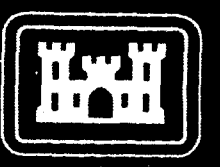
(2)

TEC-0022



US Army Corps
of Engineers
Topographic
Engineering Center

T

E

C

Knowledge-Based Vision Techniques for the Autonomous Land Vehicle Program, Final Report

Martin A. Fischler
Robert C. Bolles

SRI International
333 Ravenswood Avenue
Menlo Park, California 94025-3493

October 1991

Prepared for:
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209-2308

Monitored by:
U.S. Army Corps of Engineers
Topographic Engineering Center
Fort Belvoir, Virginia 22060-5546

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	October 1991	Final Technical Report Oct. 1990 - Sep. 1991

4. TITLE AND SUBTITLE

Knowledge-Based Vision Techniques for the Autonomous Land Vehicle Program, Final Report

5. FUNDING NUMBERS

DACA76-85-C-0004

6. AUTHOR(S)

Martin A. Fischler
Robert C. Bolles

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

SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025-3493

8. PERFORMING ORGANIZATION REPORT NUMBER

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

Defense Advanced Research Projects Agency
1400 Wilson Blvd., Arlington, VA 22209-2308

U.S. Army Topographic Engineering Center
Fort Belvoir, VA 22060-5546

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

TEC-0022

11. SUPPLEMENTARY NOTES

Effective 1 October 1991, the U.S. Army Engineer Topographic Laboratories (ETL) became the U.S. Army Topographic Engineering Center (TEC).

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The goal of this research was to develop techniques for automatically acquiring and representing knowledge about complex cultural and natural environments for such purposes as autonomous navigation, planning, intelligence analysis, and manipulation. Our research strategy was to (1) develop representations and techniques for storing (or incrementally learning) semantic and geographic information about a specific geographic area to permit both mission planning and knowledge-based interpretation of sensed data, (2) develop representations for natural and man-made objects, (3) develop techniques to recommend distinctive features of these objects that can be used for recognition purposes, and (4) develop techniques for building three-dimensional descriptions of an environment from data gathered by range or intensity sensors moving through this environment. In this report we describe our accomplishments in these areas. Near the end of the original contractual period a new task was added to evaluate the application of our cartographic stereo techniques to ground-level imagery. We characterized the strengths and weaknesses of the techniques, enhanced them based on the insights provided by the evaluation, and then evaluated the resulting systems. In this report we briefly describe the enhancements and our evaluation process.

14. SUBJECT TERMS

Computer vision, autonomous navigation, image understanding, unmanned ground vehicles

15. NUMBER OF PAGES

66

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

Contents

PREFACE

This research is sponsored by the Defense Advanced Research Projects Agency (DARPA), 1400 Wilson Boulevard, Arlington, Virginia 22209-2308 and monitored by the U.S. Army Topographic Engineering Center (TEC), Fort Belvoir, Virginia 22060-5546, under Contract DACA76-85-C-0004, by SRI International, 333 Ravenswood Avenue, Menlo Park, California 94025-3493. The Contracting Officer's Representative was Mr. George Lukes.

1 INTRODUCTION

SRI International is pleased to present this final report under contract DACA76-85-C-0004 to the Defense Advanced Research Projects Agency and the U.S. Army Engineer Topographic Laboratories. The goal of this research was to provide the technology to produce systems that can automatically model a complex physical environment (e.g., natural terrain) based on data from imaging sensors and previously stored knowledge. Requirements for such a system typically include the need for real-time interpretation and extreme reliability; e.g., if the perception system for an autonomous robot makes even one serious mistake in an extended mission, or cannot provide needed information fast enough, the robot could be disabled or destroyed.

The requirements for speed and reliability cannot be satisfied by simple extensions of existing technology. Stand-alone algorithms in which all needed knowledge is compiled into a sequence of instructions cannot be made robust enough to achieve the desired reliability goals; such algorithms must be replaced by more highly integrated systems with reasoning methods that can use significant amounts of knowledge, at least some of which is stored in declarative form. The need to operate in complex natural environments poses the corresponding requirement to develop new representations that go beyond the simple analytic descriptions adequate for the smooth surfaces characteristic of low-resolution aerial imagery, or of manufactured or cultural objects: correspondingly, serial computer architectures must be replaced by parallel hardware and corresponding algorithmic techniques to satisfy speed requirements.

We identified five technical areas in which advances were necessary to build a vision system that could satisfy required competence, speed, and reliability criteria in modeling complex natural environments.

- *Development of a Core Knowledge Structure*, which can serve as an integrating mechanism, and can store both the prior knowledge and the environmental models to be compiled from sensor data
- *Development of Compact Representations for Natural Scenes*, which can be used to render the described scenes realistically, can permit interactive or automatic three-dimensional model construction from symbolic data, and can be used by automatic recognition techniques in performing the sensor data interpretation task
- *Development of Terrain Modeling Techniques*, which can compile a description of natural terrain, from both range and intensity imaging sensors moving through some area of interest
- *Development of Object Recognition Techniques*, which can use stored descriptions to identify both cultural and natural objects in range and intensity data

- *Development of an Integrated Demonstration System*, which can be used to demonstrate and evaluate both the overall system concept and the component technology

In Sections 2 and 3 of this report we describe our accomplishments in these areas. The papers in the appendices cover some of our work in the above areas in more depth. In Section 4 we describe the evaluation and enhancement of our stereo techniques for application to ground-level data.

2 OBJECT RECOGNITION IN THE OUTDOOR WORLD

The natural outdoor environment poses significant obstacles to the design and successful integration of the interpretation, planning, navigational, and control functions of a general-purpose vision system. Many of these functions cannot yet be performed at a level of competence and reliability adequate to satisfy the needs of an autonomous robotic device. Part of the problem lies in the inability of current techniques, especially those involved in sensory interpretation, to use contextual information and stored knowledge in recognizing objects and environmental features. One of our goals in this effort was to design a core knowledge structure (CKS) that can support a new generation of knowledge-based generic vision systems. A second goal was to construct a vision system that employs the CKS, and has the competence to recognize objects appearing in ground-level imagery of natural outdoor scenes.

2.1 Core Knowledge System

The CKS is an object-oriented knowledge database that was originally designed to serve as the central information manager for a perceptual system [Smith&Strat87, Strat&Smith88]. The following facilities of the CKS are of particular importance in supporting the object recognition task.

Multiple Resolution in Space and Knowledge. The CKS employs a multiresolution octree to locate objects only as precisely as warranted by the data. Similarly, a collection of geometric modeling primitives are available to represent objects at an appropriate level of detail. In parallel with the octree for spatial resolution is a semantic network that represents objects at multiple levels of semantic resolution.

Inheritance and Inference. The CKS uses the semantic network to perform some limited types of inference that ease the burden of querying the data store. Thus, query responses are assembled not only from those objects that syntactically match the query, but also from objects that can be inferred to match, given the relations encoded in the semantic network. Spatial inference is provided based on geometric constraints computed by the octree manipulation routines.

Conflicting Data. One of the realities of analyzing imagery of the real world is that conflicts will result from mistakes in interpretation and from unnoticed changes in the world. The CKS treats all incoming data as the opinions of the data sources, so logical inconsistencies will not corrupt the database. Similarly, values derived through multiple inheritance paths are treated as multiple opinions. This strategy has several advantages and disadvantages. Rather than fusing information as it arises, the CKS has the option of postponing combination until its results are needed. This means that the fusion can be performed on the basis of additional information that

may become available, and in a manner that depends on the immediate task at hand. Some information may never be needed, in which case the CKS may forego its combination entirely. The disadvantages are the need to store a larger quantity of data and a slowed response at retrieval time. For an object recognition system like Condor (described below), the CKS seems to provide the right tradeoff.

2.2 Condor: A Contextual Vision System Built on the CKS

Much of the progress that has been made to date in machine vision has been based, almost exclusively, on shape comparison and classification employing locally measurable attributes of the imaged objects (e.g., color and texture). Natural objects viewed under realistic conditions do not have uniform shapes that can be matched against stored prototypes, and their local surface properties are too variable to be unique determiners of identity. The standard machine vision recognition paradigms fail to provide a means for reliably recognizing *any* of the object classes common to the natural outdoor world (e.g., trees, bushes, rocks, and rivers). In this effort [Strat&Fischler91, Appendix A], we have devised a new paradigm that explicitly invokes context and stored knowledge to control the complexity of the decision-making processes involved in correctly identifying natural objects and describing natural scenes.

The conceptual architecture of the system we describe, called Condor (for context-driven object recognition), is much like that of a production system; there are many computational processes interacting through a shared data structure. Interpretation of an image involves the following four process types.

- Candidate generation (hypothesis generation)
- Candidate comparison (hypothesis evaluation)
- Clique formation (grouping mutually consistent hypotheses)
- Clique selection (selection of a "best" description)

Each process acts as a daemon, watching over the knowledge base and invoking itself when its contextual requirements are satisfied. The input to the system is an image or set of images that may include intensity, range, color, or other data modalities. The primary output of the system is a labeled 3D model of the scene. The labels included in the output description denote object *classes* that the system has been tasked to recognize, plus others from the recognition vocabulary that happen to be found useful during the recognition process. An object *class* is a category of scene features such as sky, ground, geometric-horizon, etc.

A central component of the architecture is a special-purpose knowledge database used for storing and providing access to knowledge about the visual world, as well

as tentative conclusions derived during operation of the system. In Condor, these capabilities are provided by the CKS as previously discussed.

Visual interpretation knowledge is encoded in *context sets*, which serve as the uniform knowledge representation scheme used throughout the system. The invocation of all processing operations in Condor is governed by context through the use of various types of context sets: an action is initiated only when one or more of its controlling context sets is satisfied. Thus, the actual sequence of computations, and the labeling decisions that are made, are dictated by contextual information stored in the CKS, by the computational state of the system, and by the image data available for interpretation.

The customary approach to recognition in machine vision is to design an analysis technique that is competent in as many contexts as possible. In contrast to this tendency toward large, monolithic procedures, the strategy embodied in Condor is to make use of a large number of relatively simple procedures. Each procedure is competent only in some restricted context, but collectively, these procedures offer the potential to recognize a feature in a wide range of contexts. The key to making this strategy work is to use contextual information to predict which procedures are likely to yield desirable results, and which are not.

Condor operates as follows. For each label in the active recognition vocabulary, all candidate generation context sets are evaluated. The operators associated with those candidate generation context sets that are satisfied are executed, producing candidates for each class. Candidate comparison context sets that are satisfied are then used to evaluate each candidate for a given class, and if all such evaluators prefer one candidate over another, a preference ordering is established between them. These preference relations are assembled to form partial orders over the candidates, one partial order for each class. Next, a search for mutually coherent sets of candidates is conducted by incrementally building cliques of consistent candidates, beginning with empty cliques. A candidate is nominated for inclusion into a clique by choosing one of the candidates at the top of one of the partial orders. Consistency determination context sets that are satisfied are used to test the consistency of a nominee with candidates already in the clique. A consistent nominee is added to the clique; an inconsistent one is removed from further consideration with that clique. Further candidates are added to the cliques until none remain. Additional cliques are generated in a similar fashion as computational resources permit. Ultimately, one clique is selected as the best semantic labeling of the image on the basis of the portion of the image that is explained and the reliability of the operators that contributed to the clique.

