grasping and Manipulation

The focus of the modelling group is to create a dynamic mathematical model of fabric. A particle-based energy minimization approach was chosen because of the anisotropy, the non-homogeneity, and the lack of rigidity of fabric.

Various mathematical modelling techniques have been used in the recent past to deal with fabric characterization. For example, the finite element method has been used to predict buckling behavior in woven fabrics [11] and an energy minimization method has been used to predict a fabric's physical and mechanical properties.[12] One similar research effort used a particle-based energy minimization approach to model the appearance of cloth, however, their initial assumptions are different than the Rensselaer work.[13]

There are two macroscopic empirical mechanical property measurement techniques currently used in the garment industry. One is the Kawabata Evaluation System (KES) and the other is the Fabric Assurance by Simple Testing (FAST) method. Since these methods are macroscopic in nature they can only measure the bulk mechanical properties of a fabric, such as the extensibility, hygral expansion, compressional resilience, or the bending rigidity. The KES method also uses the mechanical property measurements to predict subjective qualities like the fabric's hand.[14] It is important to realize that these methods only describe a fabric's static qualities; they do not give the user a description of the motion of the fabric under the influence of external forces.

The efforts of the Rensselaer modelling group, which are focused on dynamic modelling using a particle-based method, are directed toward solving a unique modelling problem that has never been approached in the same manner.

The Rensselaer group is currently executing dynamic simulations of fabric in free-fall and draping using a 50x50 resolution grid (see Figure 2). There is a good correlation between their results and the empirical results of draping experiments.

As the modelling effort continues to develop, it will provide a database of information useful to the grasping and manipulation work. One objective of their work is to make an interactive module of the modelling program. This would allow sensory input information to be translated into useful manipulation parameters. For example, if the orientation and configuration of wrinkles in a pant leg on the press buck was provided to the model, it would be able to return the location of the optimum manipulation points and the magnitude of the force needed to remove the wrinkles. A specialized force controlled end-effector and control routine would then be able to use this information to remove the wrinkles.

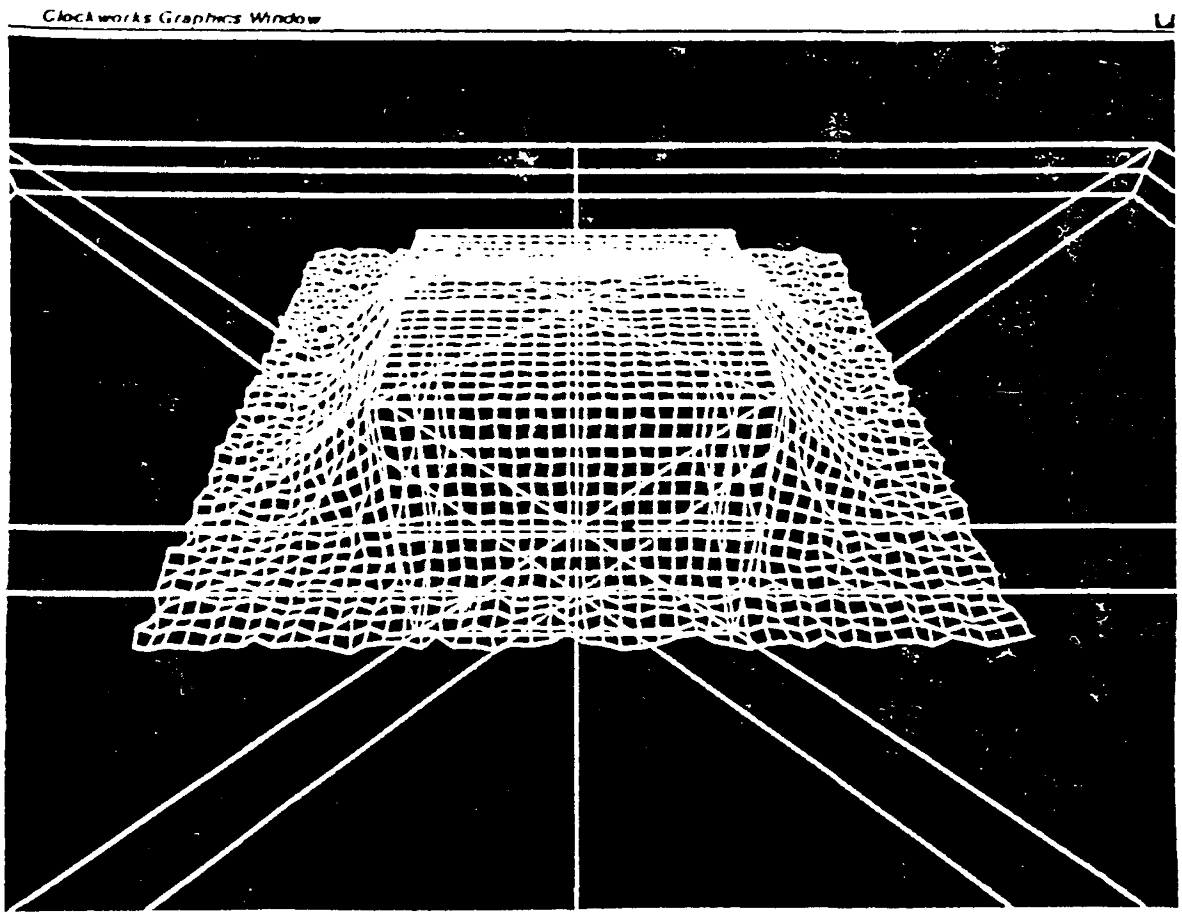


Figure 2: Modelled Simulation of Draping Fabric

3.3 The Sensing Effort and Its Impact on Grasping and Manipulation

The work of the sensing group has been directed toward two areas. First, sensing the relative location of the inseam and outseam and, second, locating and characterizing wrinkles in a pant leg on the press buck. Machine vision is used to gather information in both cases (see Figures 3 and 4).

Future efforts will use additional sensors to provide information on the location of the pants' cuff, crotch, and excess material. Also, sensors will be used to identify possible external error conditions, such as pants improperly presented, or loss of air pressure to the pneumatic components. Sensors will also identify internal error conditions like a mechanism not operating as expected or a faulty pressing cycle. The sensors will work in conjunction with the modelling module to provide the required information.

The grasping and manipulation devices will depend heavily on sensors to recognize error conditions and provide manipulation information. For example, a sensor will be used to locate the pants crotch and allow the transfer gripper to be positioned properly before attempting to grasp the pants leg. Sensors will also determine if the pants are grasp correctly before the transfer process begins.

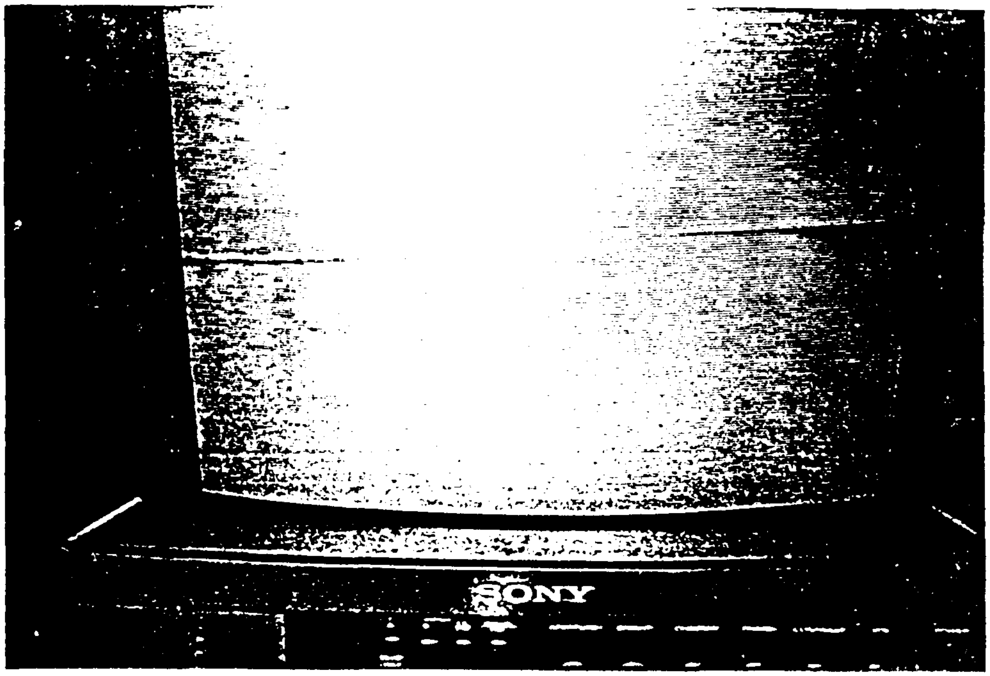


Figure 3: Machine Image of a Seam

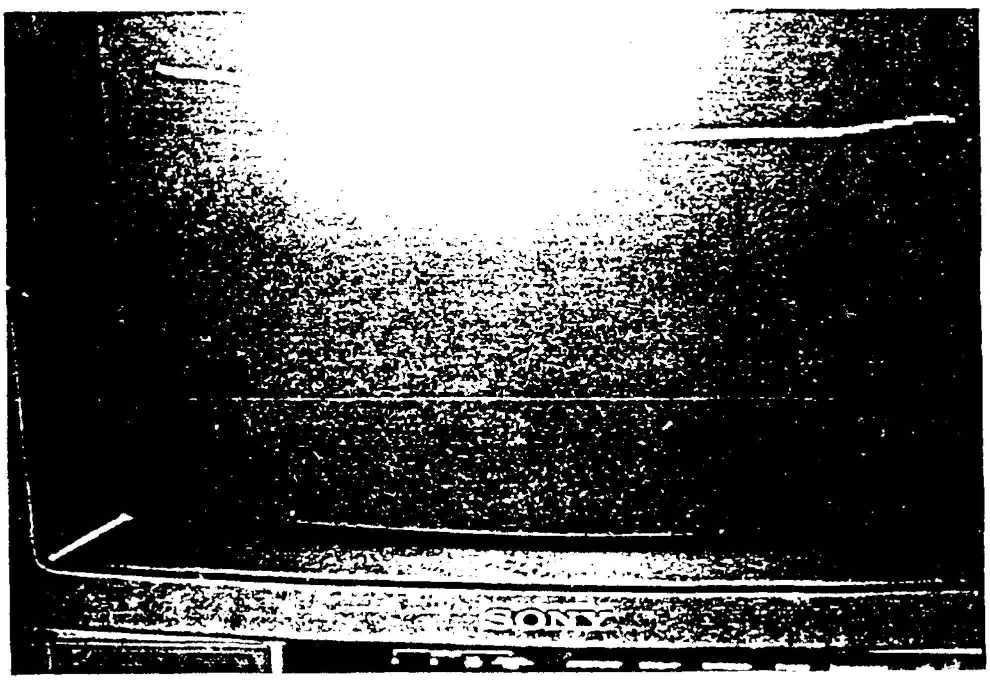


Figure 4: Machine Image showing Light Line Distortion due to the Presence of a Wrinkle

3.4 The Integration Effort and Its Impact on Grasping and Manipulation

The integration effort is divided into two sections. The sensors and vision cameras and the grasping and manipulation mechanisms will be integrated by the hardware integration group. This will ensure that the mechanisms and sensors are compatible and provide the optimum performance.

The software integration group is creating the control protocol, hardware drivers, and the user interface for the system. The overall control of the system is being done using the robot controller in conjunction with a host computer. For example, a robot program can be used to control pneumatic signals to the hardware via the auxiliary input/output ports. Stepper motors and DC servomotors can also be controlled from a robot program by utilizing the communication link between the robot controller and the motor control cards in the host computer. The press is also wired to the robot controller and can be controlled by the host computer. This type of centralized control system is either directly executing each command or is supervising the secondary controllers and monitoring their performance. Safety and robustness are optimized since each step in the process is monitored directly by the host computer through sensory information.

The integration effort is perhaps the most important part of a complex process control system. The sequencing, synchronization, supervision, and error response during every step in each process is critical to the performance and quality of the entire system.

CHAPTER 4

MECHANISM DESIGN

4.1 Design Theory and Analysis

An effective method for the structured design of a complex process or component a is called the functional analysis method.[15] A brainstorming approach was also used by the Rensselaer team to generate design alternatives. In Tables 1 and 2, the results of some of the brainstorming session in given. Some of these ideas were immediately chosen over others, but the important point is that any idea, no matter how ridiculous, should be considered.

The five types of grippers that can be used to grasp and manipulate fabric should be considered. They are:

1. Friction - development of a frictional force between the gripper surface and fabric is required.
2. Puncture - the gripper penetrates the fabric to develop holding forces.
3. Adhesive - chemical bonds create the attractive force.
4. Vacuum - atmospheric pressure is used to create a frictional force between fabric and gripper and develops

buoyancy.

5. Electrostatic - electrical potential creates an attractive force.

The state of the art review conducted by the Rensselaer team revealed that there are few commercially available fabric grippers. And a study conducted by the American Apparel Manufacturers Association showed that, of 47 different limp fabric handling devices, only 3 are currently in wide-spread commercial use.[16] These are:

1. Clupicker - a rotating wheel frictionally pulls the fabric ply in between the wheel and a finger.
2. Walton picker - saw toothed edges partially puncture the fabric and then close and clamp (see Figure 5).
3. Polytex or Littlewood Needle device - puncture the fabric with needles at different angles so the fabric can't slip off.

These three devices are only for picking up a single ply of fabric. In most applications they are used to grasp and manipulate freshly cut parts from a stack. They were not designed handling finished garments. However, an application may exist for them where it may be necessary to separate two adjacent plies of fabric.

TASK	PERFORMANCE CRITERIA	PROCESS OPTIONS	DESIGN ALTERNATIVES
TRANSFER TO PRESS	Adapted well for automated handling, simple path, must not cause wrinkles, must maintain seam alignment	>Transfer pants to horizontal press >Transfer pants to vertical press	>Grab roller alignment devices at each end of leg and remove at press >Multiple-handed gripper >Temporarily tack seam and use a simple gripper
WRINKLE DETECTION AND ELIMINATION	Reliable, accurate, must not disturb fabric	>Non-contact	>Air-jet smoothing >Light-line distortion det. >Wrinkle shadow detection
		>Contact	>Rotary brush >Multi-fingered hand >Tactile probe sensor
		>Use a highly reliable transfer device that presents pants to press with acceptable wrinkles	>Multiple handed gripper with tensioning and some smoothing capabilities

Table 2: Functional Analysis Worksheet (page 2)

TASK	PERFORMANCE CRITERIA	PROCESS OPTIONS	DESIGN ALTERNATIVES
PRESENTATION TO AUTOMATED SYSTEM	Adapted well for automated handling, simplicity, minimize the number of manipulations required to move to seam alignment station	Unit Presentation >waist-end down >waist-end up >horizontal	>Pants initially loaded by operator onto seam alignment device >Pants loaded by operator onto a simple hanger that is compatible with the alignment device
SEAM DETECTION	High repeatability, Accuracy within + or - 1/16 inch	>Selvage Detection & Calculation	>Vision -light transmissibility >Mechanically -2 ply thickness detection
		>Direct Seam Detection	>Vision -ultraviolet reflective thread -light transmission through seam -seam shadow -use pre-drilled holes or notches as markers >Mechanically -seam detection mech. using selvage fold back & triple thickness detection >Other -proximity sensor and metal thread
SEAM ALIGNMENT	High Repeatability, No damage to fabric	>Cuff-end	>Rollers inserted into end of leg
		>Waist-end	>Rollers inserted into waist end of leg >Long rollers thru leg >Slide plies relative each other other externally

Table 1: Functional Analysis Worksheet (page 1)

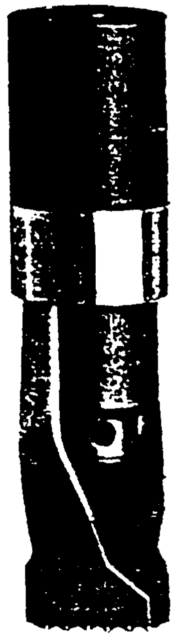


Figure 5: The Walton Picker

The primary manipulation device currently in the workcell is a GMF six-jointed anthropomorphic arm. The flexibility in motion and the re-programmability of the robot make it ideal for conducting this research effort. A similar, but less costly, robot may be used in the final prototype system. Or the necessary manipulations in the final prototype may be performed by dedicated automated devices. In either case, both robotics and automation are ideal in situations where [17]:

1. There exists hazardous working conditions.

2. The lack on skilled workers requires de-skilling the task.
3. The motion or task performed is highly repetitive.
4. Quick response to new jobs is required (This requires the re-programmability of robots).

4.2 Description of Prototypes

Each of the prototype devices is described verbally as well as with pictures in the appropriate section. Mechanical drawings of the devices are given in the appendices.

4.2.1 Internal Seam Alignment Device

The internal seam alignment device was devised as a solution to automating the alignment of the seams prior to pressing. This alignment should be seam-on-seam at the cuff end of the leg as well as at some specified point above the knee (above this specified point the seams begin to diverge). The exact point of divergence is determined by the style and design of the pants. The tolerance on the alignment is one eighth inch.

This device uses two cylindrical rollers that are inserted into the cuff end of a pant leg (see Figure 6 and Appendix 1). One of the rollers is actuated by a stepper motor, the other is a follower (or idler). The idling roller is supported by a bearing block that includes a set of linear bearings mounted perpendicular to the roller shaft. The linear bearings allow the idling roller to move with respect to the stationary roller while maintaining the same orientation. This linear degree of freedom allows the device to accommodate pants with cuff-end widths of up to 12 inches.

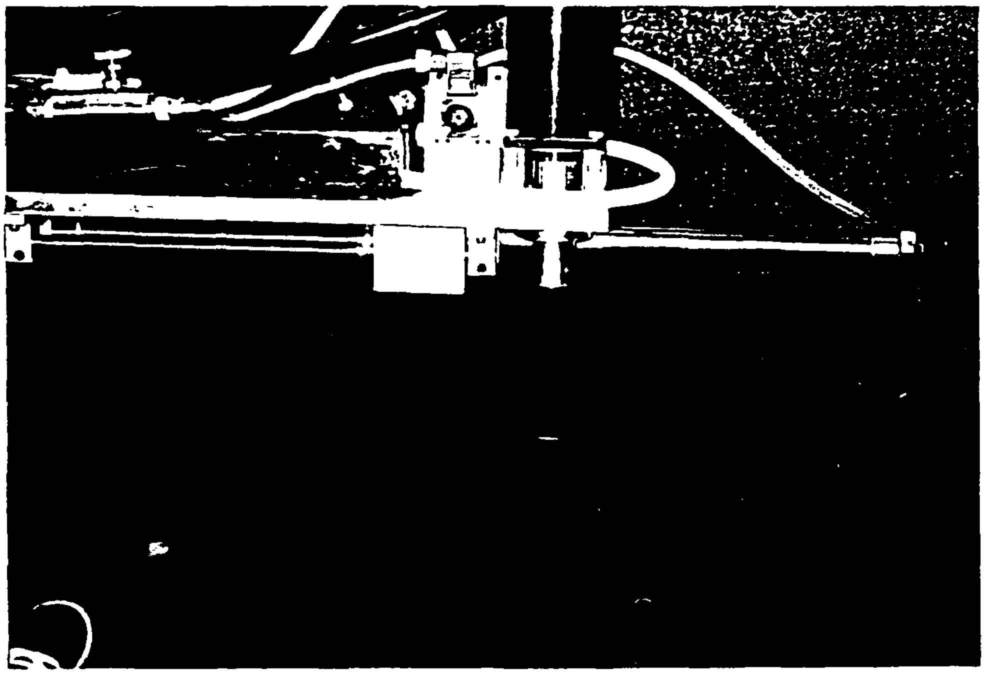


Figure 6: The Internal Seam Alignment Device

The slide is actuated by a pneumatic cylinder. The air supply to the cylinder must be controlled to avoid damaging the fabric or seam threads. This is accomplished by a pressure regulator and volume flow controls. An electronic solenoid valve in the air line is interfaced with a digital output port in the robot. This valve can be switched by a command from the teach pendant or from a robot program.

The stepper motor is controlled by software and a control card installed in the host computer. The resolution of the motor is 200 steps per revolution. Since the roller diameter is approximately 1.025 inches, and the motor directly drives the roller shaft, the linear resolution is .016 inch. Therefore, the accuracy of seam alignment is limited to one sixty-fourth of an inch.

4.2.2 External Seam Alignment Device

A drawing of the external seam alignment device is shown in Appendix 2. This device was prototyped based on the discovery that two adjacent fabric plies can be made to slide relative to each other by using a frictional gripper on the outside of each ply (see Figure 7). The most important criterion that must be met is that the coefficient of friction between the grippers and the fabric be greater than the coefficient of friction between the two fabric plies. An efficient method of moving the fabric plies was found to be rubber coated rollers. The rollers have to be driven in the same rotational direction if no net motion of the fabric is to be experienced. The rollers should be driven with a single actuator so that the rotational speed and the starting and stopping of the rollers will be synchronized. If each roller was driven independently a control system would have to be developed to to ensure synchronization. A motor with a double ended shaft is ideal as a single actuator, since no gearboxes are necessary to split the rotary motion and drive each roller.

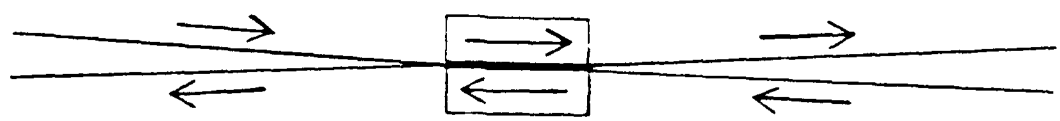


Figure 7: Relative Sliding of Adjacent Fabric Plies

This device could be used in conjunction with the internal seam alignment device. The internal seam alignment device works well at the cuff end of the pant leg. There are two reasons why it would not work well at the waist end. First, the rollers would be difficult to insert into the waist end and past pockets without human aid. Second, the shape of the pant leg is diverging from the cuff end to the waist end. This could cause the pant leg to walk off of the rollers at the waist end of the leg. These problems can be avoided by moving the seams relative to each other using a method which is external to the pant leg. There would have to be one of these devices on each side of the pant leg to move the seams and maintain the stability of the pant leg (see Figure 8).

The external seam alignment device can also be used as a mechanism with which to tension the pant leg laterally (perpendicular to the seam orientation). If the rollers are restricted from rotating then a linear force can be transferred through them to the pant leg. As will be explained later, a lateral tensioning force is required near the waist end of the leg in order for the transfer gripper to grasp the pants without inducing wrinkles.

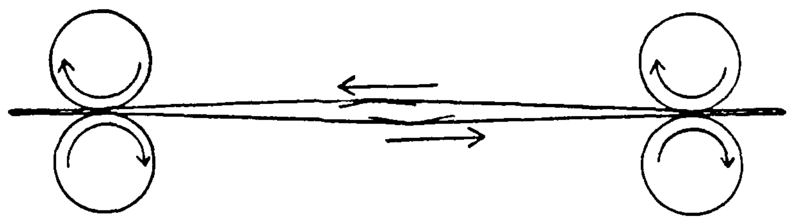


Figure 8: External Seam Alignment Device Scenario

4.2.3 <u>Mechanical Seam Detection Mechanism</u>

The mechanical seam detection mechanism is shown in Figure 9 and in Appendix 3. This prototype was developed as a low cost and reliable method of locating a seam. The change of the number of fabric plies at the seam is used with a thickness measuring scheme to identify the location of the seam.

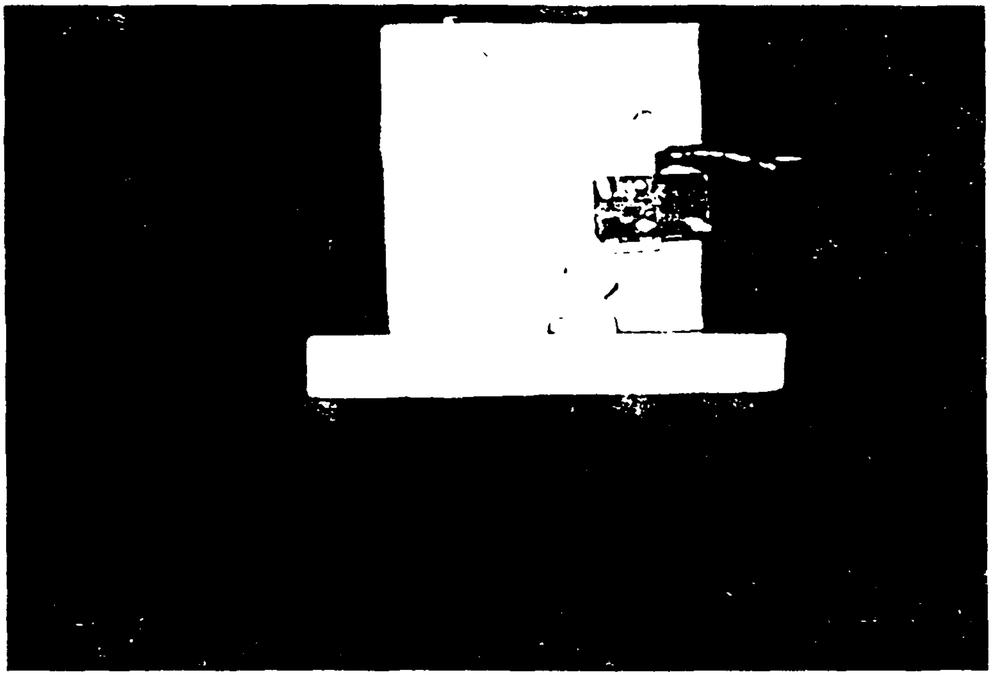


Figure 9: Mechanical Seam Detection Mechanism

The cuff end of the leg is loaded into the device as shown in Figure 10. The leg is then moved perpendicular to the seam orientation. When the seam approaches the detection apparatus the selvage is caught by one of the two selvage catches, depending on the direction of motion. (see Figure 11). The caught selvage is flipped on top of the selvage on the other side of the seam (see Figure 12). A momentary contact switch is activated by the three fabric plies as they pass between the base and the switches roller.

Figure 10: Pants Loaded into Mechanical Seam Detection Mechanism

An earlier version of this device did not have selvage catches on the base. In this case, the double thickness of fabric at the seam was going to be measured and the position of the seam was going to be found indirectly be assuming that the seam was one-half of the distance between the selvage beginning and ending points.

Tests showed that this method was inconsistent since the selvage would randomly catch and be flipped by the base. The selvage catches were employed to consistently catch and flip the selvedge and the measurement scheme was changed. Instead of measuring where the selvage begins and ends, the microswitch is activated exactly where the seam is located (see Figure 13). This measurement scheme is direct because it requires no additional calculations to locate the seam.

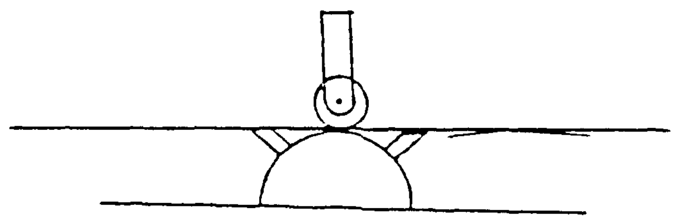


Figure 11: Seam Approaching Selvage Catch

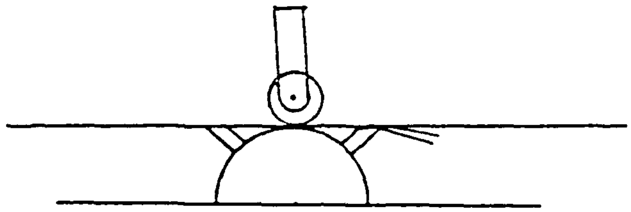


Figure 12: Seam Folded Back by Selvage Catch

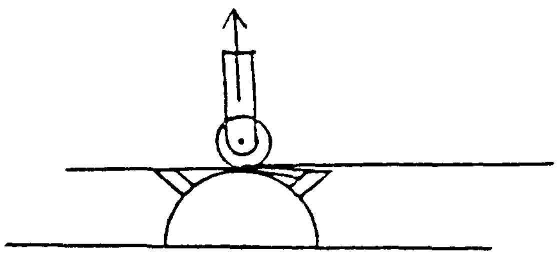


Figure 13: Switch Activated at Seam Location

Theoretically the seam detection mechanism would have to be adjusted for different thicknesses of fabric. However, in most cases adjustment is not needed. Since three fabric plies are used to activate the switch, instead of two, the device is less sensitive to changes in fabric thickness. This was demonstrated by experiment and is presented in Chapter 5.

4.2.4 Transfer Gripper

The transfer gripper is a two-handed single degree of freedom end-effector that mounts on the tool face plate of the robot (see Figure 14 and Appendix 4).

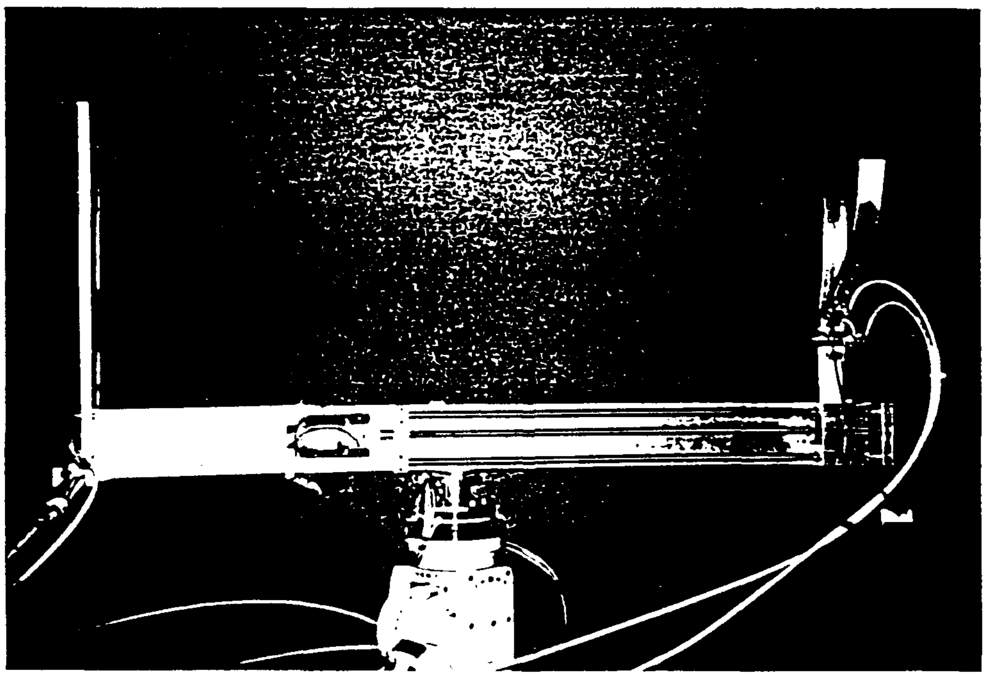


Figure 14: Transfer Gripper

The left hand gripper is a pneumatically actuated four bar linkage (see Figure 15). The linkage allows the gripper to occupy less space when it is open compared to other kinematic linkages. The motion of the gripper fingers resembles parallelogram motion, since the links are close in dimension (see Appendix 4, items 8 and 9).