We have taken over 100 photographs at an experimental site in the foothills behind Stanford University, most of which have so far been digitized and successfully processed by Condor. Based on our initial experiments, and the unique architecture of of our system, we are highly optimistic about the ability of Condor to overcome

many of the limitations with respect to object recognition inherent in traditional machine vision paradigms.

3 OBJECT MODELING FROM MULTIPLE IMAGES

Our goal in this research effort was to develop automated methods for producing a labeled three-dimensional scene model from image sequences. We view the image-sequence approach as an important way to avoid many of the problems that hamper conventional stereo techniques because it provides the machine with both redundant information and new information about the scene. The redundant information can be used to increase the precision of the data and filter out artifacts; the new information can be used for such things as filling in model information along occlusion boundaries and disambiguating matches in the midst of periodic structures.

We have developed two techniques for building three-dimensional descriptions from multiple images. One is a range-based technique that builds scene models from a sequence of range images; the second is a motion analysis technique that analyzes long sequences of intensity images. Our approach for analyzing sequences of range images is to provide the system with a wide variety of object and terrain representations and an ability to judge the appropriateness of these representations for particular sets of data. The variety of representations is required for two reasons. First, it is needed to cover the range of object types typically found in outdoor environments. And second, it is needed to cover the range of data resolutions obtained by a robot vehicle exploring the environment.

In this approach to object modeling, an object description typically goes through a sequence of representations as new data are gathered and processed. One of these sequences might start with a crude blob description of an initially detected object, include a detailed structural model derived from a set of high-resolution images, and end with a semantic label based on the object's description and the sensor system's task. This evolution in representations is guided by a structure we refer to as "representation space": a lattice of representations that is traversed as new information about an object becomes available. One of these representations is associated with an object only after it has been judged to be valid; we evaluate the validity of an object's description in terms of its temporal stability. We define stability in a statistical sense augmented with a set of explanations offering reasons for missing an object or having parameters change. These explanations can invoke many types of knowledge, including the physics of the sensor, the performance of the segmentation procedure, and the reliability of the matching technique. To illustrate the power of these ideas we have implemented a system, which we call TraX, that constructs and

refines models of outdoor objects detected in sequences of range data gathered by an unmanned ground vehicle driving cross-country [Bobick&Bolles91, Appendix B].

3.1 Building 3-D Descriptions from Image Sequences

We have developed a motion analysis technique, which we call Epipolar-Plane Image (EPI) Analysis [Bolles, et al 87]. It is based on considering a dense sequence of images as forming a solid block of data. Slices through this solid at appropriately chosen angles intermix time and spatial data in such a way as to simplify the partitioning problem. These slices have more explicit structure than the conventional images from which they were obtained. In the referenced paper we demonstrated the feasibility of this novel technique for building structured, three-dimensional descriptions of the world.

In later work, instead of analyzing slices, we extended the above technique to locate surfaces in the spatiotemporal solid of data, in order to maintain the spatial continuity of edges from one slice to the next [Baker&Bolles88]. This surface-building process is the three-dimensional analogue of two-dimensional contour analysis. We have applied it to a wide range of data types and tasks, including medical images such as computed axial tomography (CAT) and magnetic reasonance imaging (MRI) data, visualization of higher dimensional (i.e., greater than three-dimensional) functions, modeling of objects over scale, and assessment in fracture mechanics.

We have also implemented a version of EPI analysis that works incrementally, applying a Kalman filter to update the three-dimensional description of the world each time a new image is received [Baker&Bolles88]. As a result of these changes the program produces extended three-dimensional contours instead of sets of isolated points. These contours evolve over time. When a contour is initially detected, its location is only coarsely estimated. However, as it is tracked through several images, its shape typically changes into a smooth three-dimensional curve that accurately describes the corresponding feature in the world.

Recently we have further extended of the EPI analysis technique in two directions. The first is the modeling of biological structures from tomographic data [Baker90]. The descriptive formalism we are developing models tissue as two-dimensional manifolds in three-dimensional space. We have used this type of model to demonstrate simple versions of surgical simulation, kinematic modeling, and kinematic analysis. In the second extension we are using the temporal tracking mechanism in EPI analysis to detect and track moving objects from moving sensors. We have added evaluation routines that select key features to be tracked on the moving objects.

3.2 Detecting Moving Objects from Moving Sensors

Building upon our work in motion vision and terrain modeling, we have developed techniques for detecting and tracking moving objects from a moving platform.

Motion in a sequence of images provides one of the strongest cues available about the presence of a possible target in a scene. However, when a sensor is moving, everything in the image is moving. Therefore, detection of possible targets requires separating the motion created by the movement of the sensor from the motion caused by the movement of the target. One approach to this problem is to model the "background" image flow as a simple parametric flow field, then use this model to eliminate image motion consistent with that flow. Any motion not consistent with the background movement is labelled as a possible moving object. Of course, such an approach fails dramatically when the simple background assumption is violated, e.g., when the terrain contains many ridges and valleys, which generate a wide variety of background image motion.

The approach we have taken to handle these complex backgrounds is to integrate a full three-dimensional terrain map into the target detection system. The basic idea is to (i) use the model of the terrain and the known motion of the sensor to predict the motion observed by the sensor, (ii) compute the actual motion present in the imagery, and (iii) use the differences to robustly detect and track moving targets. The addition of a terrain model yields a significantly more robust and sensitive detection and tracking system than those relying on simpler background assumptions.

Note, however, that inaccuracies in the terrain model could produce differences between the predicted and computed image motion that, over a small number of images, look similar to moving objects. Therefore, integral to this approach is the ability to correct an a priori model of the environment as new data are acquired. In future work, we plan to draw upon current techniques for recovering structure from motion to dynamically update our models. By continually improving the underlying model, the motion detection procedure will be able to distinguish short-term deviations from moving objects.

As part of our research strategy we tested our algorithms on both simulated and real data, using the Cartographic Modeling Environment [Hanson&Quam88] to provide extensive simulation data. The advantage of simulated data is that we know "ground truth" and therefore are in a better position to judge the competence of the algorithms along some key dimensions than when we analyze real data. This strategy paid off. Our initial experimentation with simulated data pointed out a serious weakness in displaying warped images to demonstrate the results of optic flow computations. At occlusion boundaries, where flow vectors are undefined, optic flow techniques locate matches and compute flow vectors for points that have similar greyscale values. This procedure leads to stabilized intensity images, but completely bogus flow vectors. Thus, the results look better than they really are in these areas.

Given a terrain model, we are now able to predict occlusion boundaries and avoid these erroneous results.

4 SCENE MODELING FROM PASSIVE DATA

In previous contracts, our stereo research has concentrated on cartographic applications where the emphasis was on precision instead of speed and where the *variation* in sensor-to-world distances was relatively small. The goal of the stereo-evaluation task, which was added to this contract, was to evaluate approaches to fast, passive ranging techniques for high-oblique, ground-level operation. Our strategy was to characterize the strengths and weaknesses of current techniques, then investigate novel sensor configurations and processing techniques for achieving the goals of the passive ranging system.

To evaluate current and future stereo techniques we adopted an iterative strategy that contains four steps. The first step is to analyze the relationship between the geometric parameters of a stereo sensing system and the expected accuracy of the computed depth measurements. The basic relationships are well known, but they provide an initial set of constraints for designing a stereo system. The second step is to gather several different types of stereo data from a calibrated site. The third step is to generate some synthetic stereo sequences. The advantage of synthetic sequences is that they provide complete knowledge of the scene, including camera locations and the range to every point in the scene. The disadvantage is that it is not possible to generate completely realistic images. The fourth step is to apply the techniques to the data and evaluate their performance.

We selected a portion of the Stanford campus as the primary site for our data acquisition. It is a relatively flat area containing several well-spaced oak trees and a few eucalyptus trees. Since it does not include many small obstacles, we added a set of rocks and stumps of known dimensions. To gather real-time stereo sequences we mounted a pair of cameras on a truck, genlocked them to take their images simultaneously, and used a pair of videotape recorders to record the data. We used several sensor configurations, changing such parameters as the camera baseline, vergence angle, focal length, and aperture setting.

We concentrated our evalution on two stereo techniques developed here at SRI International. The first, called CYCLOPS, uses a global optimization technique (simulated annealing) to compute a range value for every point in an image. The second, called StereoSys, uses correlation patches to compute ranges at information-rich points in an image.

We began our evaluation by applying the techniques to an initial set of ground-level data. As expected, the techniques had trouble with such things as the large

range of disparities on the ground. To handle these large ranges, we modified our techniques so that they started their hierarchical matchers at a coarser level than for cartographic imagery. In addition, we implemented a technique for using the results from the analysis of one image pair in the sequence to initialize the analysis of the next pair. Both approaches were effective. However, more research is required to select the most effective control strategy for a particular Unmanned Ground Vehicle (UGV) task.

Since stereo analysis will be computation bound for the next few years, we began the exploration of focus-of-attention techniques to concentrate our analysis on the most important scene features. We implemented a "foveal" version of the CYCLOPS algorithm and started to explore control mechanisms for applying it. The idea was to use a terrain model to project the planned path of the vehicle into the imagery and then concentrate our analysis on that portion of the data.

In summary, the stereo techniques developed at SRI and at other research organizations around the world are mature enough to form the basis for an effective passive ranging system. In addition, hardware support for such techniques has progressed sufficiently to make them fast enough for specific applications. On the other hand, there are two key areas for future work. The first is in the characterization of the strengths and weaknesses of current techniques. This is required so that we know when to apply them to a task. The second area is the development of control strategies for applying these techniques to such demanding tasks as UGV perception.

5 CONCLUSIONS

We see the work described in this report as an important step in building a new generation of generic vision systems that are knowledge-base-driven, rather than task specific and designed around techniques in which domain knowledge is compiled into the algorithms. This new approach poses significant scientific problems that cannot be completely solved over a few years. However, we have made significant progress in four areas. First, we have developed a core knowledge system for integrating scene models from multiple sources, including both maps and sensory processing. Second, we have developed representation schemes for natural objects that support incremental modeling and recognition tasks. Third, we have developed a technique, called Epipolar-Plane Image Analysis, which builds structured three-dimensional models of a scene from image sequences. And fourth, we have developed a new technique that uses stored models and contextual information to recognize natural objects and man-made structures.

Our longer-range plans, which extend beyond the scope of this contract, are to increase the competence of the system so as to meet or exceed the terrain modeling and obstacle detection requirements for ground-level robotic devices, and to extend

the generality of the system so that it can be successfully applied to a wide range of vision problems.

6 BIBLIOGRAPHY

Baker, H.H., "Building Surfaces of Evolution: The Weaving Wall," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Baker, H.H. and R.C. Bolles, "Generalizing Epipolar-Plane Image Analysis on the Spatiotemporal Surface," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Baker, H.H., "Reimplementation of the Stanford Stereo System: Integration Experiments with the SRI Baseline Stereo System," Technical Note 431, Artificial Intelligence Center, SRI International, Menlo Park, CA, February 1989.

Baker, H.H., "Surface Modeling with Medical Imagery," *Proc. NATO Advance Workshop on 3D Imaging in Medicine*, Travemunde, Germany, June 1990.

Baker, H.H., "Scene Structure from a Moving Camera," in *AI and the Eye*, Edited by A. Blake and T. Troscianko, John Wiley & Sons, Ltd., 1990.

Barnard, S.T., "Stochastic Stereo Matching Over Scale," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Barnard, S.T., "Stochastic Stereo Matching on the Connection Machine," *Proc. of Fourth International Conference on Supercomputing*, Santa Clara, CA, 30 April - 5 May 1989.