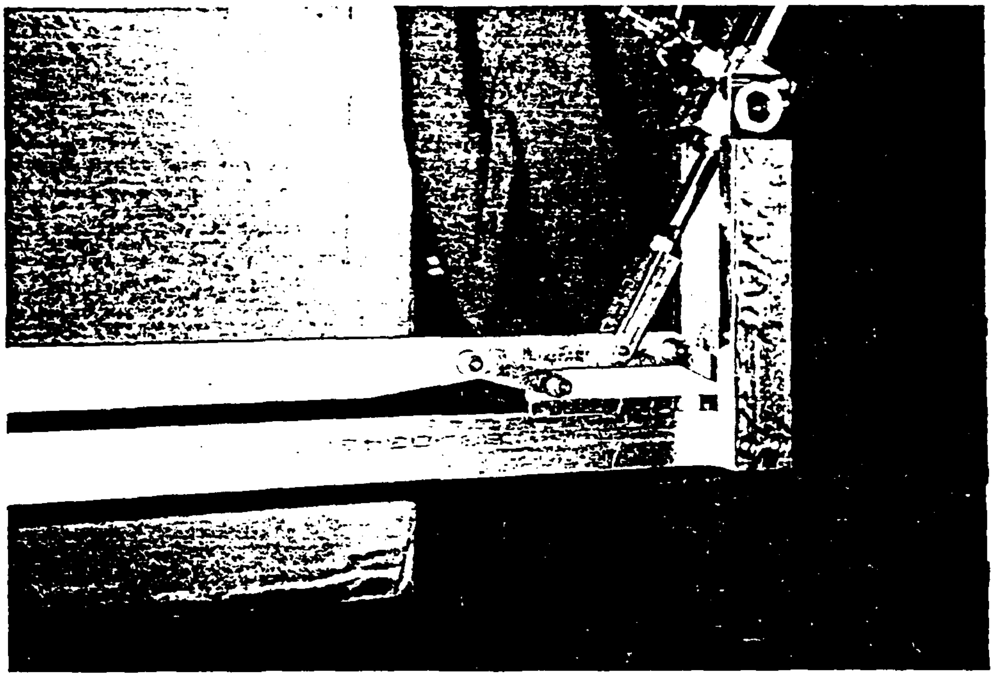


Figure 15: Left Hand of Transfer Gripper

The right hand gripper is a scissors type of gripper, since it has a single revolute joint (see Figure 16). This gripper is also pneumatically actuated. A set of linear bearings allows this gripper to move relative to the left hand gripper. The single degree of freedom provided by the slide serves two purposes. First, the pants can be tensioned in the longitudinal direction (parallel to the seam orientation). The tension keeps wrinkles from being introduced into the pant leg during the transfer process. Second, the slide is long enough so that the right hand gripper can be entirely removed from the press buck. Thus, while the pressing cycle is running the left hand gripper can remain grasping the waist end of the pant leg near the crotch while the right hand gripper is removed from the press buck.

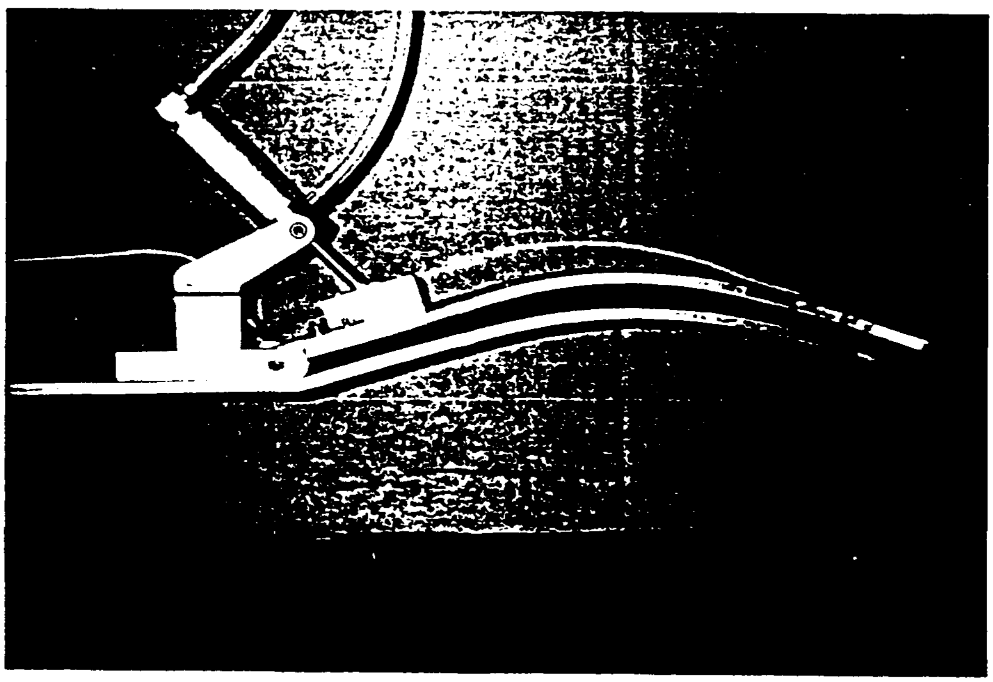


Figure 16: Right Hand of Transfer Gripper

The sliding motion is actuated by a DC servomotor and a ball screw assembly. A servomotor control card and software installed in a PC allow the slide to be accurately positioned. Limit switches mounted on the gripper base indicate the end of motion limits of the slide. A third limit switch provides a home position signal to the motor control card. This is needed because a relative encoder is used on the servomotor.

The air flow to the grippers is controlled by two electric solenoid valves interfaced with the digital output ports of the robot controller. In addition, the air flow rate to the left hand gripper is restricted by port mounted flow controls on the pneumatic cylinder. The air flow rate to the left hand gripper must be restricted to avoid damaging inertial forces while opening the fingers.

Both grippers have adhesive-backed foam rubber tape as the interface between the gripper and the fabric.

CHAPTER 5

TESTING AND RESULTS

5.1 Description of Testing Procedures

The testing procedures for each of the devices is given in flowchart form. A short description at the beginning of each sub-section is used to clarify the testing conditions for each device.

The experiments were conducted using three pants of different fabrics, waist sizes, and style. The fabric and garment specifications for the selected pants are given in Table 3. A variety of fabric types and garment sizes were chosen for several reasons. First, it shows how a device's accuracy and repeatability does or does not vary between different fabric's and garment sizes. This will help to define the useful working range of the device. Second, it will indicate the adjustments and changes that may have to be made to the device in order for it to work successfully over a sufficient range of materials.

For the tests that required distance measurements, a machinists scale with one thirty-second of an inch graduations was used. This allows for a theoretical measurement resolution of one sixty-fourth of an inch on either side of the measured value.

FABRIC NUMBER	I.D. #	FABRIC DESCRIPTION		GARMENT DESCRIPTION			CUFF-END WIDTH (inches)	FABRIC THICKNESS (inches)
		GENERAL	CONTENT	WAIST SIZE	INSEAM LENGTH	CUFF		
1	3	Texfi Blends Style: 53326 Zenith Linear Yield: 10-10.5 oz.	15/1 blend of 65% Fortrel, 35% Rayon	42	37	NO	9 1/2	.016
2	11	Biltmore Textile Style: 27498 Weight: 6 oz./ square yard	100% Cotton	50	37	NO	11 1/2	.018
3	8	Windsor Textile Style: Brown Twill Weight: 11.5 - 12 oz.	100% Wool	28	37	NO	9	.020

Table 3: Fabric and Garment Specifications

5.1.1 Internal Seam Alignment Device

The internal seam alignment device was tested for accuracy, repeatability, and pant leg walking. The accuracy test was used to determine if the seams moved the amount they were expected to move based on knowing the number of steps the motor rotated. The procedure used to test the accuracy of the internal seam alignment device is given in Figure 17.

The repeatability test determined whether the seam would return to its starting position if the motor was commanded to rotate clockwise 300 steps and then counter-clockwise 300 steps. The test procedure is shown in Figure 18. The repeatability is perhaps the most important performance parameter of this device. In most mechanical devices, if the accuracy is less than required it will most likely be an amount that remains unchanged or slowly degrades as the device wears. Thus, this constant amount can be accounted for in the design of other components in the device or in the control system. However, if the accuracy of the device varies randomly (its repeatability is low) then the accuracy offset cannot be corrected, since its value is not known.

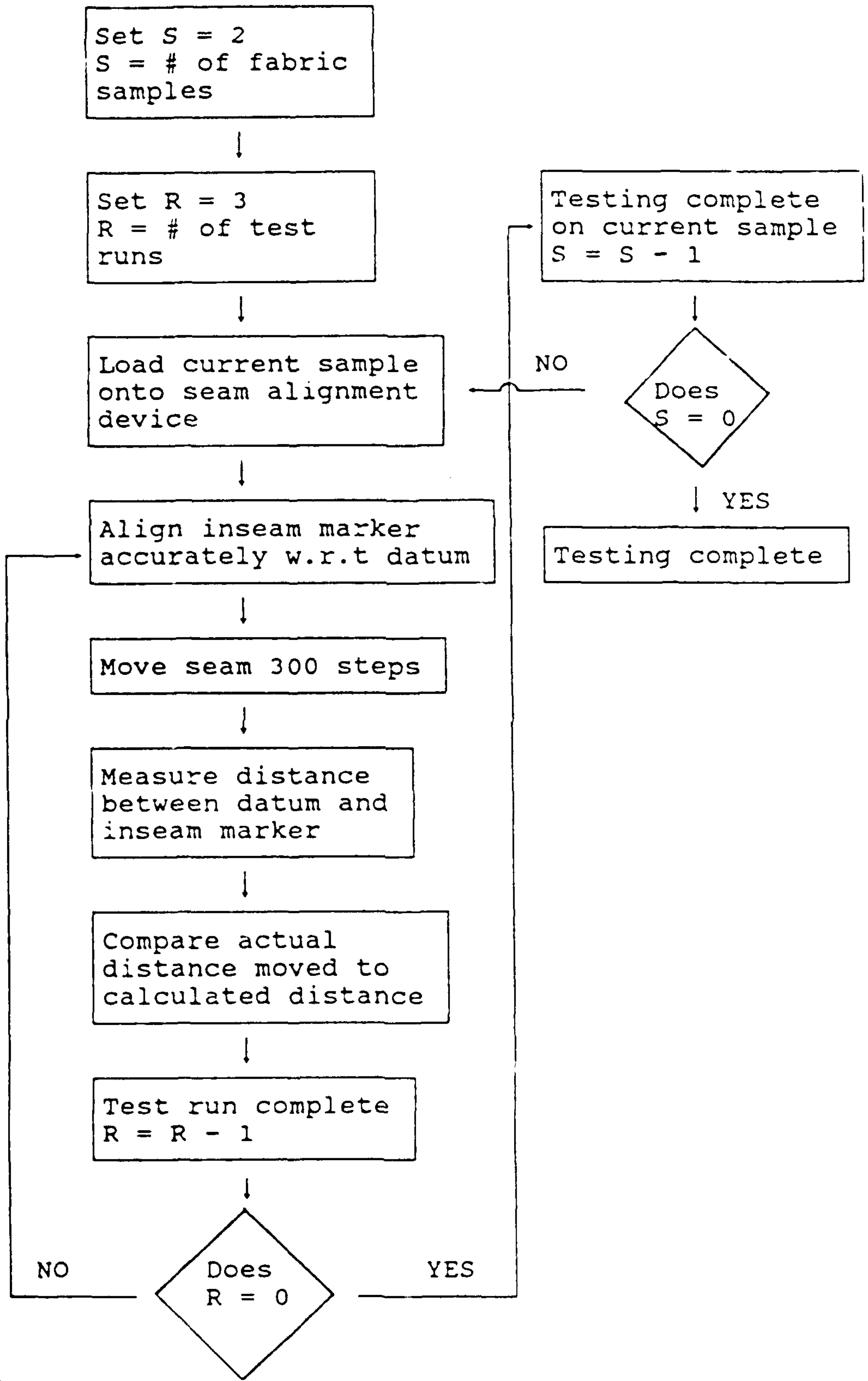


Figure 17: Accuracy Testing Procedure

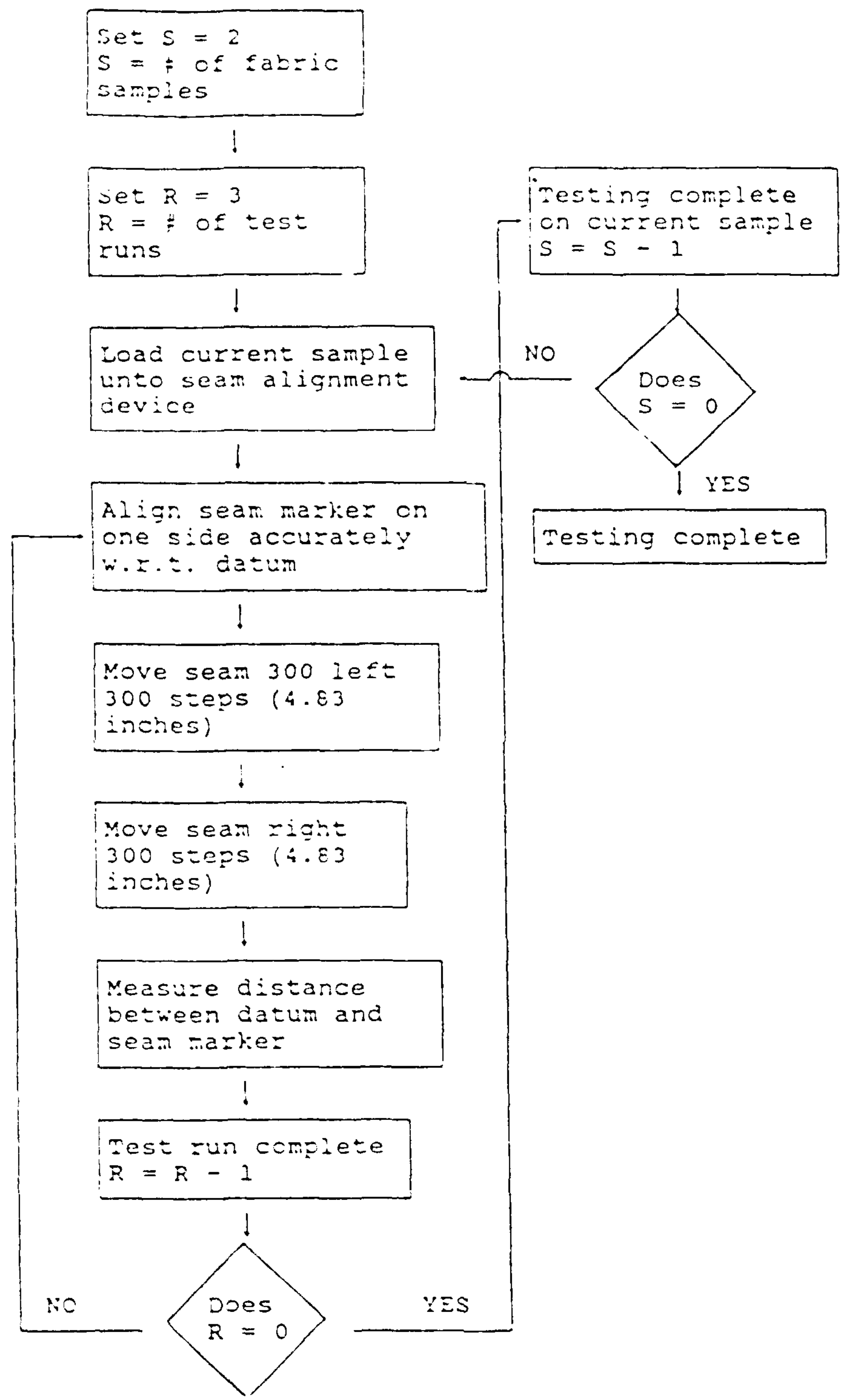


Figure 18: Repeatability Testing Procedure

The walking of the pants as they are suspended from the internal seam alignment device was also investigated. The pants have a tendency to move down the rollers as the rollers are rotating due to the effect of gravity. The phenomena of walking should be differentiated from slippage.