Barnard, S.T., "Stochastic Stereo Matching Over Scale," *Int. J. of Computer Vision*, 3(1)1989.

Barnard, S.T., "Recent Progress in CYCLOPS: A System for Stereo Cartography," *Proc. DARPA Image Understanding Workshop*, Pittsburgh, PA, pp. 449-455, September 1990.

Barnard, S.T. and M.A. Fischler, "Computational and Biological Models of Stereo Vision," to appear in *Wiley Encyclopedia of AI*, 2nd ed; *Proc. DARPA Image Understanding Workshop*, Pittsburgh, PA, pp. 439-448, September 1990.

Bobick, A.F. and R.C. Bolles, " Representation Space: An Approach to the Integration of Visual Information," *Proc. of DARPA Image Understanding Workshop*, Palo Alto, CA, pp. 263-272, May 1989.

Bobick, A.F. and R.C. Bolles, "An Evolutionary Approach to Constructing Object Descriptions," *Proc. of 5th International Symposium of Robotics Research*, Tokyo, Japan, pp. 172-180, August 1989.

Bobick, A.F. and R.C. Bolles, "The Representation Space Paradigm of Concurrent Evolving Object Descriptions," to appear in *IEEE PAMI* in November 1991.

Bolles, R.C., H.H. Baker, and D.H. Marimont, "Epipolar-Plane Image Analysis: An Approach to Determining Structure from Motion," *International Journal of Computer Vision*, Kluwer Academic Publishers, Vol.I, No.1, pp. 7-5, June 1987.

Bolles, R.C. and A.F. Bobick, "Exploiting Temporal Coherence in Scene Analysis for Autonomous Navigation," *Proc. Robotics and Automation Conference*, Scottsdale, AZ, May 1989.

Fischler, M.A. and T.M. Strat, "Recognizing Objects in a Natural Environment: A Contextual Vision System (CVS)," *Proc. of Image Understanding Workshop*, Palo Alto, CA, pp. 774-796, May 1989.

Fischler, M.A., "An Overview of Computer Vision Research at SRI International - Themes and Progress," *International Journal of Computer Vision*, Vol. 3, No. 1, 1989.

Fua, P.V. and A.J. Hanson, "Extracting Generic Shapes Using Model-Driven Optimization," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Fua, P.V. and Y.G. Leclerc, "Model Driven Edge Detection," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Hannah, M.J., "Test Results from SRI's Stereo System," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Hannah, M.J., "A System for Digital Stereo Image Matching," *Photogrammetric Engineering and Remote Sensing*, Vol. 55, No. 12, pp. 1765-1770, December 1989.

Hanson, A.J. and L. Quam, "Overview of the SRI Cartographic Modeling Environment," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Leclerc, Y.G., "Constructing Simple Stable Descriptions for Image Partitioning," *International Journal of Computer Vision*, Vol. 3, No. 1, 1989a.

Leclerc, Y.G., "Segmentation via Minimal-Length Encoding on the Connection Machine," *Fourth International Conference on Supercomputing*, Santa Clara, CA, April 30–May 5, 1989b.

Langdon, J.H., J. Bruckner, and H. H. Baker, "Pedal Mechanics and Bipedalism in Early Hominids," in **Origines de la Bipedie chez les Hominides**, Editors Y. Coppens and B. Senat, Paris, France, June 1990.

Quam, L.H. and T.M. Strat, "SRI Image Understanding Research in Cartographic Feature Extraction," to appear in *Proc. of ISPRS Workshop*, Munich, Germany, September 1991.

Smith, G.B. and T.M. Strat, "Information Management in a Sensor-Based Autonomous System," *Proc. DARPA Image Understanding Workshop*, University of Southern CA, Vol.I, pp. 170-177, February 1987.

Strat, T.M. and G.B. Smith, "Core Knowledge System: Storage and Retrieval of Inconsistent Information," *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, April 1988.

Strat, T.M. and M.A. Fischler, "A Context-Based Recognition System For Natural Scenes and Complex Domains," *Proc. DARPA Image Understanding Workshop*, Pittsburgh, PA, 1990.

Strat, T.M. and M.A. Fischler, "Natural Object Recognition: A Theoretical Framework and Its Implementation," to appear in *Proc. of IJCAI 1991*, Sydney, Australia, August 1991.

Strat, T.M. and M.A. Fischler, "Context-Based Vision: Recognizing Objects using both 2D and 3D Imagery" to appear in IEEE PAMI, 1991.

Appendix A

Natural Object Recognition:
A Theoretical Framework and Its Implementation,
T.M. Strat and M.A. Fischler

to appear in IJCAI-91,
Sydney, Australia, August 1991.

Natural Object Recognition: A Theoretical Framework and Its Implementation

Thomas M. Strat and Martin A. Fischler*
Artificial Intelligence Center
SRI International
333 Ravenswood Avenue
Menlo Park, California 94025

Abstract

Most work in visual recognition by computer has focused on recognizing objects by their geometric shape, or by the presence or absence of some prespecified collection of locally measurable attributes (e.g., spectral reflectance, texture, or distinguished markings). On the other hand, most entities in the natural world defy compact description of their shapes, and have no characteristic features with discriminatory power. As a result, image-understanding research has achieved little success toward recognition in natural scenes. We offer a fundamentally new approach to visual recognition that avoids these limitations and has been used to recognize trees, bushes, grass, and trails in ground-level scenes of a natural environment.

1 Introduction

The key scientific question addressed by our research has been the design of a computer vision system that can approach human-level performance in the interpretation of natural scenes such as that shown in Figure 1. We offer a new paradigm for the design of computer vision systems that holds promise for achieving near-human competence, and report the experimental results of a system implementing that theory which demonstrates its recognition abilities in a natural domain of limited geographic extent. The purpose of this paper is to review the key ideas underlying our approach (discussed in detail in previous publications [12, 13]) and to focus on the results of an ongoing experimental evaluation of these ideas as embodied in an implemented system called Condor.

When examining the reasons why the traditional approaches to computer vision fail in the interpretation of ground-level scenes of the natural world, four fundamental problems become apparent:

Universal partitioning — Most scene-understanding systems begin with the segmentation of an image into homogeneous regions using a single partitioning algorithm applied to the entire image. If that partitioning is wrong, then the interpretation must also be wrong, no matter how a system assigns semantic labels to those regions. Unfortunately, universal partitioning algorithms are notoriously poor delineators of natural objects in ground-level scenes.

Shape — Many man-made artifacts can be recognized by matching a 3D geometric model with features extracted from an image [1, 2, 4, 6, 7, 9, 15], but most natural objects cannot be so recognized. Natural objects are assigned names on the basis of their setting, appearance, and context, rather than their possession of any particular shape.

Computational complexity — The object recognition problem is NP-hard [16]. As a result, computation time must increase exponentially as additional classes are added to the recognition vocabulary, unless a strategy to avoid the combinatoric behavior is

Figure 1: A natural outdoor scene of the experimentation site.

*Supported by the Defense Advanced Research Projects Agency under Contracts DACA76-85-C-0004, DACA76-90-C-0021, and 89F737300.

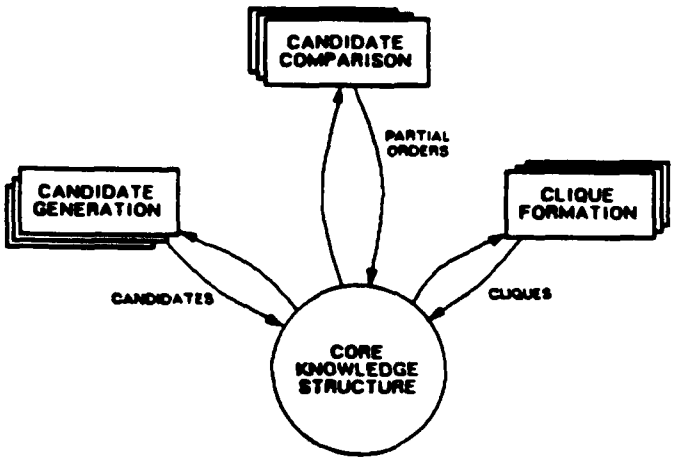


Figure 2: Conceptual architecture of Condor.

incorporated. Such provisions are a necessary component of any recognition system that can be scaled to embrace a real domain.

Contextual knowledge — Despite general agreement that recognition is an intelligent process requiring the application of stored knowledge [3, 5, 14], computer vision researchers typically use artificial intelligence techniques only at the highest levels of reasoning. The design of an approach that allows stored knowledge to control the lower levels of image processing has proved elusive.

Except for the continuing work at the University of Massachusetts [3], the understanding of natural scenes has received surprisingly little attention in the last decade.

2 Approach

A new paradigm for computer vision systems has been developed, which addresses all four of the problems described above. The key provision of this novel approach is a mechanism for the application of stored knowledge at all levels of visual processing. A *context set*, which explicitly specifies the conditions and assumptions necessary for successful invocation, is associated with every procedure employed by the recognition system.

The architecture is organized into three modules as depicted in Figure 2 and described below (a more complete description is also available [13]):

Candidate Generation —
Hypotheses concerning the presence in a scene of specific categories of objects are generated by delineating regions in an image using special-purpose operators whose invocation is controlled by context sets, thereby avoiding the need for universal partitioning algorithms. The employment of large numbers of operators ensures that quality hypotheses can be generated in nearly every context and provides redundancy that decreases the reliance on the success of any individual operator.

Candidate Comparison — Hypotheses are accepted only if they are consistent with all other members of a *clique* (consistent subset). Candidate hypotheses for each label are ranked so that the best candidates for each label can be considered before the others. Ranking the candidates ensures that the largest cliques can be found early in the search, thereby limiting the computational complexity of the entire paradigm to a linear growth as the recognition vocabulary is expanded. By constructing only a small number of cliques for each image, the approach loses any guarantee of finding the largest clique, but assures the availability of a credible answer compatible with the computational resources of the system.

Clique Formation — Consistency is enforced by procedures (controlled by context sets) that detect and reject physically impossible combinations of hypotheses. The clique that most completely explains the available data is offered as the interpretation of an image. Thus, individual objects are labeled on the basis of their role in the context of the complete clique, rather than solely on the basis of individual merit.

The invocation of all processing elements throughout the system is governed by context. All processing actions are controlled by context sets, and are invoked only when their context sets are satisfied. Thus, the actual sequence of computations (and the labeling decisions that are made) are influenced by contextual information, which is represented by prior knowledge about the environment and by the computational state of the system.

Definition: A *context set*, CS_k, is a collection of context elements that are sufficient for inferring some relation or carrying out some operation on an image.

Syntactically, a context set is embedded in a *context rule* denoted by

$$L : \{C_1, C_2, \cdots, C_n\} \Longrightarrow A$$

where L is the name of the class associated with the context set, A is an action to be performed, and the C_i comprise a set of conditions that define a context.

Example: The context rule

SKY : {SKY-IS-CLEAR, CAMERA-IS-HORIZONTAL RGB-IS-AVAILABLE } $\Longrightarrow$ BLUE-REGIONS

defines a set of conditions under which it is appropriate to use the operator BLUE-REGIONS to delineate candidate sky hypotheses.

There is a collection of context rules for every class in the recognition vocabulary, and each collection contains rules of three types: candidate generation, candidate comparison, and consistency determination. In theory, Condor performs the actions A that are associated with every satisfied context set.

Figure 3: Result of analyzing Figure 1.