Consider the situation shown in Figure 19. If the rollers are in a vertical orientation the material will experience a gravitational force proportional to its weight in the downward direction. If the material is rigid it experiences little deformation for a given load and the only way the material can move relative to the rollers is by slipping. If the material is flexible, as is fabric, then the vertical force can deflect the material in between the rollers (Figure 20). When this deflected portion reachs the roller it is simply maintained at its same vertical position (Figure 21). As rotation continues the material proceeds to walk down the roller and will eventually leave the roller.

The experimental procedure used to determine the amount of walking for two different pants is shown in Figure 22.

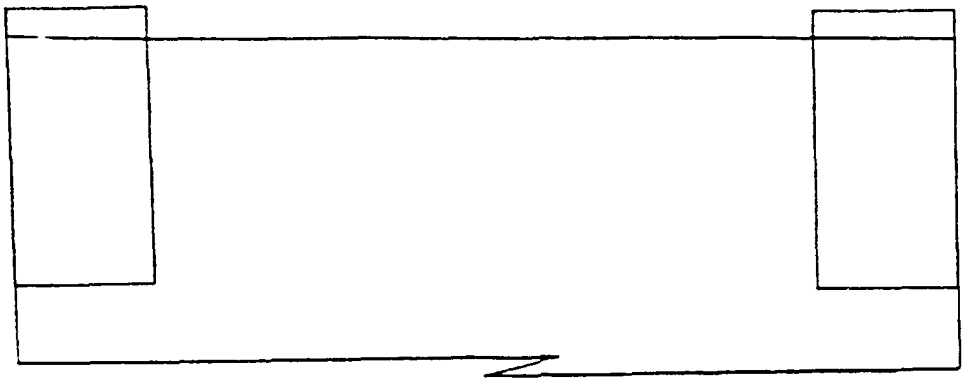


Figure 19: A Rigid Material on Vertical Rollers

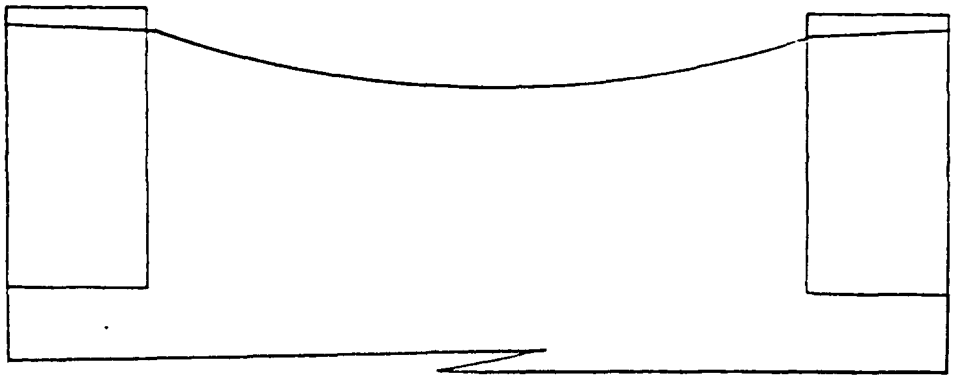


Figure 20: A Flexible Material on Vertical Rollers

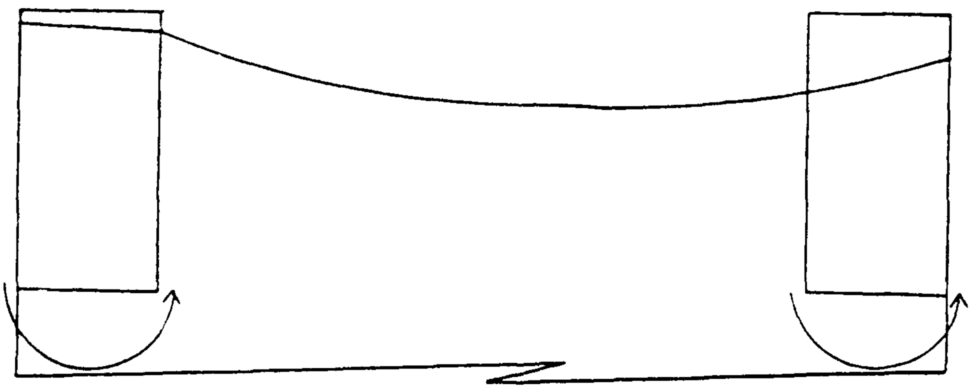


Figure 21: Flexible Material Walking Phenomena

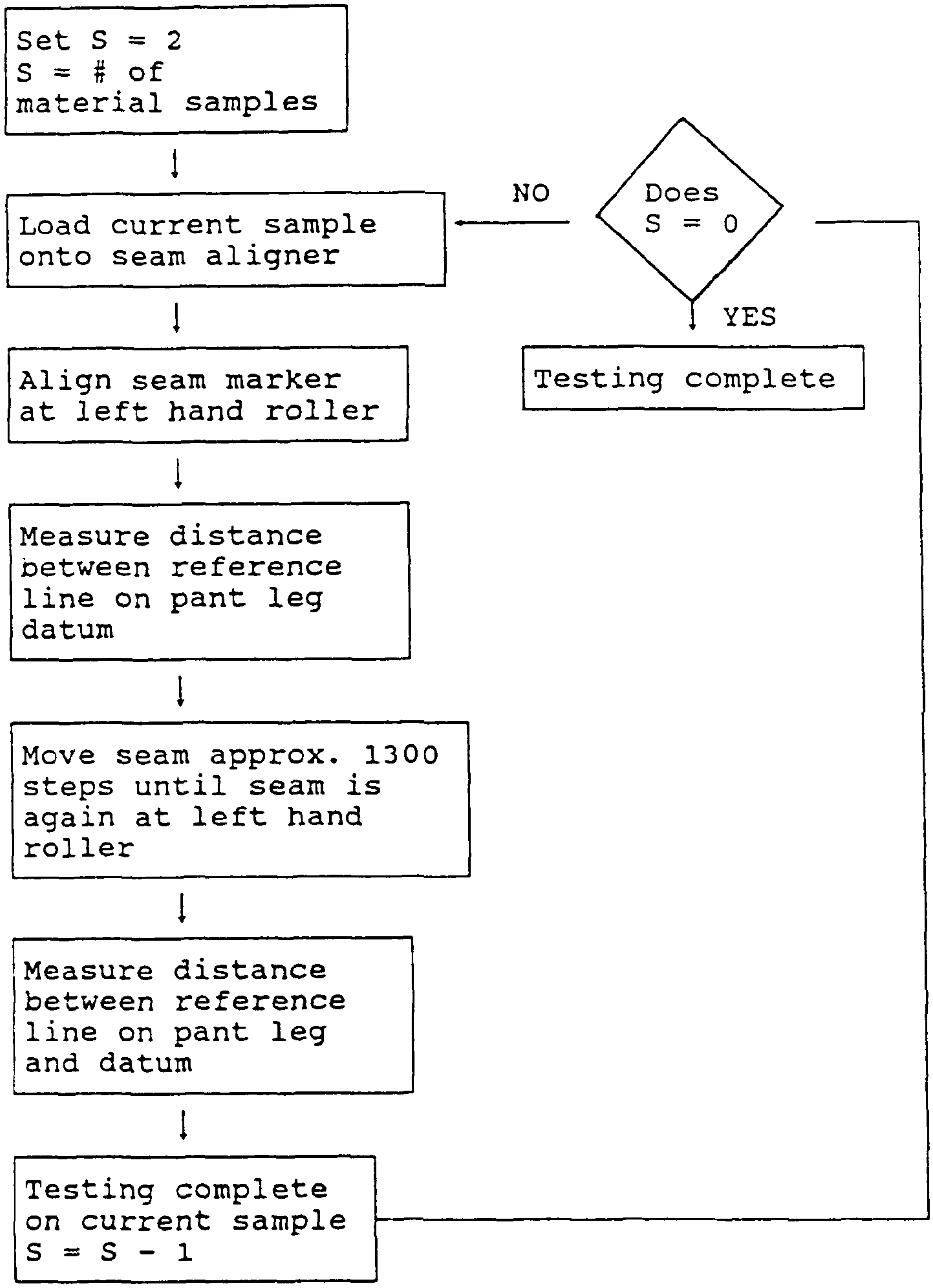


Figure 22: Walking Testing Procedure

5.1.2 External Seam Alignment Device

No formal testing was conducted with the external seam alignment device. However, prototype was tested informally by observing the response of several fabrics as the rubber rollers were manually actuated. Several useful obervations were made as a result of this testing. The results of the informal tests are given in section 5.2.2.

5.1.3 Mechanical Seam Detection Mechanism

The mechanical seam detection prototype was tested with the four different fabrics given in Table 4, using the procedure described in Figure 23. The pants were manually moved through the mechanism.

FABRIC	THICKNESS (inches)	CUFF	SELVAGE WIDTH (inches)	BLEND
A	.012	YES	7/8	POLY/WOOL
B	.012	NO	15/16	POLY/WOOL
C	.014	NO	15/16	POLY/WOOL
D	.021	NO	1	100% WOOL

Table 4: Fabric Specifications for the Mechanical Seam Detection Experiments

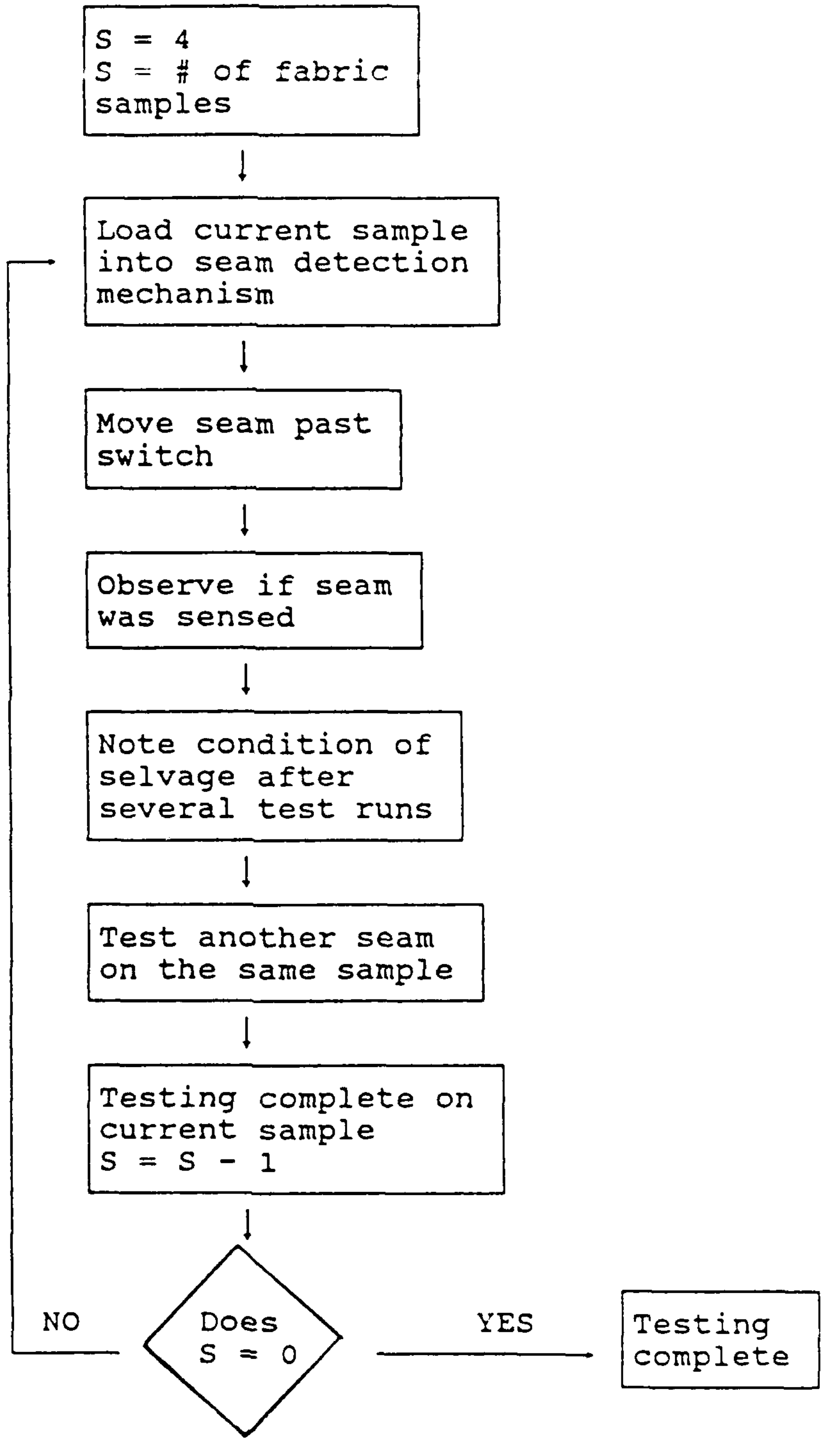


Figure 23: Mechanical Seam Detection Testing Procedure

5.1.4 Transfer Gripper

The transfer gripper was tested with three different pants since it was expected that the performance of this device would be the most effected by changes in the fabric or garment specifications. A stationary reference, or datum, was used to determine the relative locations of the seams both before and after the transfer of the pants to the press. In some cases the seams were not perfectly aligned before transfer. This is inconsequential since the objective of the experimental was to determine if the seam alignment changed during the grasping and transfer process.

A small amount of tension was applied to the pants immediately after they were grasp and before transfer to the press started. The tension was applied by moving the right-hand gripper an additional .30 inch after the grippers were closed. This kept wrinkles from forming in the pant leg during transfer.

The testing procedure is given in Figure 24. The robot program code used during the testing of the transfer gripper is given in Figures 25 and 26.