3 The recognition process

For each label in the active recognition vocabulary, all candidate-generation context sets are evaluated. The operators associated with those that are satisfied are executed, producing candidates for each class. The candidate-comparison context sets that are satisfied are then used to evaluate each candidate for a class, and if all such evaluators prefer one candidate over another, a preference ordering is established between them. These preference relations are assembled to form partial orders over the candidates, one partial order for each class. Next, a search for mutually coherent sets of candidates is conducted by incrementally building cliques of consistent candidates, beginning with empty cliques. A candidate is nominated for inclusion into a clique by choosing one of the candidates at the top of one of the partial orders. Consistency-determination context sets that are satisfied are used to test the consistency of a nominee with candidates already in the clique. A consistent nominee is added to the clique; an inconsistent one is removed from further consideration with that clique. Further candidates are added to the clique until none remain. Additional cliques are generated in a similar fashion as computational resources permit. Ultimately, one clique is selected as the best semantic labeling of the image on the basis of the portion of the image that is explained and the reliability of the operators that contributed to the clique.

The interaction among context sets is significant. The addition of a candidate to a clique may provide context that could trigger a previously unsatisfied context set to generate new candidates or establish new preference orderings. For example, once one bush has been recognized, it is a good idea to look specifically for similar bushes in the image. This tactic is implemented by a candidate-generation context set that includes a context element that is satisfied only when a bush is in a clique.

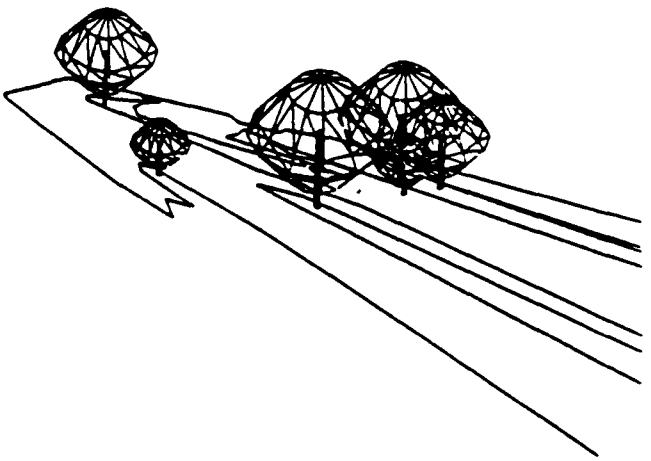


Figure 4: A perspective view of the 3D model produced from the analysis of the image shown in Figure 1.

4 Evaluation scenario

The approach has been implemented in the form of a complete end-to-end vision system, known as Condor. Images that are monochromatic or color, monocular or stereo, provide the input to the system, along with a terrain database containing prior knowledge about the environment. Condor produces a 3D model of the environment labeled with terms from its recognition vocabulary which is stored in the Core Knowledge Structure (CKS) [10, 11] and can be superimposed on the input image (Figure 3) or viewed from another perspective (Figure 4). The model is used to update the terrain database for use by Condor during the analysis of subsequent imagery.

To evaluate the Condor approach, we selected a two-square-mile region of foothills immediately south of the Stanford University campus as our site for experimentation. This area contains a mixture of oak forest and widely scattered oak trees distributed across an expanse of gently rolling, grass-covered hills and is criss-crossed by a network of trails.

We chose 14 classes for the recognition vocabulary on the basis of their prevalence in the experimentation site and their importance for navigation. These terms are:

{sky, ground, geometric-horizon, foliage, bush, tree-trunk, tree-crown, trail, skyline, raised-object, complete-sky, complete-ground, grass, tree}

Procedures have been devised to extract, evaluate, and check the consistency of candidates for each of these classes. Context sets have been constructed to control the invocation of each of those procedures. Currently the knowledge base contains 88 procedures whose invocation is governed by 156 context sets. All the results described in this paper have been generated using this knowledge base.

Initial contextual information was extracted from a USGS map and an aerial photograph; this includes a 30-meter-grid digital elevation model (DEM), the road network, and the location of forested regions as shown on the map. The aerial photo, being more recent, was used to update the map information. These data were extracted by hand and stored in the Core Knowledge Structure.

5 Experimentation

The research results presented here are indicative of the performance of Condor when analyzing scenes from the Stanford experimentation site. By themselves, these results do little to endorse the Condor approach, but together with similar results that have been obtained with several dozens of other images, they attest to the validity of the ideas contained therein.

5.1 Experiment 1

One shortcoming of many machine vision systems is their brittleness when analyzing scenes that exhibit significant variance in the setting or appearance of their components. Our design has focused on this problem because natural scenes possess great variability in their appearance. How well we have achieved this goal can be partially assessed by testing the following claim:

Assertion 1 *The Condor architecture is suitable for recognizing natural objects in many contexts.*

In this experiment, Condor analyzed images taken under a variety of conditions at the Stanford experimentation site. These images were selected to study how Condor deals with changes in scale, view angle, time of day, season, cloud cover, and other ordinary changes that occur over the course of several years. Here we present a sample of those images that illustrates the breadth of competence exhibited by Condor.

Figure 5 shows four images of the same tree obtained with the specified image acquisition parameters. In all four of these images, Condor successfully identified the tree without the benefit of any prior information. In three of the images, the trunk was identified by a specialized operator designed to detect thin, dark, vertical lines. In the fourth image, one of Condor's wide-trunk detection algorithms (a variant of a correlation-based road-tracking algorithm) was responsible for generating the correct trunk. Given that context, Condor used several of its texture measures to help identify the foliage and assembled the results into 3D models to confirm the existence of the tree. These results are indicative of Condor's abilities to recognize a tree from any view angle, to accommodate a 7:1 range in scale, to be immune from changes that occurred over a period of 21 months, and to deal with seasonal variation. When Condor has prior knowledge of the existence of this tree, it can be recognized from a distance of at least 590 feet (as demonstrated in Experiment 3), thereby extending its abilities to a 20:1 range in scale.

Experiments applying Condor to other images (not reproduced here) confirm the viability of the approach for recognizing natural objects in a wide variety of settings

range:	194 feet	28 feet
view angle:	160°	124°
date:	12 April 1990	28 July 1988
range:	56 feet	87 feet
view angle:	208°	258°
date:	12 April 1990	12 April 1990

Figure 5: The models of the trees as they were recognized by Condor.

that occur at the experimentation site. The modularity of the context sets makes it possible to expand the breadth of competence still further without degrading previously demonstrated capabilities.

5.2 Experiment 2

To support autonomy in an intelligent, ground-based vehicle, it is necessary to synthesize a reasonably complete description of the entire surroundings, and not just recognize a few isolated objects. This description can be built incrementally because the world does not change very rapidly considering the spatial and temporal scales at which an autonomous ground vehicle would operate. The following assertion summarizes this notion:

Assertion 2 *A geographic database of an extended region can be constructed by combining the recognition results from multiple images, taken over an extended period of time and under multiple viewing conditions.*

To validate this assertion, a sequence of imagery was collected which simulates the movement of a vehicle through a portion of the Stanford experimentation site. The vision system is to construct a labeled, 3D map of the primary features in the vicinity of the simulated vehicle by analyzing each image in turn.

Figure 6 shows the location of the vehicle when each image in the sequence was acquired. Condor was tasked to locate the trees, bushes, trails, and grass in each of these images, beginning with only the information extracted from the USGS map. The results of Condor's

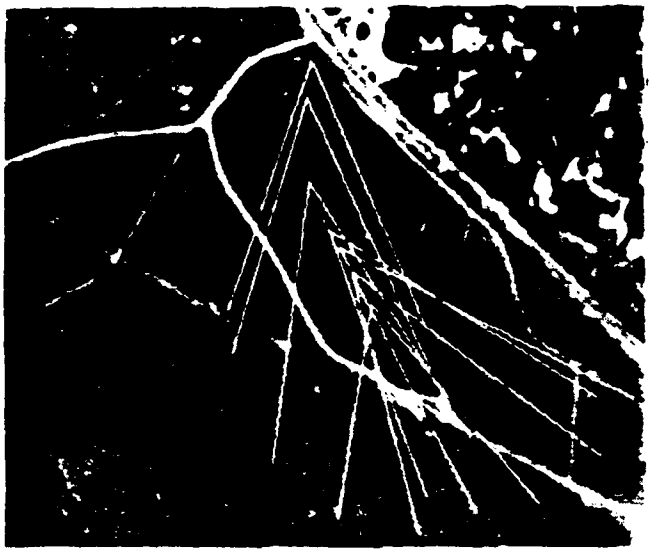


Figure 6: The location and orientation of the camera when each image in Figure 7 was acquired.

analysis are portrayed in Figure 7. Here we highlight a few of the more interesting chains of reasoning and explain the misidentifications that were made:

Image 1 — Condor has correctly labeled the sky, the ground, the trail, and part of the grass, although the trees on the horizon were too indistinct to be reliably identified. These results are transformed into three-dimensional models and positioned in 3-space using depth data acquired from binocular stereo.[1] The resulting models were added to the CKS database to be used as context for the analysis of subsequent images.

Image 2 — The model of the trail from the first image was projected into the second image and used to help identify a portion of the trail. This is accomplished by an operator that superimposes a pair of parallel 3D curves and deforms them to find the model with maximum edge strength while minimizing its curvature (as in [8]). Statistics from the intensity and texture of the grass in the first image were used to help identify the grass in this second image. In this case, the trail-finding operators failed to find the upper half of the trail; as a result, the grass hypotheses in that area were not contradicted.

Image 3 — The tree is finally close enough to allow reliable recognition and a 3D model for it is computed by extracting the envelope of its foliage. The entire visible portion of the trail was correctly identified.

Image 4 — Two additional trees are recognized and stored.

Image 5 — The same trees are recognized by predicting their location and verifying their existence — a much more reliable process than initially extracting

[1] When range data are not available, Condor estimates the depths by projecting each region onto the DEM.

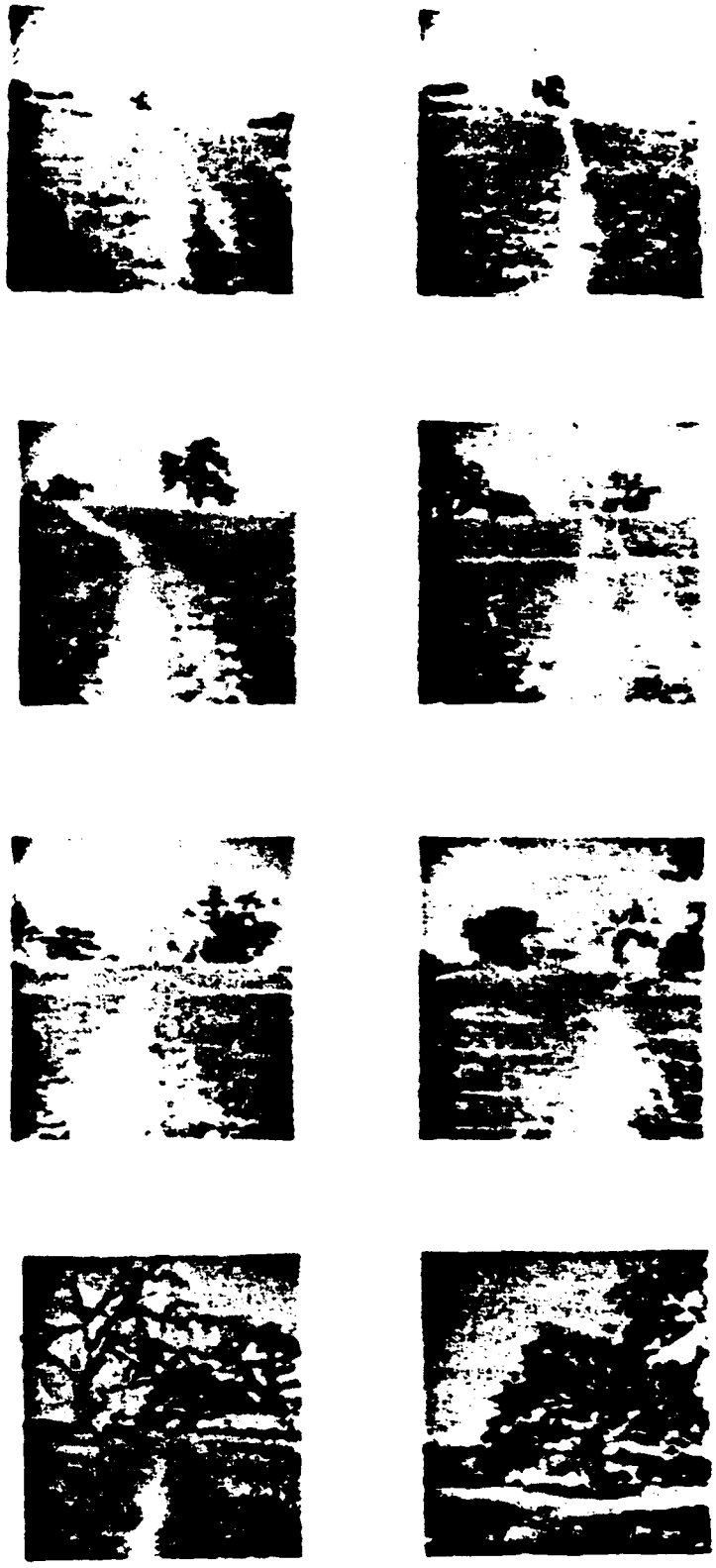


Figure 7: Results of Condor's analysis of the sequence of eight images.

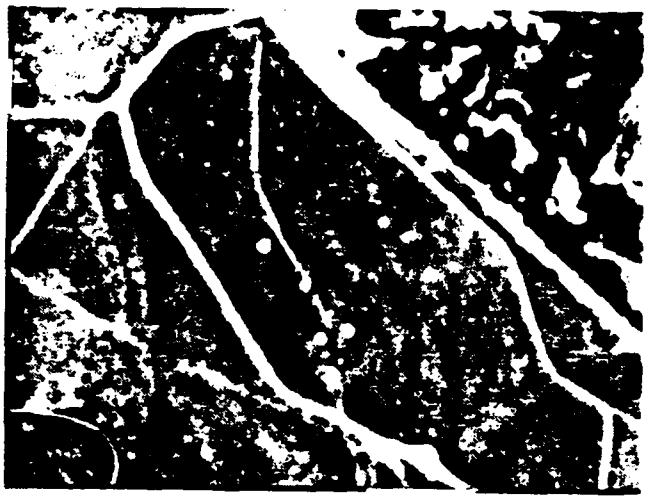


Figure 8: The composite model resulting from the analysis of the image sequence in Figure 7.

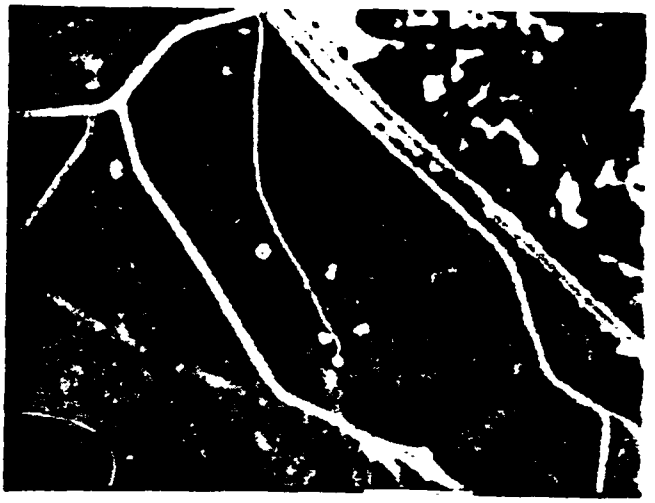


Figure 9: The ground-truth database

them. No trunk was detectable in the foliage to the left of the image, so Condor labeled it as bush

Image 6 — The texture in the lower corners of the sixth image was found to more closely resemble foliage than grass, so these regions were erroneously identified as bushes. Beause they are very near the camera, they occupy a significant part of the image, but the 3D model created for them reveals that they are less than 2 feet tall.

Image 7 — Several more trees, grass areas, and part of the trail are recognized in the seventh image

Image 8 — The primary tree is recognized despite the strong shadows, but the lower portion of the trunk was missed by all the trunk operators. Most of the tree crown operators were unable to provide a decent candidate because of the overhanging branches in the upper-right corner — the only operator that succeeded was the one that predicts the crown based on the size and location of the trunk. The combined effects of the incomplete trunk, the nearness of the tree, and the lack of range data account for poor extraction of the tree crown.

This experiment illustrates how Condor is able to use the results of analyzing one image to assist the analysis of other images. Although some trees and parts of the trail were missed in several images, the 3D model that results is nearly complete. Figure 8 shows an aerial view of the composite model contained in the CKS after processing all eight images. For comparison, Figure 9 portrays a model of the objects actually present on the ground, which was constructed by physically measuring the locations and sizes of the individual objects. Note that all of the trees that were visible in at least one image have been correctly labeled, although some of them were misplaced. Most of the trail has been detected, enough to allow a spatial reasoning process to link the portions into a single continuous trail. Furthermore, everything that was labeled *tree* actually is a tree

5.3 Experiment 3

Regardless of the architecture, knowledge-based vision systems are difficult to build. If the programmer needed to specify in advance all the information necessary for successful recognition, his task would be hopeless. Therefore, it is essential that a vision system have the ability to improve its competence autonomously, thereby learning through experience how to recognize the objects in its environment

Assertion 3 *Using context allows Condor to learn how to recognize natural objects*

To test the validity of this assertion, we return to the first image of the sequence used in Experiment 2 (Figure 7). When originally analyzed, Condor recognized the trail and part of the grass, but not the trees.

Condor was tasked to reanalyze the first image, this time making use of the contents of the entire database constructed during the analysis of the sequence of eight images. The resulting interpretation is depicted in Figure 10.

Two trees that could not be extracted on the first pass are now identified. Condor employed a tree-trunk operator whose context set requires knowledge of the approximate location of a tree in the field of view. The operator projects a deformable 3D model of the trunk onto the image, and optimizes its fit to extract the trunk. This operator successfully identified two of the trees without contradicting any of the original recognition results.

This experiment (along with others not described here) illustrates that the ability to use prior recognition results as context while interpreting subsequent images enables Condor to improve its performance as its exposure to its environment increases

Figure 10: The results of analyzing the first image from Figure 7 with and without the information extracted from subsequent images.

6 Conclusion

In its present embodiment, Condor is still a demonstration system that should be evaluated primarily in terms of its architectural design and innovative mechanisms, rather than its absolute performance. While Condor has demonstrated a recognition ability approaching human-level performance on some natural scenes, it is still performing at a level considerably short of its ultimate potential (even for the Stanford experimentation site). The knowledge acquisition mechanisms, which are a key aspect of the architecture, should allow continued improvement in performance with exposure to additional site imagery.

A new paradigm for image understanding has been proposed, and used to recognize natural features in ground-level scenes of a geographically limited environment. This context-based approach is exciting because it deemphasizes the role of image partitioning and emphasizes the recognition context in a way that has not been attempted before. This new focus could lead to the construction of vision systems that are significantly more capable than those available today.

7 Acknowledgment

Condor includes software provided by many present and former members of the Perception Group at SRI. In addition, Marty Tenenbaum, Jean-Claude Latombe, and Lynn Quam have contributed to the research reported here.

References

[1] Bolles, R.C., R. Horaud, and M.J. Hannah, "3DPO: A 3D Part Orientation System," in *Proceedings 8th International Joint Conference on Artificial Intelligence*, Karlsruhe, West Germany, pp. 1116–1120 (August 1983).

[2] Brooks, Rodney A., "Model-Based 3-D Interpretations of 2-D Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 5, No. 2, pp. 140–150 (March 1983).

[3] Draper, Bruce A., Robert T. Collins, John Brolio, Allen R. Hanson, and Edward M. Riseman, "The Schema System," *International Journal of Computer Vision*, Vol. 2, No. 3, pp. 209–250 (January 1989).

[4] Faugeras, O.D., and M. Hebert, "A 3-D Recognition and Positioning Algorithm using Geometrical Matching Between Primitive Surfaces," *Proceedings 8th International Joint Conference on Artificial Intelligence*, Karlsruhe, West Germany, pp. 996–1002 (August 1983).

[5] Garvey, Thomas D., "Perceptual Strategies for Purposive Vision," Ph.D. Dissertation, Department of Electrical Engineering, Stanford University, Stanford, CA (December 1975).

[6] Grimson, W.E.L., and T. Lozano-Perez, "Model-Based Recognition from Sparse Range or Tactile Data," *International Journal of Robotics Research*, Vol. 3, No. 3, pp. 3–35 (1984).

[7] Huttenlocher, Daniel P., and Shimon Ulman, "Recognizing Solid Objects by Alignment," *Proceedings: DARPA Image Understanding Workshop*, Cambridge, MA, pp. 1114–1122 (April 1988).

[8] Kass, Michael, Andrew Witkin, and Demitri Terzopoulos, "Snakes: Active Contour Models," *Proceedings, ICCV*, London, England, pp. 259–268 (June 1987).

[9] Ponce, Jean, and David J Kriegman, "On Recognizing and Positioning Curved 3D Objects from Image Contours," *Proceedings: DARPA Image Understanding Workshop*, Palo Alto, CA, pp. 461–470 (May 1989).

[10] Smith, Grahame B., and Thomas M. Strat, "A Knowledge-Based Architecture for Organizing Sensory Data," *International Autonomous Systems Congress Proceedings*, Amsterdam, Netherlands (December 1986).

[11] Strat, Thomas M., and Grahame B. Smith, "The Core Knowledge System," Technical Note 426, Artificial Intelligence Center, SRI International, Menlo Park, CA, (October 1987).

[12] Strat, Thomas M., and Martin A. Fischler, "A Context-Based Recognition System for Natural Scenes and Complex Domains," *Proceedings, DARPA Image Understanding Workshop*, Pittsburgh, PA, pp. 456–472 (September 1990).

[13] Strat, Thomas M., "Natural Object Recognition," Ph.D. Dissertation, Department of Computer Science, Stanford University, Stanford, CA (December 1990).

[14] Tenenbaum, Jay M., "On Locating Objects by Their Distinguishing Features in Multisensory Images," *Computer Graphics and Image Processing*, pp. 308–320 (December 1973).

[15] Thompson, D.W., and J.L. Mundy, "Three-Dimensional Model Matching from an Unconstrained Viewpoint," in *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 208–220, 1987.

[16] Tsotsos, John K., "A Complexity Level Analysis of Immediate Vision," *International Journal of Computer Vision*, Vol. 1, No. 4, pp. 303–320 (1988).

Appendix B

The Representation Space Paradigm of Concurrent Evolving Object Descriptions,
A.F. Bobick and R.C. Bolles

to appear in IEEE PAMI,
November 1991.