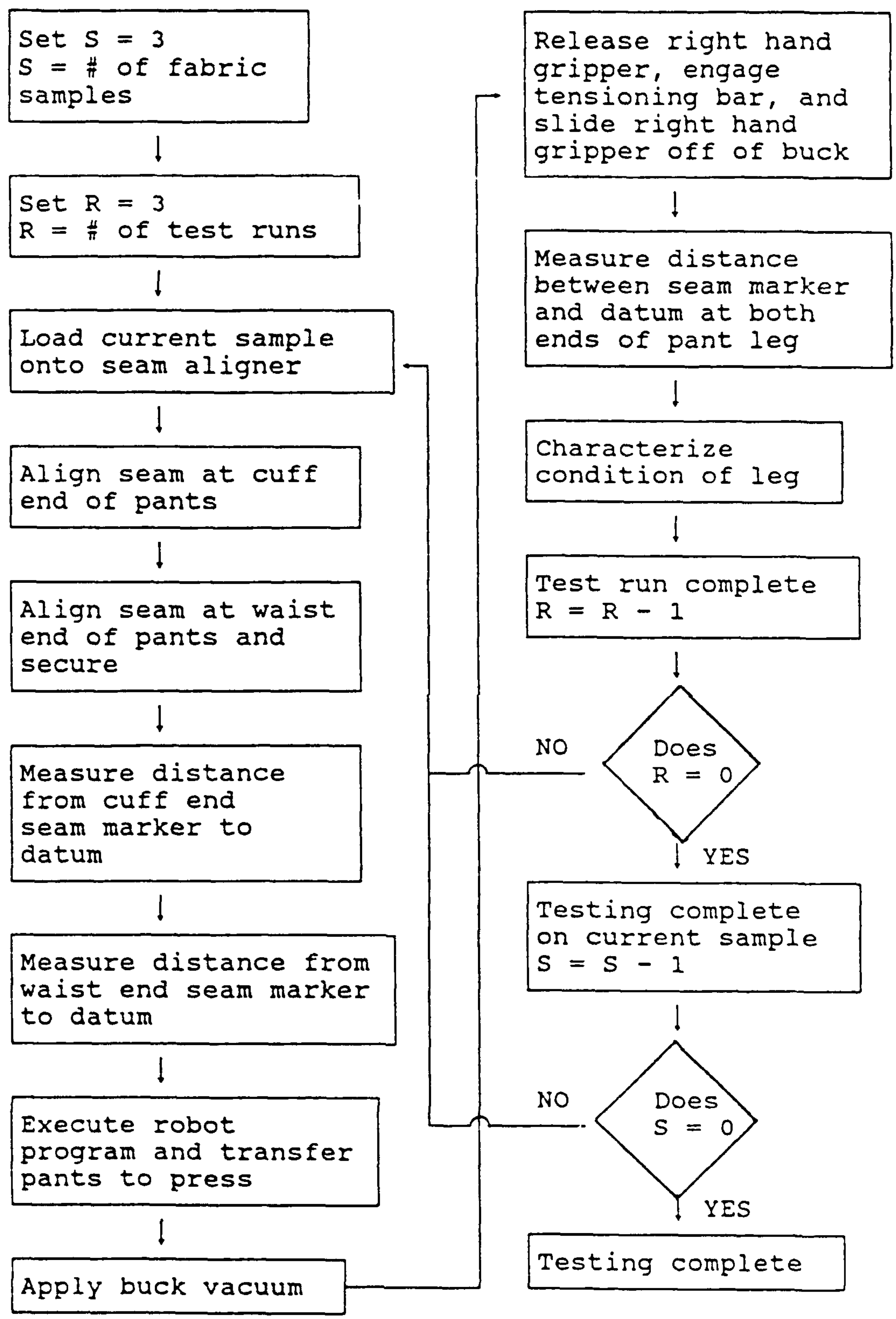


Figure 24: Transfer Gripper Testing Procedure

```
program grasp

-- TRANSFER GRIPPER TESTING PROGRAM


-- initialization

var
clear_roll,near_roll,behind_gripp,grippoint,below_gripp:position
intermediat1,intermediat2,clear_press,above_press,above_press2:position
release_pnt:position
begin
$uframe=$nilp
$utool=$nilp
$termtype=nodecel
$motype=joint
$speed=300

-- set all outputs to the default, grippers open and roll-align tensioned

dout[1]=off
dout[2]=off
dout[3]=off

-- move clear of roll-align and press

move to intermediat2
move to intermediat1

-- move to point clear_roll, just clear of roll-align

move to clear_roll

-- move close to pants, then behind them to approach gripping position

move to near_roll
$motype=linear
move to behind_gripp

-- move to gripping position, then actuate grippers

$termtype=fine
$speed=150
move to grippoint
delay 1500
dout[1]=on
dout[2]=on

-- pause to tension slide

pause

-- release pants from roll-align

delay 1000
dout[3]=on

-- move pants down to clear roll-align

move to below_gripp

-- first clear roll-align, then continue transfer to press

$termtype=nodecel
```

Figure 25: GMF Robot Transfer Testing Program (page 1)

```
$speed=300
move to clear_roll
$motype=joint
move to intermediatl
move to intermediat2
move to clear_press

-- approach to press, first move above release point, then descend
-- in two steps

$speed=150
$motype=linear
move to above_press
move to above_press2
$termtype=fine
move to release_pnt

-- pause to apply buck vacuum

pause

-- open right-hand gripper

dout[1]=off

-- pause to engage tension bar and slide right gripper off of buck

pause

-- move away from press to safely restart cycle

move to above_press2
move to above_press
$speed=300
$termtype=nodecel
move to clear_press
$motype=joint
move to intermediat2
dout[2]=off

-- end program

end grasp
```

Figure 26: GMF Robot Transfer Testing Program (page 2)

5.2 Experimental Results

The results of each of the experiments performed are given in tabular form in this section, with the exception of the results of the informal testing of the external seam alignment device. The raw data is given as well as the derived data. The results will be discussed in Chapter 6.

5.2.1 Internal Seam Alignment Device

The results of the accuracy, repeatability, and walking tests for the internal seam alignment device are given in Tables 5, 6, and 7, respectively.

5.2.2 External Seam Alignment Device

The external seam alignment prototype was tested on a variety of fabrics with both polyurethane rollers and rubber rollers. Both of these interfacial materials provided excellent frictional gripping of the fabrics. The clamping force that holds the rollers against the fabric was provided by a spring and was very small (about a pound). Even with this small normal force the frictional force was great enough so that no fabric slippage was experienced between the rollers and the fabric.

FABRIC NUMBER	TEST RUN	BEFORE MOVEMENT	AFTER MOVEMENT	Actual distance moved (inches)	Difference between actual and theoretical
		Measurement to datum (in.)	Measurement to datum (in.)		
FABRIC 1	Run 1	10.72	5.75	4.97	.08
	Run 2	10.72	5.78	4.94	.05
	Run 3	10.72	5.75	4.97	.08
FABRIC 2	Run 1	9.88	4.94	4.94	.02
	Run 2	9.88	4.94	4.94	.02
	Run 3	9.88	4.91	4.97	.05

Table 5: Results of Internal Seam Alignment Device Accuracy Tests

FABRIC NUMBER	TEST RUN	BEFORE MOVEMENT	AFTER MOVEMENT	Difference between measurements (inches)
		Measurement to datum (in.)	Measurement to datum (in.)	
FABRIC 1	Run 1	10.50	10.47	.03
	Run 2	10.47	10.47	.00
	Run 3	10.47	10.44	.03
FABRIC 2	Run 1	9.88	9.88	.00
	Run 2	9.88	9.88	.00
	Run 3	9.88	9.88	.00

Table 6: Results of Internal Seam Alignment Device Repeatability Tests

FABRIC	BEFORE MOVEMENT	AFTER MOVEMENT	NUMBER OF STEPS TRAVELED/ INCHES TRAVELED	DISTANCE WALKED (inches)
	MEASUREMENT FROM MARKER TO DATUM(in.)	MEASUREMENT FROM MARKER TO DATUM(in)		
1	1.00	1.78	1270/ 20.50	.78
2	1.25	1.75	1450/ 23.35	.50

Table 7: Results of Internal Seam Alignment Device Walking Tests

5.2.3 Mechanical Seam Detection Mechanism

The results of the mechanical seam detection experiments are given in Table 8.

FABRIC	NUMBER OF TEST RUNS	NUMBER OF SUCCESSFUL DETECTIONS	NUMBER OF FAILURES
A	40	36	4
B	45	42	3
C	56	55	1
D	43	40	3
TOTALS	184	173	11

Table 8: Results of Mechanical Seam Detection Mechanism Tests

5.2.4 Transfer Gripper

The raw data of the transfer gripper tests are given in Tables 9, 10, and 11. The derived results are given in Table 12. It should be noted that none of the pants used in testing had previously been pressed.

	BEFORE TRANSFER (measurements in inches)				AFTER TRANSFER (measurements in inches)				WRINKLE CHARACTERIZING (#) height/length (inches)
Run #	Cuff Front	Cuff Rear	Waist Front	Waist Rear	Cuff Front	Cuff Rear	Waist Front	Waist Rear	
1	7.56	7.56	10.75	10.03	10.63	10.63	10.56	9.91	(1) .6H / 1L (1) .2H / 2L
2	7.56	7.56	10.56	10.44	10.75	10.69	10.38	10.25	(1) .5H / 1L (2) .1H / 3L
3	7.56	7.556	10.41	10.00	10.53	10.69	10.25	9.84	(1) .5H / 1L (1) .2H / 3L (1) .1H / 4L

Table 9: Results of Transfer Gripper Tests for Fabric 1

	BEFORE TRANSFER (measurements in inches)				AFTER TRANSFER (measurements in inches)				WRINKLE CHARACTERIZING (#) height/length (inches)
Run #	Cuff Front	Cuff Rear	Waist Front	Waist Rear	Cuff Front	Cuff Rear	Waist Front	Waist Rear	
1	9.97	10.19	10.72	10.53	11.06	11.38	11.50	11.25	(1) .1H / 14L (1) .2H / 5L (1) .1H / 4L
2	9.94	10.06	10.88	10.38	11.06	11.31	11.75	11.31	(1) .2H / 14L (1) .1H / 5L (1) .1H / 6L
3	10.00	10.13	10.69	10.31	11.06	11.31	11.50	11.13	(1) .2H / 14L (1) .1H / 4L (2) .1H / 5L

Table 10: Results of Transfer Gripper Tests for Fabric 2

	BEFORE TRANSFER (measurements in inches)				AFTER TRANSFER (measurements in inches)				WRINKLE CHARACTERIZING (#) height/length (inches)
Run #	Cuff Front	Cuff Rear	Waist Front	Waist Rear	Cuff Front	Cuff Rear	Waist Front	Waist Rear	
1	9.97	10.19	10.72	10.53	11.06	11.38	11.50	11.25	(1) .1H / 14L (1) .2H / 5L (1) .1H / 4L
2	9.94	10.06	10.88	10.38	11.06	11.31	11.75	11.31	(1) .2H / 14L (1) .1H / 5L (1) .1H / 6L
3	10.00	10.13	10.69	10.31	11.06	11.31	11.50	11.13	(1) .2H / 14L (1) .1H / 4L (2) .1H / 5L

Table 11: Results of Transfer Gripper Tests for Fabric 3

FABRIC NUMBER	AVERAGE SEAM MOVEMENT	
	CUFF END (INCHES)	WAIST END (INCHES)
1	.07	.03
2	.12	.04
3	.12	.04

Table 12: Average Seam Movement during Pants Transfer

CHAPTER 6

DISCUSSION AND CONCLUSIONS

The test results indicate that future efforts in this area should concentrate on several areas. The first area is seam alignment at the waist. The informal testing done on the external seam alignment device has shown promising results. The prototypes should be further developed to allow pneumatic opening and closing and sliding of the two device for lateral tensioning of the leg. Pneumatic cylinders would provide a cost effective and simple method of closely controlling the clamping force of the rollers. It is necessary to control this force, so a variety of fabrics and their associated frictional characteristics can be accomodated.

The tests on the transfer gripper also showed some room for improvement. The fact that the seam alignment is not maintained to a sufficient degree of accuracy warrants further development effort. It should be noted that the largest change in seam alignment occurs at the cuff end of the pant leg (see Table 12). Observations during testing indicate that the source of the problem may be in the design of the gripper mechanism. The distortion in the leg as the gripper is closed seems to be caused by the fact that the side of the leg closest to the gripper is experiencing clamping forces before the side of the leg farthest from the gripper. This is due to the nature of the scissors type clamping geometry. A semi-parallel gripper like is used at the waist area may be a solution.

It was also noticed during the course of testing that the lateral tension (perpendicular to the seam orientation) on the leg was a critical factor in determining if the leg could be laid on the press relatively wrinkle free. If this tension was not sufficient or incorrectly oriented the fullness in the leg near the waist area would bunch up as the gripper closed.

The typical tranfer process starts with grasping the pant leg on the seam alignment device as shown in Figure 27. Then longitudinal tension is applied with the servomotor and the cuff end of the leg is released from the seam alignment device. The robot then guides the gripper to the press. Note that in Figure 29, the extreme cuff end of the leg is free to droop over the right hand gripper. The path of the robot has been programmed to aid in reducing wrinkles caused by this dangling end. This is done by positioning the gripper at a position farther to the right of its releasing point and then moving the leg to the left as it descends to the buck (see Figure 30). The typical cotton pant leg after transfer is shown in Figure 33. A sequence of the transfer is shown in Figures 27 through

32.

Figure 27: Grasping the Pant Leg at the Seam Alignment Station

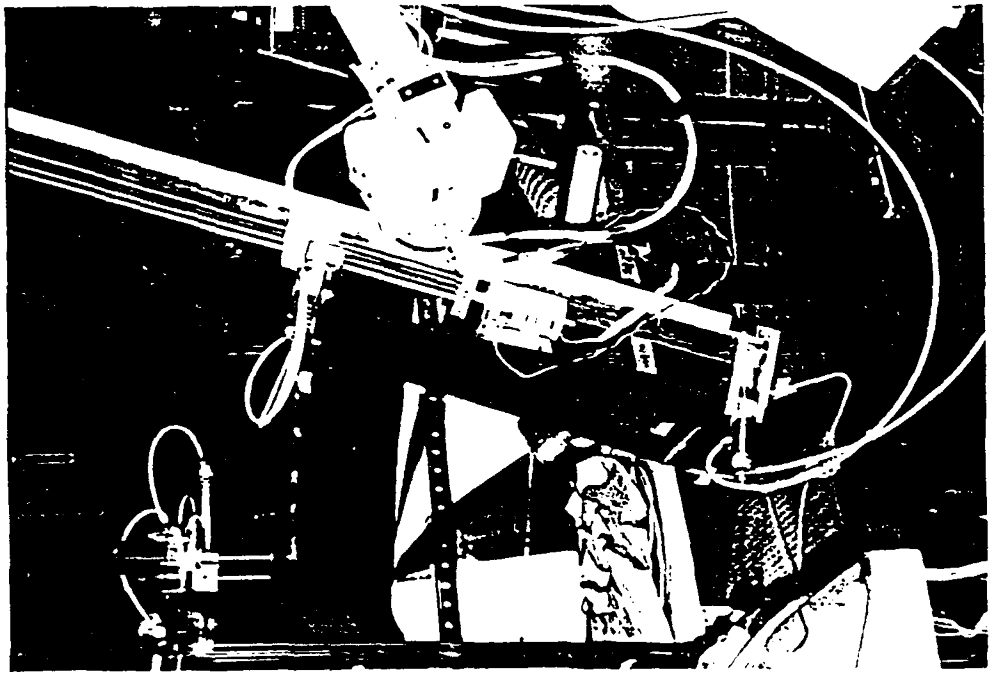


Figure 28: Pants Beginning the Transfer Process

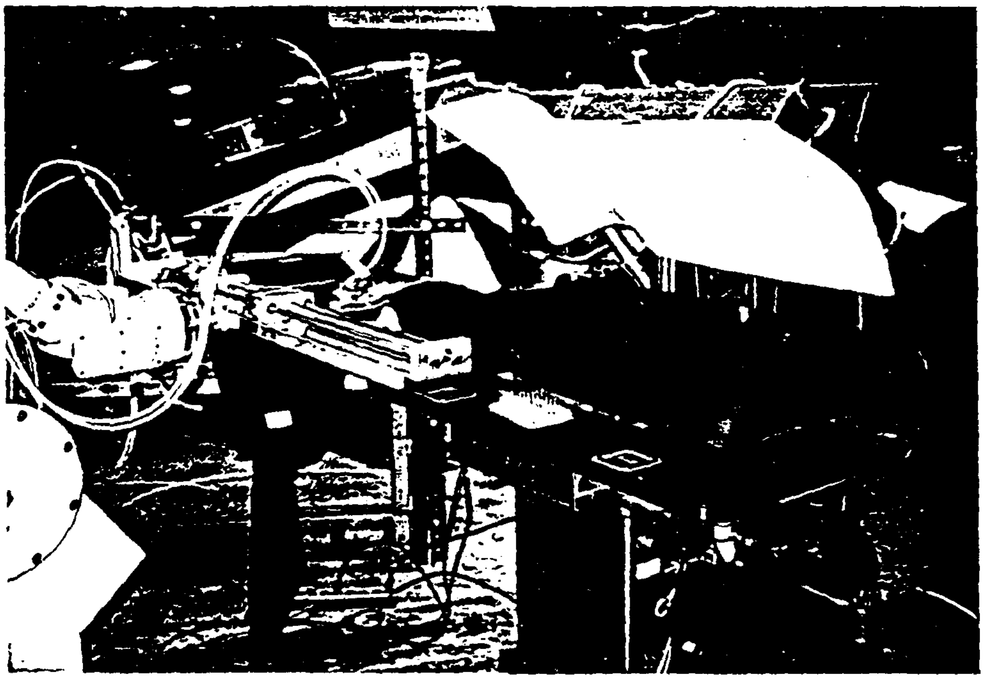


Figure 29: Pants Approaching Press

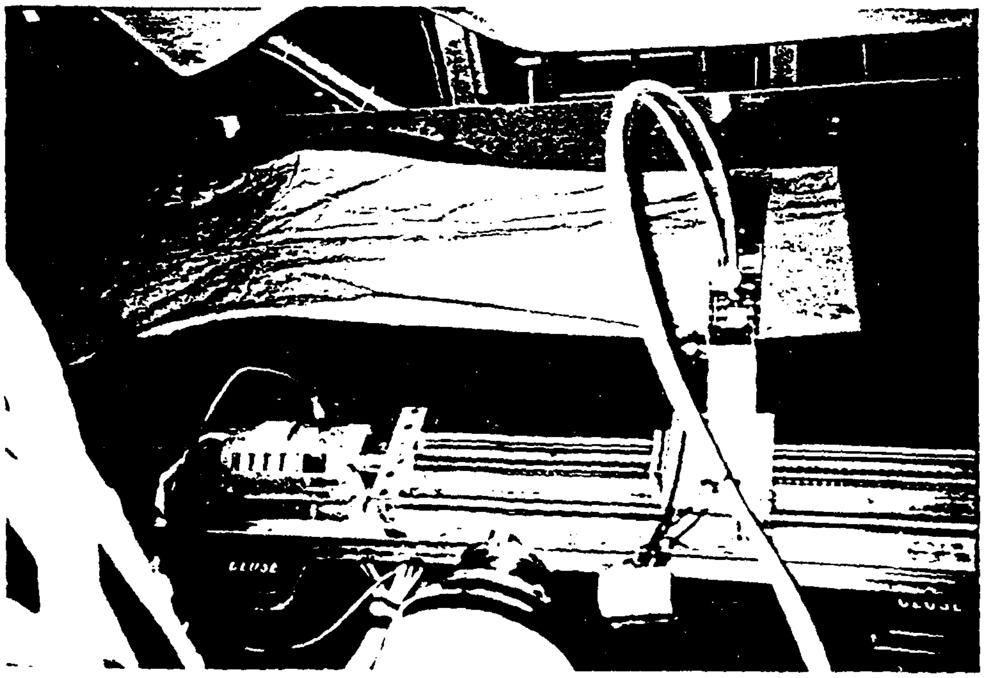


Figure 30: Pants Laid on Press Buck

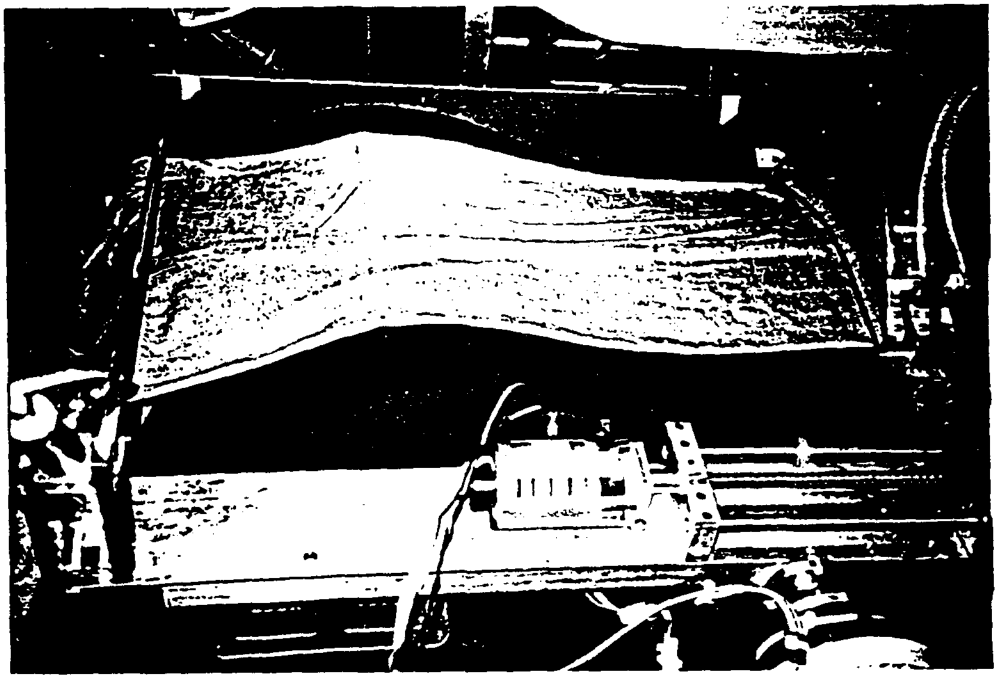


Figure 31: Pant on Buck with the Tensioning Bar is Engaged

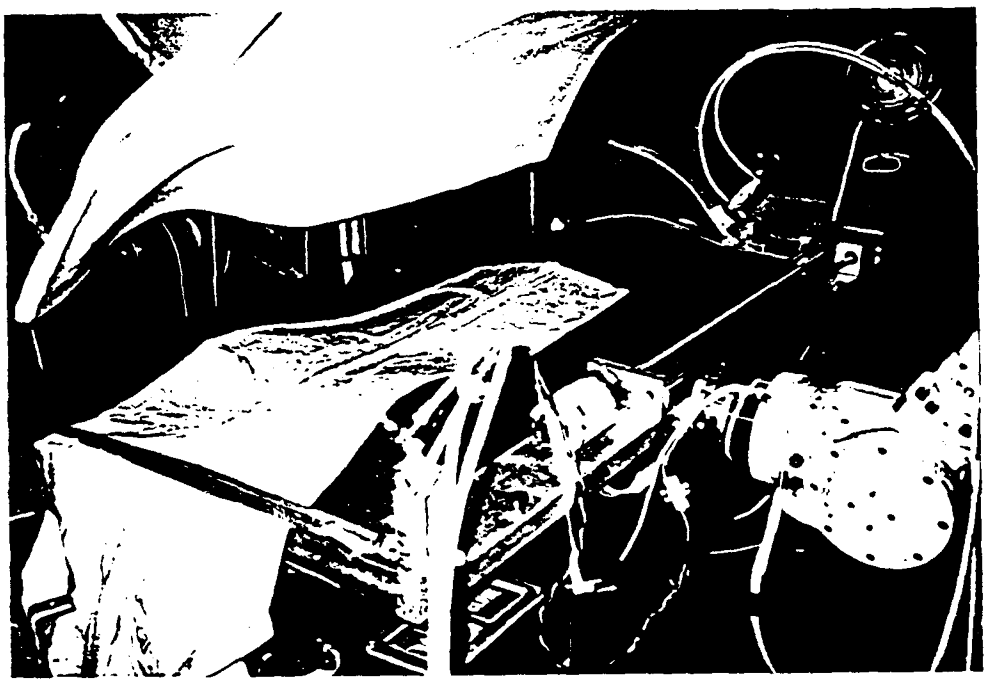


Figure 32: Pants on Buck with the Right Hand Gripper Removed

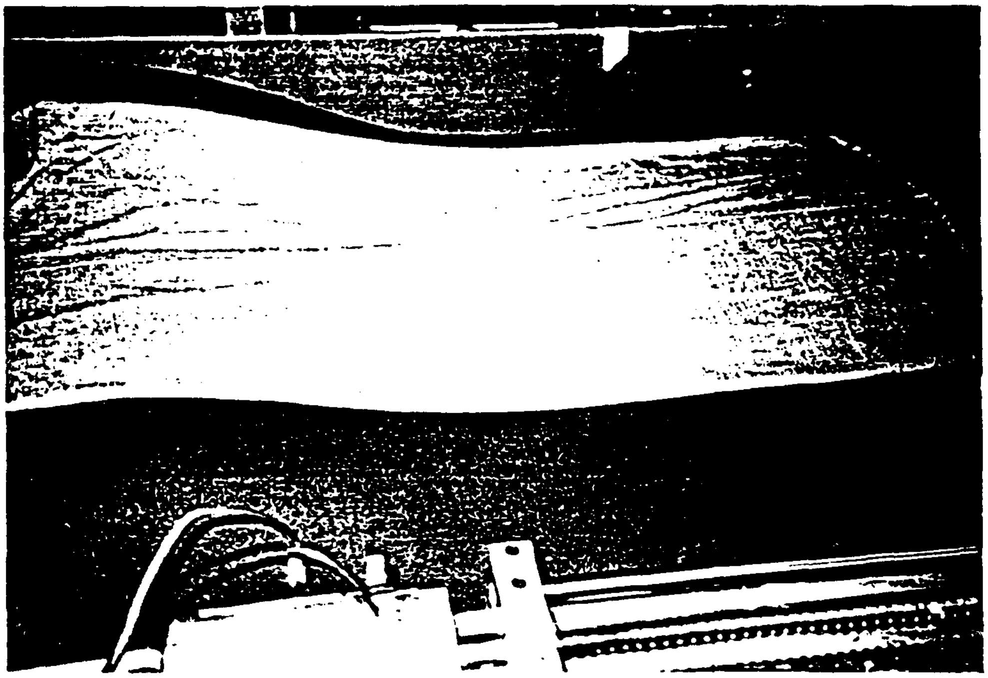


Figure 33: Typical Cotton Pants Lay-up

During the testing procedures no single wrinkle could be identified as being too large that it would be pressed into the leg. All of the wrinkles characterized in Tables 9, 10, and 11 were located in the top ply of the leg. Wrinkles in the bottom ply are almost certain to be pressed into the leg. Further research should be conducted to determine what sizes and number of wrinkles are acceptable in various fabrics, before pressing is initiated, and not be pressed in.

Based on the quality of the transfer process it is believed that more than two grippers are not needed to grasp and manipulate the leg. The longitudinal tension applied to the leg kept the fabric from drooping in between the grippers. However, it may be advantageous to provide some means of support to the extreme cuff end of the leg.

The results of the internal seam alignment tests show that this is an acceptable solution to the seam alignment problem. Even though the accuracy is not a good as expected, the repeatability is very high. The accuracy of the movement seemed to be dependent on the amount of walking of the leg (see Tables 5 and 7). The walking of the pant leg can be controlled by a finger oriented in such a way as to force the fabric upward as the pants move.

The seam detection mechanism has a 94% success rate in reliably detecting the seam. The probable sources of most of the failures were identified during testing. First, as with the transfer gripper, it is important that proper lateral tension be utilized. If the lateral tension in the fabric was not sufficient the selvage would ride over the selvage catch and not be flipped. This caused the switch to be activated before the seam reached it. If the detection mechanism was an integral part of the internal seam alignment device then the tension would be consistent and controllable. Second, the quality of the switch should be improved. It was noted in several cases that the switch got bound in its "on" position by small movements of the fabric parallel to the switches roller axis.

There are many problems that must be overcome to successfully grasp and manipulate pants in the pressing operation. Perhaps the most challenging problems arise from fabric's handling characteristics. The work at Draper Laboratories, in developing the automated suit-coat sleeve sewing machine, revealed many of these problems.[18] Many of the same problems were encountered at Daimler Benz in Germany when trying to automate the process of inserting flexible insulating material into cars.[19]

1. The manipulation of fabric parts requires the use of multiple gripping points.
2. Fabric buckles under its own weight.
3. Excessive manipulation can cause wrinkles.
4. The low rigidity of fabric practically eliminates the application of force feedback or contact sensors.
5. The mechanical properties of fabric, like its stiffness and coefficient of friction, vary from part to part and may even vary across a single part.
6. Mechanical properties are dependent upon environmental conditions like humidity and are time-dependent.
7. Precise alignment can be spoiled by residual

stresses in the part.

8. Usually it is inefficient to attempt to duplicate human manipulation motions, although much can be learned from them.

The steps that were taken to counteract these problems were:

1. Gated vacuum tables were used to hold sections of the part from moving while the remainder of the part was being manipulated.
2. A flexible gripper was needed to accomodate various part shapes.
3. Air jets were used for non-contact manipulation.
4. Creasing bars and auxiliary manipulation devices were used when standard gripping is not sufficient in itself.
5. Non-contact sensing was used when the part could not be disturbed.
6. Sensors were used to determine if grasping and manipulation occurred correctly. If not, then the gripper and manipulation path are adjusted and the sequence is tried again.
7. Use a unique trajectory for manipulating parts made of different fabrics. This helps to account for the mechanical property variations from fabric to fabric.

Other problems exist because of the nature of the apparel industries' manufacturing methods, the existing workforce, and their traditional attitudes:

1. Part geometries vary considerably because tolerances build-up from previous operations or the operations themselves are inconsistent.
2. A simple user interface and simplicity of operation are needed to allow for untechnical operators. Ease of maintainance is critical. Most apparel manufacturers do not have maintainance technicians experienced with high-technology machinery. Off-the-shelf components should be used to allow for quick repair times.
3. There exist few quantitative measures of performance with which to judge your system.
4. The financial philosophies of the apparel industry must be reprogrammed to accept longer payback periods and larger initial capital investments.