The Representation Space Paradigm of Concurrent Evolving Object Descriptions

Aaron F. Bobick Robert C. Bolles

Artificial Intelligence Center
SRI International
333 Ravenswood Ave.
Menlo Park, CA 94025

Abstract: A representation paradigm for instantiating and refining multiple, concurrent descriptions of an object from a sequence of imagery is presented. This paradigm is designed to be used by the perception system of an autonomous robot that (1) needs to adequately describe many types of objects, (2) initially detects objects at a distance and gradually acquires higher-resolution data, and (3) is continuously collecting sensory input. We argue that multiple, concurrent descriptions of an object are necessary because different perceptual tasks are best performed using different representations and because different levels of description require different quality of data to support their computation. Since the data changes significantly over time, the paradigm supports the evolution of descriptions, progressing from crude two-dimensional "blob" descriptions to complete semantic models, such as bush, rock, and tree. To control this accumulation of new descriptions, we introduce the idea of a *representation space*. The representation space is a lattice of representations that specifies the order in which they should be considered for an object. One of the representations in the lattice is associated with an object only after the object has been described multiple times in the representation and the parameters of the representation have been judged to be "stable." We define stability in a statistical sense enhanced by a set of explanations describing valid reasons for deviations from expected measurements. These explanations may draw upon many types of knowledge, including the physics of the sensor, the performance of the segmentation procedure, and the reliability of the matching technique. To illustrate the power of these ideas we have implemented a system, which we call TraX, that constructs and refines models of outdoor objects detected in sequences of range data.

1 Introduction

Much of computer vision research is directed at the problem of constructing computational descriptions of the world. To that end, many representations — description languages — have been devised to describe different types of objects and support different types of tasks (e.g., see Agin & Binford, 1973; Marr & Nishihara, 1978; and Oshima & Shirai, 1978). In addition, there is an extensive body of research on filtering techniques for incrementally refining object description parameters as new sensory data are acquired. However, little research has been devoted to the coordination of multiple, concurrent descriptions of objects, particularly when the descriptions are to be refined over time. In this paper we present a representation paradigm that supports the instantiation, accumulation, and refinement of significantly different descriptions of an object.

The goal of constructing a multiplicity of descriptions of an object is motivated by the following two observations: First, different objects and different tasks require different representations. A description language well suited for describing the shape of vegetation may be poorly suited to describing the shape of a hippopotamus. Second, as the quality of sensory data changes, the types of representations that can be supported change. The initial description of a distant object may be as simple as a bounding sphere, while a fully developed model, built from high resolution data, may be a complex structure of parts. It is premature to try to compute a multi-part description of an object that spans only a few pixels in an image.

The motivation for our research is the development of a perception system for an autonomous robot. One of our primary goals for such a system is for it to construct a reliable model of the environment that is complete enough to support such tasks as route planning, obstacle detection, and landmark recognition. This need to support a wide range of tasks requires the perception system to compute a rich set of descriptions. Also, within the domain of autonomous navigation the availability of new and improved data arises naturally: approaching an object yields better resolution and repeated observations from different directions provides increasing shape information. To provide an intuition as to the desired performance of such a perception system, and to motivate the use of multiple, concurrent descriptions, consider the following example of an autonomous system constructing a map of its environment as it moves along:

Assume that a robot vehicle using range imagery initially detects a small object at a distance of 20 meters (the obstacle is actually a thin thistle bush). At that range, the system cannot be certain whether the object is a real obstacle or an artifact of the detection process; confirmation from the analysis of subsequent images is required. By analyzing 3 or 4 new images of the scene, the program determines that the object is real, and then formally enters the object into the robot's model of the environment. Poor sensor resolution, however,

permits only a crude estimate of the object's size and position. As the vehicle continues to approach the object, the increased resolution allows the robot to specify the size and position more precisely; again, agreement between estimates from one image to the next provides a high degree of confidence in these estimates. As the vehicle gets closer yet, the program detects and builds descriptions of individual parts of the object. It detects and describes four stick-like parts that correspond to the stem and branches of the thistle. When these parts have been confirmed over several images, they are added to the object's model and refinement procedures are instantiated to update their shape and location estimates over time. And finally, since the descriptions of the object's parts match those of thistle bushes, which are expected in the area, the robot classifies the object as a thistle bush, and adds this semantic description to the object's model. This cumulative description process is shown in Figure 1.

If during the analysis that produces these descriptions, the bush is not detected in an image, the program tries to explain why the bush was not detected instead of assuming that it disappeared. Perhaps the bush is out of the sensor's field of view, is occluded by another object, or was missed by the low-level segmentation process. Incorporating such an explanation subsystem into the description evaluation process extends our definition of temporal stability to include such events and improves performance by successfully accounting for rare, yet expected, situations. To identify the possible causes of loss of continuity, the explanation subsystem invokes a wide variety of heuristics designed to match the characteristics of particular sensory and processing stages.

As the perception system computes and validates more descriptions of the objects in a scene, it is able to provide better responses to requests from other vehicle modules, such as the route planner or the landmark recognizer. For example, if the perception module has only computed and validated the crude 3D description of an object, its response to a question about possible obstacles in front of the vehicle would only consist of a description of the object's approximate size and location. Not knowing the identity of the object, the planner would have to select a route that avoids the object. If, on the other hand, the perception module has identified the object as a thistle, the planner has more options, including running over the object, if there is no convenient clear path around it.

In this representation paradigm, the system only compares two object descriptions in the context of a specific task. The key question is "Which description is better for answering the particular question?" not "Which description is intrinsicly better?" Thus, an occupancy grid may be the best description for answering questions about the empty space around an object; a viewer-centered description may be best for tracking an object from image to image; a generalized-cylinder model may be the best for predicting the appearance of an object from another point of view. The system employs several representations as equal partners in its description of the object.

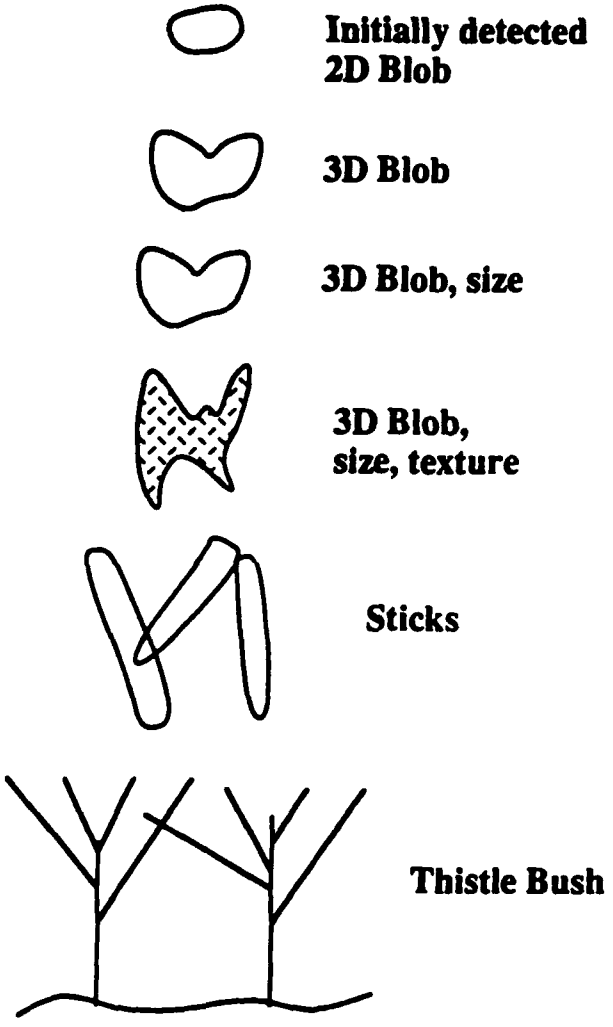


Figure 1: The evolution of the description of an object. As more information becomes available, the parameters of previously instantiated descriptions are refined and, as more descriptions can be computed reliably, the model of the object is expanded to include these new descriptions.

In the remainder of this paper we describe a representational framework for a vision system that maintains multiple, concurrent descriptions of objects. The representations used to form the descriptions are designed to model different types of objects, to support different types of inferences, and to require different specificity and accuracy of data to warrant their computation. We embed this framework within a system that incrementally constructs object descriptions over time, such that the complete description of an object evolves. We make use of temporal stability to assess the validity of computed descriptions. In the final section of the paper, we discuss some the difficult questions that are raised by employing such a representational scheme.

Throughout this paper, we illustrate our ideas with results from the TraX system, an implemented system that constructs and refines models of outdoor objects, such as bushes, trees, and rocks, detected in sequences of range data. With regards to the implementation of TraX we make two observations: First, we do not mean to imply that the particular set of representations presented is adequate to describe the entire outdoor world. In fact, our research is designed to allow the seamless introduction of additional representations as is necessary; additional representations are needed as new object types or new tasks are considered. Second, the particular components we assemble for addressing the autonomous navigation task are not of primary importance; they were constrained by the sensory data available and the objects of interest. What we do hope to convey is the importance of incorporating a detailed understanding of the sensors and the processing algorithms into the multi-representational framework; this understanding is critical to successfully choosing available representations and exploiting computed descriptions to perform necessary tasks.

2 A Space of Representations

In a multiple representation system, should all the representations be used to describe all the known objects all of the time? We argue that the answer to this question is "no" for two reasons: First, the resolution of the data may only support simple models. Not only would computing a more complex structural description be a waste of computational resources, the model produced would be erroneous, possibly leading to false conclusions on the part of the perception system. Second, the diversity of objects in the world is such that some objects are best described using one set of representations whereas others are best characterized by another. It is unreasonable to expect a single representation to be appropriate for all objects in the outdoor world; this is especially true for high-level representations such as generalized cylinders [Agin and Binford, 1973], superquadrics [Pentland and Bolles, 1988], or geometric solids [Popplestone, etal. 1975; Oshima and Shirai, 1978]. In the domain of autonomous navigation, a building might be well represented by geometric solids while more irregularly shaped objects, such as trees and bushes, would require quite different representations.

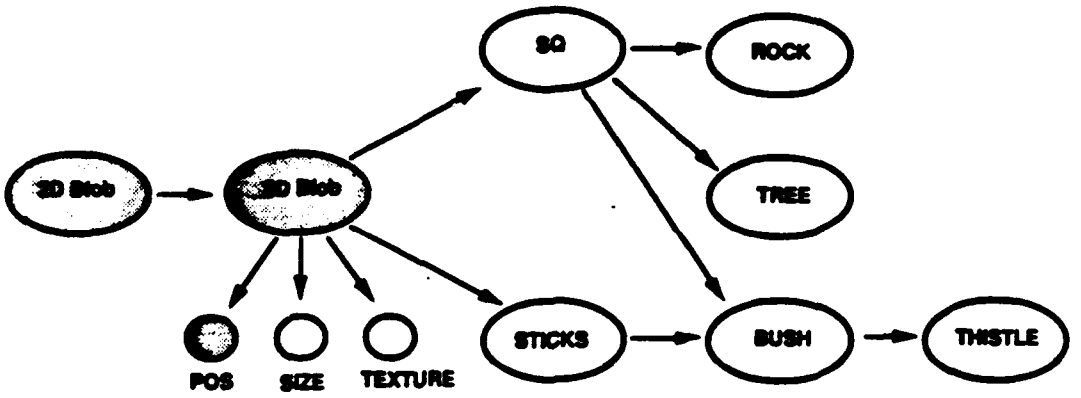


Figure 2: The representation space for the TraX system. The shaded nodes represent components of the representation in use. A new node can be shaded only if one of its connecting nodes is shaded and the stability conditions necessary for its acceptance have been met.