REFERENCES

1. Augusto Carroli, "Potential Uses for Robotics in the Textile and Allied Industries," in Automation and Robotics in the Textile and Apparel Industries, (Noyes Publications, 1986), pg 97.

2. T. W. George, "Aspects of Robotic Material Handling in Textiles," in Automation and Robotics in the Textile and Apparel Industries, (Noyes Publications, 1986), pg 135.

3. Jud Early, "Adhesive Joining Update," in Proceedings of the 15th International Apparel Research Conference, 1988

4. D. R. Buchanan, "Directions of Technological Change in the Fiber, Textile, and Apparel Industries," in Automation and Robotics in the Textile and Apparel Industries, (Noyes Publications, 1986), pg 189.

5. B. K. Kahn, "Pressing Primer," from the Apparel Production Management Department, Fashion Institute of Technology.

6. Alan N. Hunter, "The Need for Automation" in Automation and Robotics in the Textile and Apparel Industries, (Noyes Publications, 1986), pg 162.

7. Julia C. Bonham, "American Society in the Age of Advanced Automation: Trends and Prospects," in Automation and Robotics in the Textile and Apparel Industries, (Noyes Publications, 1986), pg 275.

8. E. M. McPherson, "Corporate Planning Issues - Apparel Options," in Automation and Robotics in the Textile and Apparel Industries, (Noyes Publications, 1986), pg 231.

9. L. Michel, N. Williamson, G. Berkstresser, H. Kelly, "Textile and Apparel Technology in the United States," in Automation and Robotics in the Textile and Apparel Industries, (Noyes Publications, 1986), pg. 243.

10. Dennis A. Swyt, "The Technology and Workforce of Manufacturing and the Apparel Industry in Post - Industrial America," in Proceedings of the 15th International Apparel Research Conference, 1988

REFERENCES (continued)

11. T. G. Clapp, H. Peng, "Buckling of Woven Fabrics," Textile Research Journal, April 1990, Vol 60.

12. R. J. Bassett, R. Postle, "Fabric Mechanical and Physical Properties," International Journal of Cloth Science and Technology, December 4, 1989, Vol 2.

13. Carl Feynman, "Modelling the Appearance of Cloth," Master's Thesis, M.I.T., May 1986.

14. S. Kawabata. M. Niwa, K. Ito, "Use of Mechanical Properties to Predict Tailorability and Improved Garment Appearance," Proceedings of the 15th International Apparel Research Conference, 1988.

15. Nigel Cross, Engineering Design Methods (John Wiley and Sons, 1989), pp 60-70.

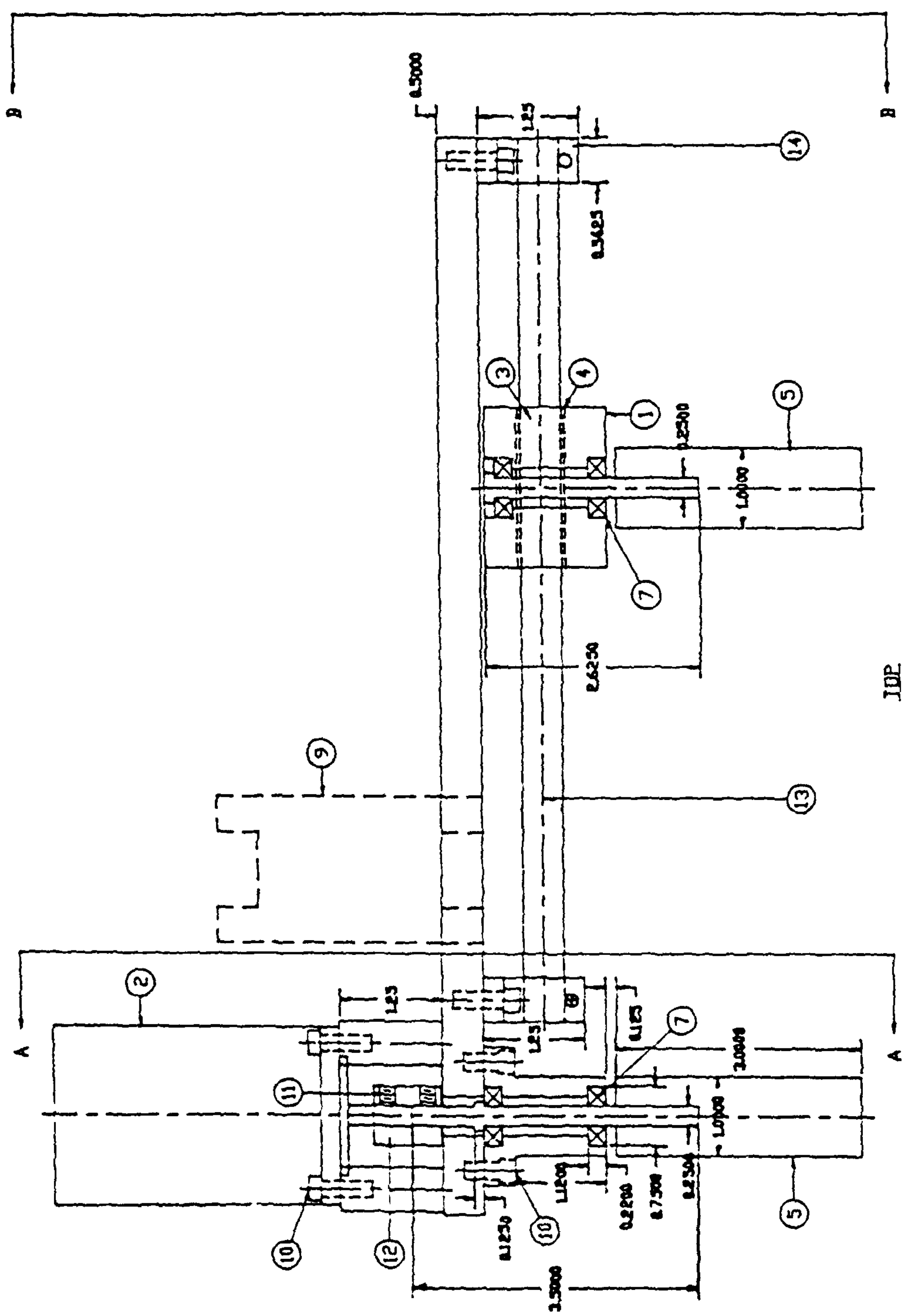
16. Henry S. Seesselberg, "Automated Handling of Limp Fabrics," in Proceedings of the 15th International Apparel Research Conference, 1988

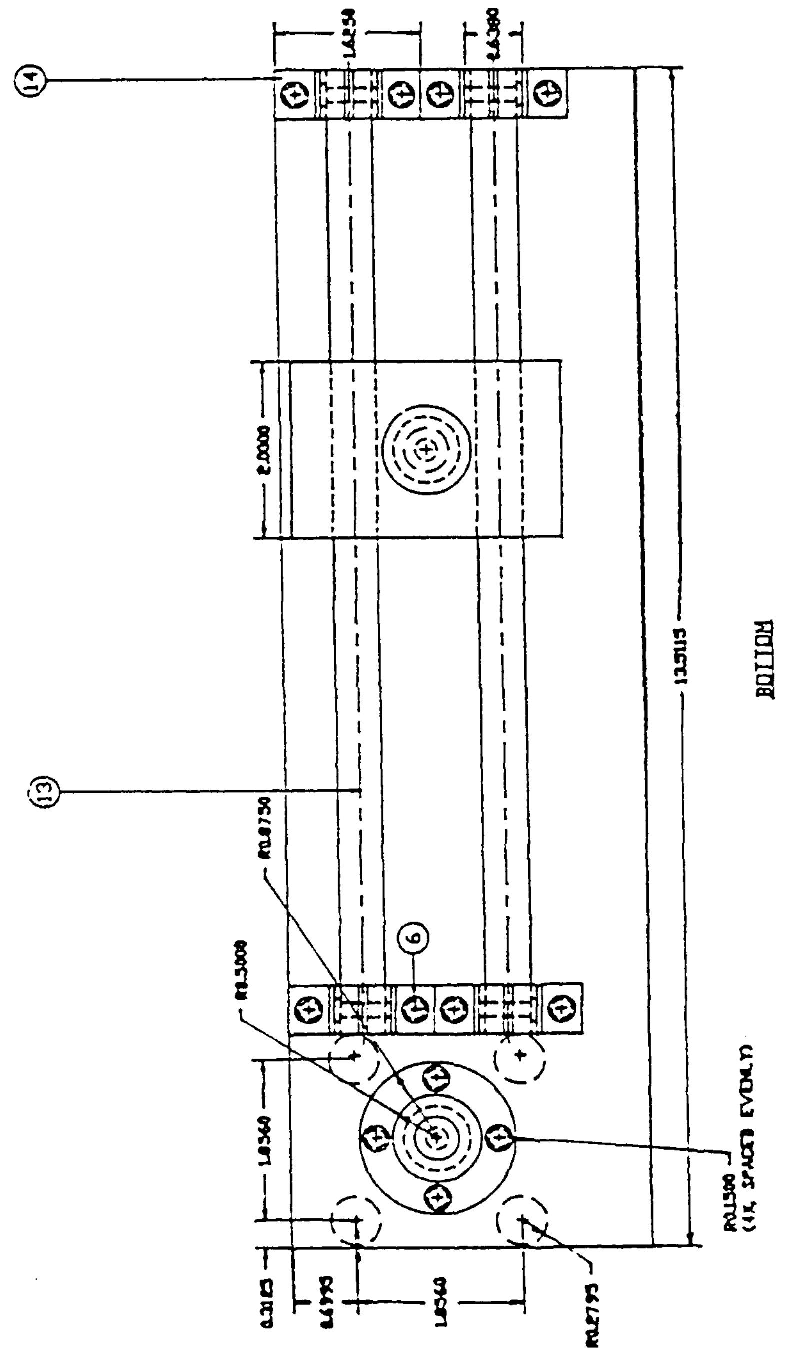
17. "Automated Systems and Robotic Handling: A Blueprint for Success Fabrics," in Proceedings of the 15th International Apparel Research Conference, 1988

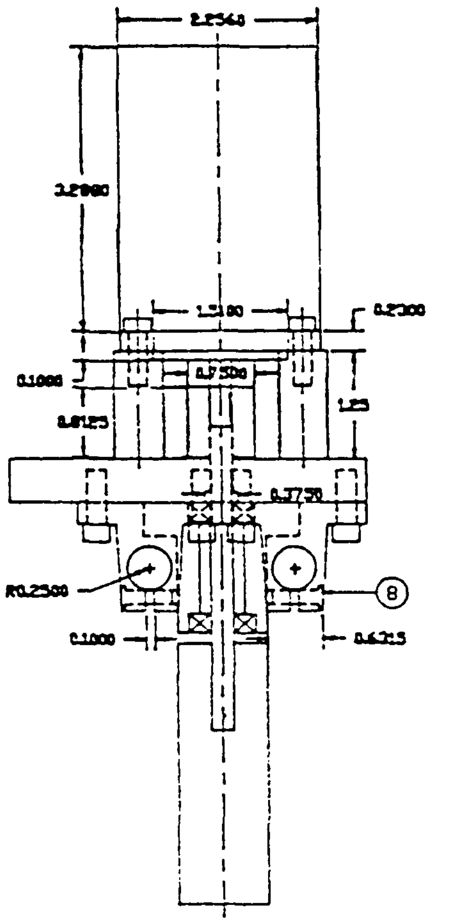
18. Edward Bernardon, "Utilizing Robots in Garment Assembly," in Proceedings of the 15th International Apparel Research Conference, 1988

19. M. Petry, W. Bick, "Automatic Insertion of Insulating Mats on Car Assembly Line," Assembly Automation, Vol 8, Number 4, 1988, pp 183-185.

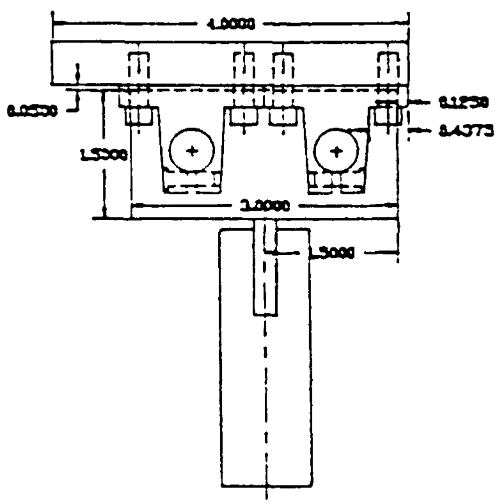
APPENDIX 1




14
13
6
2.0000
BOTTOM



SECTION A-A

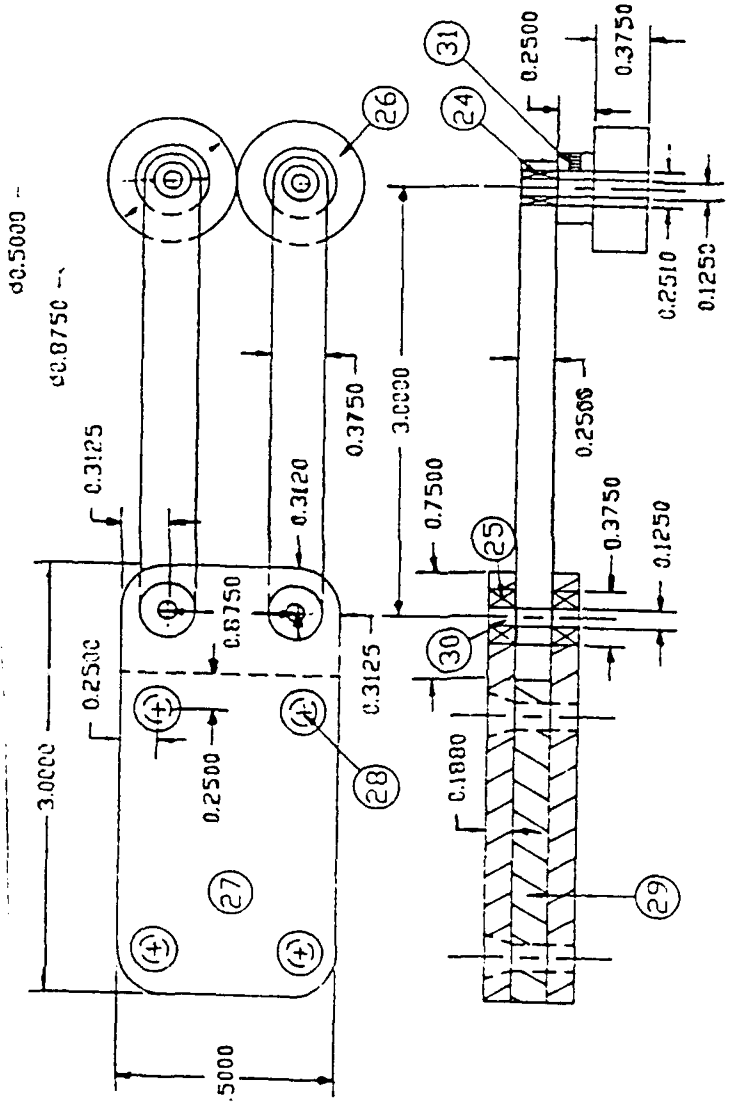


SECTION B-B

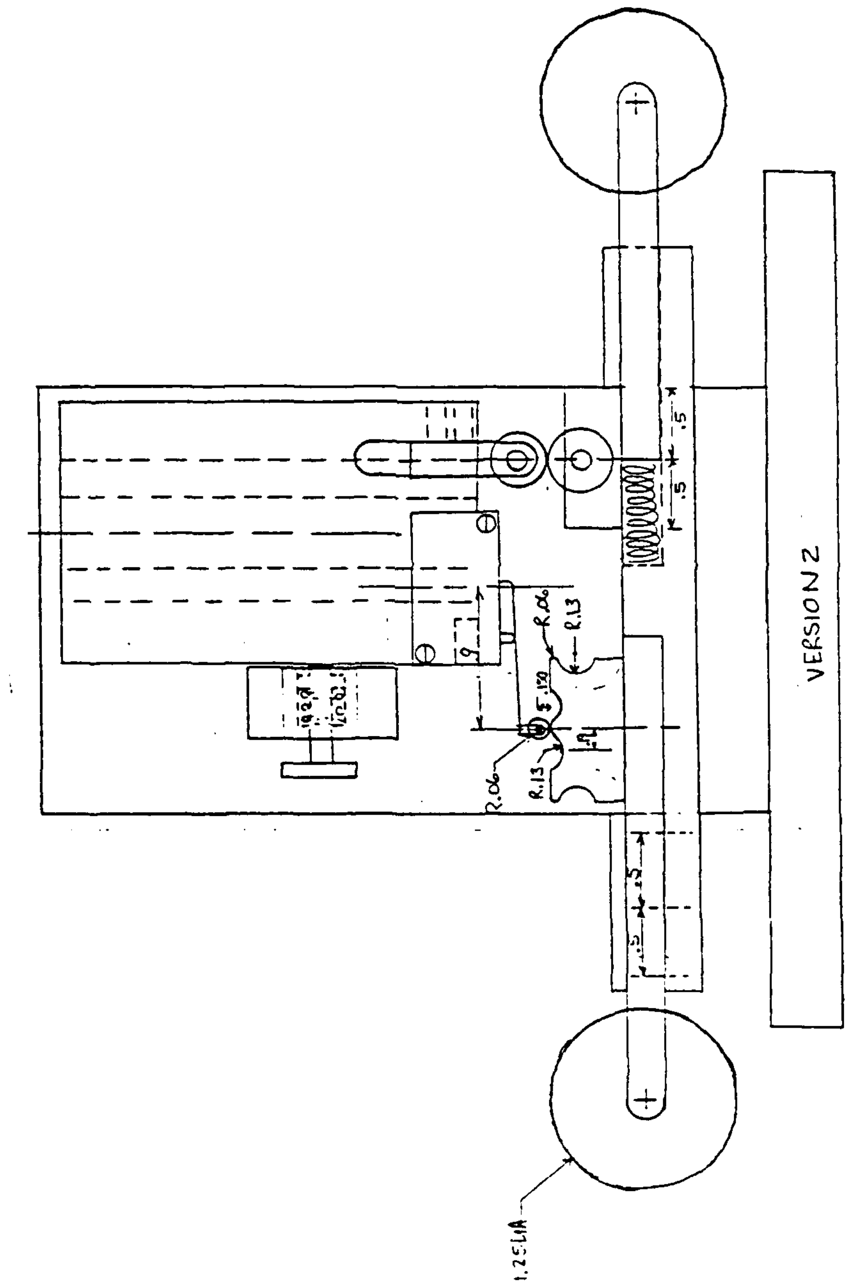
DETAIL	DESCRIPTION	QUANTITY
14	SHAFT SUPPORT BLOCK	4
13	GUIDE SHAFT 3/8 D X 12 L	2
12	MOTOR-SHAFT COUPLING	1
11	#6-32 UNC X 1/4 L SET SCREW	12
10	#10-24 UNC X 1/2 L SHCS	12
9	LOCATION OF AIR VALVE ASSEMBLY	1
	(SEE DWG # 3)	
8	#6-32 UNC X 1/2 L SHCS	4
7	BEARING, BALL 1/4ID X 3/4OD X .22W	4
6	#6-32 UNC X 1/2 L SHCS	8
5	ROLLER (COVERED BY RUBBER)	2
4	BALL BUSHING 3/8ID X 5/8OD X 7/8L	4
3	PUSH-IN RETAINING RING	4
2	STEPPER MOTOR	1
1	SLIDER BLOCK	1

DRAWN BY: TOBIAS HOPPE	DATE: 07/31/90	RPI Troy, N.Y. 12180
APPROVED BY:	DATE:	
SCALE: 0.45" = 1"	PART: ROLLER/SEAM ALIGNMENT DEVICE	
TOLERANCE UNLESS SPEC'D:	USED ON:	
	PROJECT: F.I.T.	PROJECT #:
MATERIAL UNLESS SPEC'D: 6061-T6 ALUM.	DWG #: 2 REV:	SHEET 1 OF 1

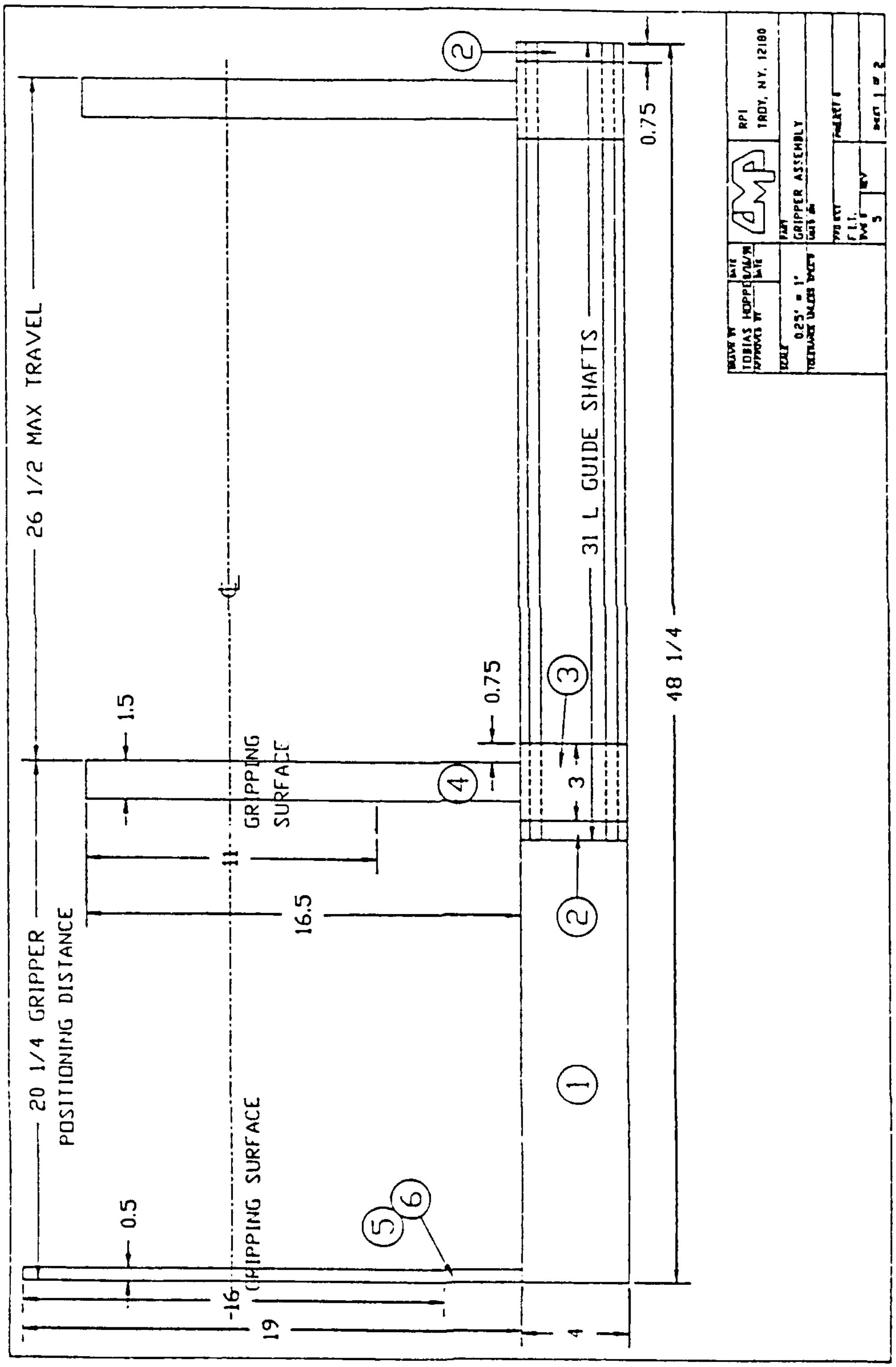
APPENDIX 2

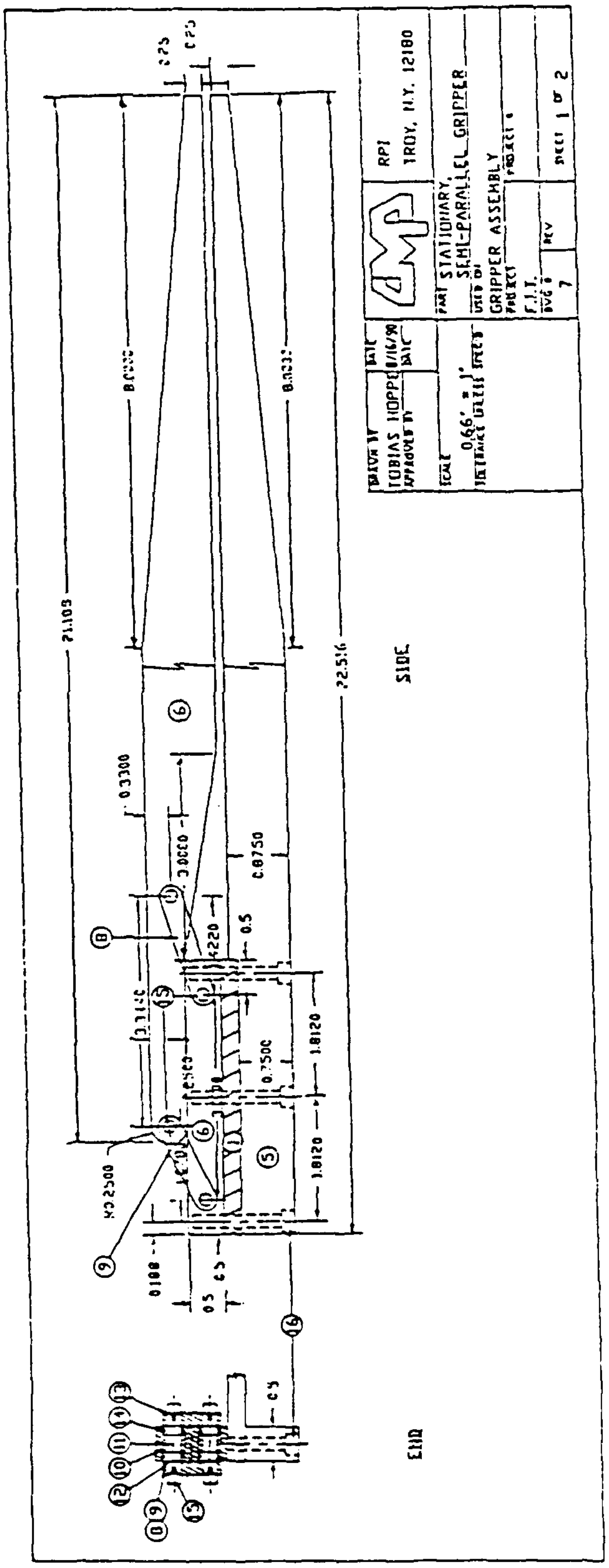


This Page Shall Not be Duplicated


VERSION 2
R.06
R.13
1.25 DIA

APPENDIX 4


26 1/2 MAX TRAVEL
31 L GUIDE SHAFTS
48 1/4
0.75
1.5
0.75
3
GRIPPING SURFACE
11
16.5
20 1/4 GRIPPER
POSITIONING DISTANCE
GRIPPING SURFACE
0.5
16
19
4
RPI
TROY, NY. 12180
GRIPPER ASSEMBLY
TOBIAS HOPPE
0.25" = 1'


RPI
TROY, N.Y. 12180
STATIONARY, SEMI-PARALLEL GRIPPER
GRIPPER ASSEMBLY
TOBIAS HOPPE
SIDE
END

DETAIL	DESCRIPTION	QTY
31	SET SCREW #6-32 UNC	2
30	PIN, STEEL 1/8D X 5/8L	2
29	ALUMINUM 1/4 X 1 1/2 X 2 1/4	1
28	SCREW #10-24 UNC X 3/4L FLAT HEAD	8
27	ALUMINUM 3/16 X 1 1/2 X 3	4
26	RUBBER ROLLER 1/4 ID X 7/8 OD X 3/8 FACE	2
25	BEARING 1/8 ID X 3/8 OD X 0.156 W	4
24	NEEDLE BEARING B-24	2
23	SPACER, ALUMINUM 1/2ID X 7/8OD X 1/8W	2
22	INTERNAL RET. RING 1/2ID X 7/8OD	4
21	LINEAR BEARING 1/2ID X 7/8OD X 1 1/4L	4
20	STEEL 1/2D X 31L	2
19	SCREW #6-32 UNC X 3/8 L SET	4
18	SCREW #10-24 UNC X 1/2L FLAT HEAD	9
17	#10-24 UNC X 2L SHCS	8
16	#10-24 UNC X 1 1/4L SHCS	3
15	#6-32 UNC X 1/4L SHCS	8
14	BALL BEARING 3/16ID X 3/8OD X 1/8W	8
13	WASHER, STEEL 0.167ID X 1/4OD X 0.020TH	8
12	SHIM, STEEL 3/16ID X 1/4OD X 0.010TH	16
11	SHAFT, STEEL 3/16D X 7/8L	4
10	SPACER, ALUMINUM 3/8OD X 3/16ID X 1/4W	4
9	SEMI-PARALLEL GRIPPER LINK	2
	1.188 C-C DISTANCE 3/8W X 3/16TH	
8	SEMI-PARALLEL GRIPPER LINK	2
	1.483 C-C DISTANCE 3/8W X 3/16TH	
7	ALUMINUM 1/2 X 1/2 X 4	1
6	STATIONARY, SEMI-PARALLEL GRIPPER TOP	1
	ALUMINUM 1/2 X 1 X 22	
5	STATIONARY, SEMI-PARALLEL GRIPPER BOTTOM	1
	ALUMINUM 1/2 X 1 X 23	
4	TRANSLATING CONTOURED SCISSORS GRIPPER	1
	ALUMINUM 3/16 X 1 1/2 X 21	
3	CONTOURED GRIPPER BEARING BLOCK	1
	ALUMINUM 2 X 3 X 4	
2	GUIDE SHAFT SUPPORT END BRACKETS	2
	ALUMINUM 3/4 X 4 X 2	
1	GRIPPER BASE, ALUMINUM 1/4 X 4 X 48	1

List of Theses Pending

The following is a list of master's and PhD theses that will result from the Automated Handling of Garments for Pressing project:

Master's Theses

Richard Heines, MechE - title to be determined, to be completed December 1991.

PhD Theses

Cliff Lansil, MechE - "The Automated Handling of Cloth Using Robots", to be completed June 1992.

Tiehua Cao, ECSE - "An Object-Oriented Environment for Robotic Workstation Planning", to be completed December 1992.

Dave Breen, ECSE - title to be determined, to be completed June 1992