To capture the natural progression of representations supported by better data and to cover the diversity of objects in the world, we have introduced a partially ordered set — a lattice — of representations, which we call *representation space*. The importance of having a lattice is that it focuses the perception system on the most appropriate representations for an object, given both the resolution of the data and the inherent properties of the object.

Figure 2 shows the representation space used in the TraX system, which we implemented to explore the issues associated with a multiple representation system. We consider representation space to be composed of *fundamental representations* and *enhancements*. Each fundamental representation reflects a qualitatively distinct representation, while an enhancement corresponds to the addition of a few parameters to a fundamental representation.[1] In the diagram each large node corresponds to a fundamental representation; each small node, to an enhancement. As indicated, fundamental representations available in our TraX system include 2-d blobs, 3-d blobs, superquadrics (SQ), sticks (a 3-d parts representation described later), and several semantically based representations including bush and tree.

[1] We recognize that there is no formal distinction between levels and parameters. However, the intuition that there are several qualitatively different representations, each of which can be enhanced by the addition of a few parameters, is strong and we have found the distinction useful.

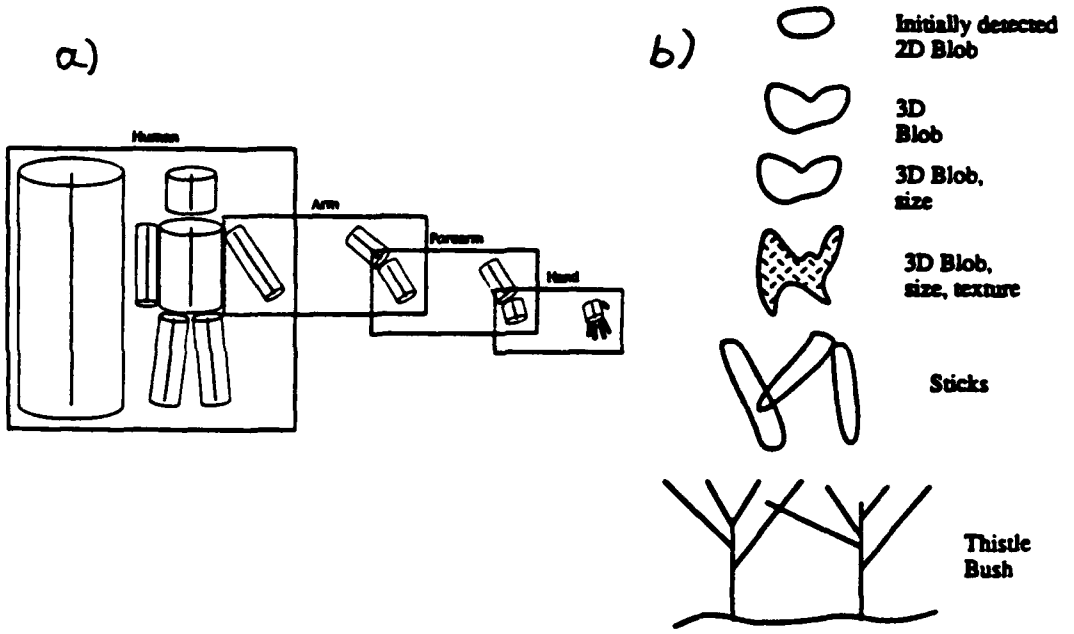


Figure 3: Contrasting hierarchical representations (a) with representation space (b). In representation space, descriptions vary in the types of representations used; additional information not only causes accuracy to improve but also allows an object description to contain different types of information. [Reprinted from Marr and Nishihara, 1978.]

Representation space is similar to scale space [Witkin, 1983] in that the representation of an object is not restricted to any one level of description; different levels of specificity are possible. Unlike scale space, however, and unlike hierarchical representations [e.g., Marr and Nishihara, 1978; Nishihara, 1981; Brooks, 1981] representation space is not homogeneous. For example, Marr and Nishihara propose using generalized cylinders of many scales to achieve a representation that spans data of different resolutions. Although the description of an object improves as more detailed information is acquired, there is no change in the type of inferences the representation can support. Only the size and number of primitives and the corresponding level of accuracy improves. In representation space, however, a change in representation often implies the ability to assert new properties about an object. These two approaches are schematically contrasted in Figure 3.

One of the implications of representation space is that as new data are processed, the description of an object can be modified in one of three ways. First, the parameters of the active components of the representation can be updated. We refer to this process as *refinement*; refinement procedures use standard filtering techniques and are similar to algorithms used by others to reduce parameter uncertainty. [Ayache and Faugeras, 1987; Matthies and Kanade, 1987; Smith, Self, and Cheeseman, 1987; Crowley, 1989]. The second type of change is the activation of a parameter or property attached to an active representation. For example, the active representation indicated in Figure 2 could be expanded by activating the TEXTURE node under 3D–BLOB. This type of modification is referred to as *enhancement*; the representation is enhanced by the addition of a new parameter. The final type of update is *augmentation*; in Figure 2 this would correspond to activating either the SQ (superquadric) or the STICK fundamental representation. The augmentation of a representation for an object means that the object can be described in a completely new vocabulary. As a collection, the methods of modifying the description of an object are designed to combine well-known quantitative techniques for integrating information with a more qualitative approach that permits the nature of a representation to change over time.

Arcs in the representation space diagram indicate ways that the description of an object can be extended; that is, they provide the control structure for the accumulation of descriptions. A new node in representation space can be become active, indicating that the corresponding representation is active for a given object, only if one of its predecessor nodes is active. By shading nodes in this diagram we indicate the active components for a particular object. For example, in Figure 2 the large shaded node labeled 3D–BLOB indicates that a reliable 3-dimensional blob description has been computed for the object. The small shaded nodes labeled SIZE and POS reflect the fact that the size and position of the blob are known.[2] Thus, for this particular object, the TEXTURE, SQ, and STICKS nodes can be activated the next time data for this object is analyzed.

Note that the arcs in representation space do not imply computational dependency. For example, the algorithms in the TraX system for computing a superquadric model of an object are independent of those for computing a 3D–BLOB description. This differs from typical level of abstraction hierarchies where each new descr.ption is computed from the previous level representation; in a typical sequence, lines are computed from edges, planar facets from lines, volumetric primitives from lines, etc. [Hanson and Riseman, 1987]. Such chaining of representations leads to the compounding of processing errors. In contrast, the different levels of representation space can be used to check the validity of a computed description. If the 3D–BLOB predicted by operations performed on the superquadric model

[2] For this discussion we are ignoring the issue of uncertainty in the estimate of a parameter. In actuality, once the measurement of a parameter is determined to be relatively stable we use Kalman filtering techniques to update the value of the parameter and maintain an explicit estimate of the uncertainty of the value.

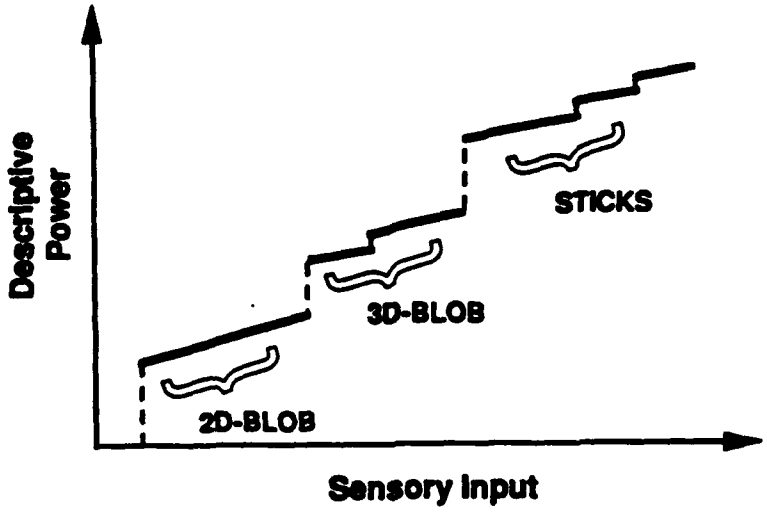


Figure 4: A conceptual graph demonstrating the utility of representation space. The abscissa indicates increased sensory input. The ordinate indicates the "power" of the description of an object as constructed by the system. The large steps indicate *augmentation*, where a new fundamental representation has been activated, and that new representation supports many new inferences about the object. The small steps reflect *enhancement*, where new parameters have been added to a current description. Finally, the increasing slope of the tops of the steps indicates *refinement*, the improvement in the accuracy of the current representation.

is not similar to the blob computed directly from the data, then the system would have evidence that at least one of its descriptions is not valid. While we have not yet explored this issue in detail, we would hope to make use of the independence of representations to increase the overall robustness of the system.

Also, it is important to realize that when a new representation is used to compute a description for an object, the previous descriptions are not discarded. They are retained because they may be the best representation to answer a task-related question, even if they are at a reduced level of specificity or accuracy. Included in Section 3 are examples of employing multiple levels of description for accomplishing different perceptual tasks.

To underscore the point that the construction of a description of an object is a cumulative process, consider the conceptual graph in Figure 4. This graph is intended to reflect the utility of representation space. The abscissa indicates the amount of processed sensory input, which in the case of an autonomous robot is monotonically related to time. The ordinate indicates the "power" of the description of an object as constructed by the system.

As more data are processed, the description of an object becomes more precise and thus more "powerful." The large steps indicate augmentation, where a new level of representation has been activated which supports many new inferences about the object. The small steps reflect enhancement, where new parameters have been added to a current description. Finally, the increasing slope of the tops of the steps indicates refinement, the improvement in the accuracy of the current representation.

3 Stability and Validity

Representation space controls the order in which representations are explored for describing a particular object. How does the system decide that a computed description is *valid* and, therefore, should be added to the object's model? In this context we use the term valid to mean that the description correctly characterizes some aspect of the object, as opposed to being a transient artifact of the processing. To address the question of validity we must consider the causes of artifacts.

Artifacts can arise for several reasons. Computing a description of an object using an inappropriate representation can easily lead to a model that does not reflect any intrinsic property of the object; for example, a stick description of a boulder is mostly determined by idiosyncracies of the stick fitting algorithm. Also, artifacts can arise because of rare, yet expected, events that violate the assumptions embodied in the processing; for example, accidental alignment can make two objects appear to be one larger object. Finally, artifacts can occur because of unmodeled errors; for example, a segmentation algorithm can hallucinate an object from an unlikely variation in the data. The ability to determine when a description is valid is important for any perception system; it is critical in a multi-representational framework in which many descriptions are tried and only a few characterize an object well.

Our current approach to assessing the validity of a computed description relies on an analysis of temporal stability. We do this by tracking an object over time, computing a new description of it in each image, and then analyzing the sequence of these independently computed descriptions. If the descriptions are similar over a period of time, we compute a composite description, validate it as a real entity, and add its description to the model of the scene. For example, if a particular stick description is computed repeatedly for a part of an object, we assume that the consistency across independently computed descriptions is due to a real structural property of the part, and therefore the stick description of the part is added to the model of the object.

Given this basic strategy, there are two key phrases that need to be functionally defined in order to convert it into an algorithm: "similar descriptions" and "over a period of time." To define these terms, we ideally would like to rely strictly on strong models of components

of the perception system, such as the physics of the sensor, its noise characteristics, and the characterizations of the image analysis techniques. However, in practice, these models are not adequate to completely predict the behavior of the system. As a result, we use these explicit models when they are available and, when necessary, augment them with statistical, empirically determined models. For example, we predict where we expect to see a previously detected object and how large it will be from a combination of three strong models: a model of the physics of the range sensor, a model of its scanning geometry, and a model of the vehicle's motion based on inertial navigation data or land navigation data. However, to predict how frequently an object might be missed by our low-level segmentation procedures, we built a simple statistical model by applying the procedures to hundreds of images and accumulating failure statistics.

Deep and Shallow Models and Explanations

Jain and Binford [1991] use the term "shallow" to refer to statistically derived models; we will thus use the term "deep" to refer to models derived from known physical systems. These deep models, such as the model of the range sensor's scanning technique, can support quite precise predictions and can be embedded in algorithms that cover a wide range of tasks. For example, we use the scanning model of the sensor in conjunction with high-frequency inertial navigation data (i.e., a set of measurements for each image scan line) to compensate for the bouncing of the vehicle during the four tenths of a second required to gather a range image using the Environmental Research Institute of Michigan range sensor. This process not only corrects for bouncing, but also corrects for the relativistic effect that causes vertical telephone poles to bend in the imagery because the vehicle is significantly closer to the pole when it measures the bottom of the pole than when it measures the top. Deep models are robust in the sense that they always contribute an accurate characterization of the processing system.

Shallow models, however, such as our simple statistical model of the failure frequency of our segmentation procedures, are always suspect. A commonly cited reason for their restricted utility is the difficulty of ensuring that the training set adequately covers the range of expected scenes [Duda and Hart, 1973]. A more serious difficulty with shallow models is that situations arise in which the application of those models is inappropriate. As mentioned, we use the empirically determined probability of failure of a segmentation algorithm to determine the number of times an object should be detected in a sequence of imagery before being declared valid. With our current segmentation procedures this threshold is set at three. However, there are many reasons other than the failure of the segmentation procedure for an object to be undetected in a given image, and, furthermore, these situations are predictable from additional system component models. Thus, for robust performance, the use of shallow models must be tempered by *explanations*, here defined to

be understood situations that cause shallow models to be inappropriate. For example, the decision about whether an object has left the field of view is based on the model of the sensor's scanning technique and the vehicle's location estimates; this is an example of an explanation based upon a deep model. On the other hand, one of the common mistakes of our system occurs in our column-oriented analysis of the raw range image. For that problem, we only have an ad hoc description of the pathology of an error, in this case a columnar plume suddenly appears in the shape of the object. If some object is not matched in a new image, and an apparently new object has appeared with a large columnar piece, the system explains the situation as being a possible error in processing.

The idea of a *possible* error introduces our last point about employing shallow models: decisions based upon shallow models are always suspect and should be confirmed by further data. As such, it often becomes necessary for the system to maintain multiple, competing hypotheses about the state of an object. Continuing the example of the last paragraph, the system does not absolutely conclude that the newly shaped object is indeed a hallucination caused by a processing error. Rather, a wait-and-see attitude is adopted, and both possibilities are maintained; the processing of subsequent images resolves the ambiguity.

3.1 Stability in blob detection

The first representation used to describe an object is 2D-BLOB. In the TraX system, the 2D-BLOB description of an object consists of the range pixels corresponding to the object, as viewed in the most recently processed image. This representation is important not only because it is the first instantiation of a model for an object and therefore endows existence to some object, but also because the actual pixels viewed in one image are best description for matching that object in the next image of a sequence. The question of whether a 2D-BLOB description is valid is really a question of whether the segmentation process correctly detected a real obstacle, or it mistakenly isolated some pixels that are part of the ground. Robustly detecting obstacles and tracking these objects from image to image are critical in a system that integrates information over time to construct reliable models.

The segmentation procedure in the TraX system consists of classifying each pixel in each range image as ground or obstacle. This classification is made by applying a multi-step procedure that first identifies regions in the image that are well fitted by planes. We next determine the consistency of these planes with an a priori digital terrain map (DTM), using orientation as the principal factor. The consistent planes are then extended to completely cover the gaps between them, essentially forming a new, local DTM. Any range pixels that are more than a certain distance above or below this new DTM are then marked as obstacles. Because the ground clearance of the Martin Marietta Autonomous Land Vehicle is about six inches, we use that value as a threshold. Notice that because we make use of

the temporal stability of the detected obstacles to validate the segmentation, we can set the threshold according to the specifications of the task, instead of concentrating on the expected single image false alarm rate.

As mentioned earlier, we use a shallow model of obstacle detection to determine the best control strategy for assessing validity. Empirically we have determined that if an object detected in three consecutive images, then there is a very high probability that it is a real object. In addition, once an object has been validated, it is unlikely to be missed twice consecutively. We implement this simple control strategy using a quasi finite-state machine (QFSM). We use the term "quasi" because as an object moves through these states the history of its traversal is recorded and can sometimes affect operations that occur outside the FSM control structure.[3]

Figure 5 shows a simplified portion of the QFSM used for the analysis and tracking of two dimensional blobs. Notice that there are several ways to enter the **Initially-Detected** state, including being detected in the first image of the sequence, coming out from behind another object, and splitting off a previously detected object. The importance of making these paths explicit is that we can later use this information to help explain unexpected phenomena and to affect decisions made outside the control structure of this QFSM, such as deciding how to combine the models of two objects later decided to be only one object.

Once an object is **Initially-Detected** we try to match that object in subsequent images. As an object is successfully matched it moves into the **Stable** state; at this point the object is considered to be "real" and attempts to extend the description are begun. If, however, after initial detection the object is no longer matched, the object quickly moves to the **Artifact** state indicating that the detected obstacle is an artifact of some processing step and should be discarded.

Notice, that at each state there is a **missed-but-can-explain** transition. This type of arc represents a situation where the object is not successfully located in an image in which it is expected, but there is an external explanation as to why not. Increasing the competence of the system requires recognizing these situations and incorporating explanations of them into the evaluation process. We currently have implemented the analysis required to support the following explanations:

- The object is no longer in the field of view of the sensor.
- The object is occluded by another known object.
- The object is a small, short blob far away so it can be easily missed.

[3] We could implement the control structure using a true FSM by simply increasing the number of states. We choose not to do so because we would end up with many states that were qualitatively similar, obscuring the general structure.

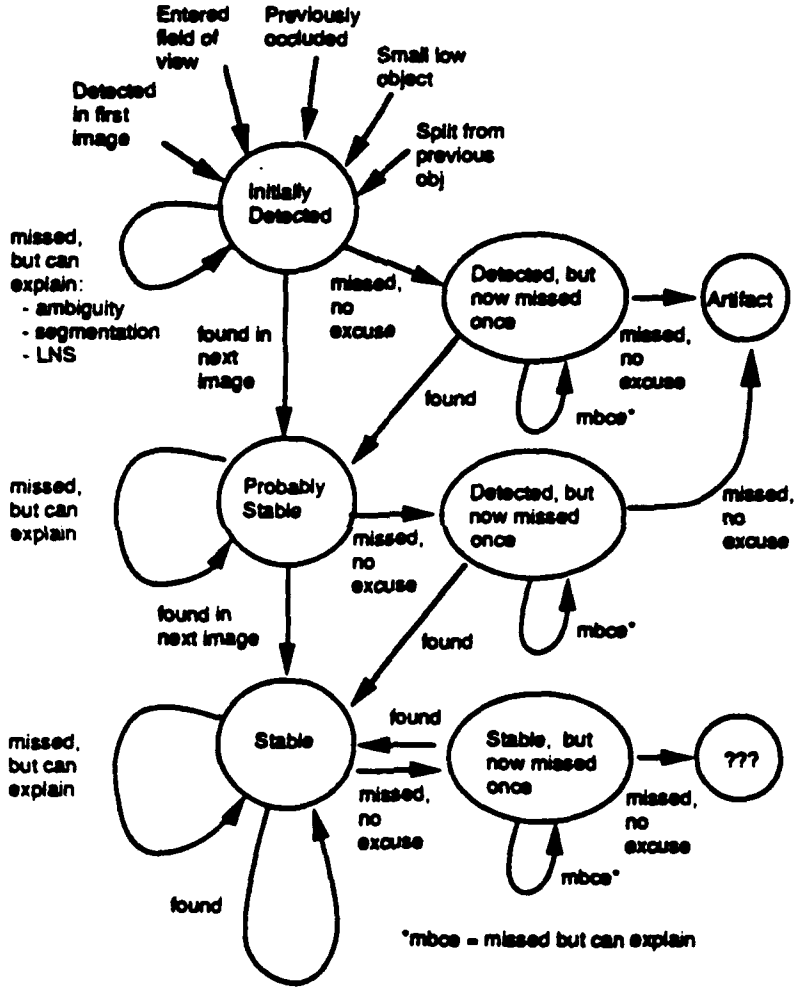


Figure 5: Part of the finite-state machine for determining stability of 2D-BLOBS.

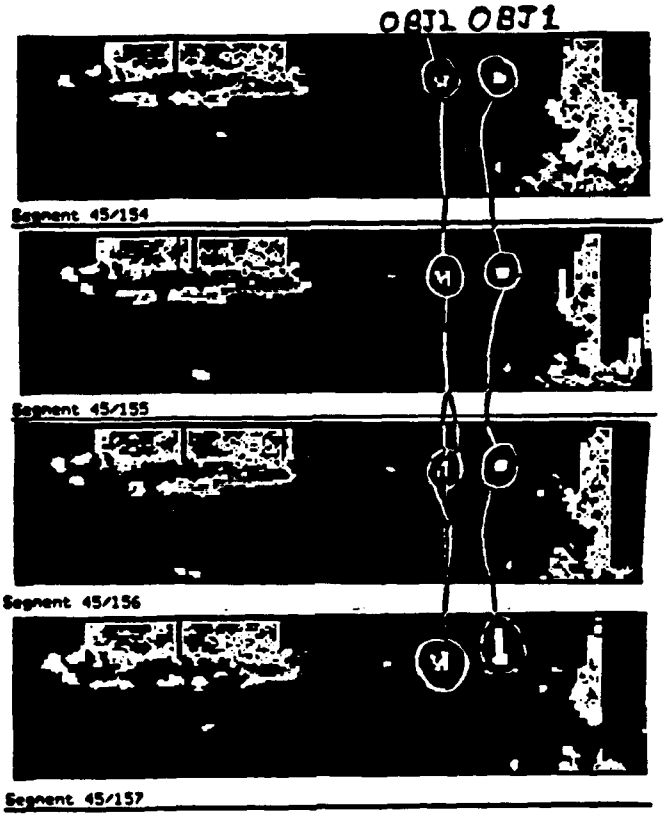


Figure 6: Tracking detected obstacles from image to image.

- The object merged with another object to form a larger object.
- The object is unmatched because an error in the ambiguity interval assignment greatly changed the apparent characteristics of the object. (Ambiguity interval assignment is a preprocessing step necessary for determining the true range from a phase shift range image.)

Figure 6 is a sequence of segmentation images produced by the single-image analysis. Object 1 (the short object on the right) is detected in all 4 images. Initially, this object is not matched in the fourth image because the object's shape has changed dramatically. However, the change is mostly characterized by the addition of a column of pixels in the last image. In the TraX system, a column scanning algorithm is used to disambiguate the phase-encoded

raw range signals; we refer to this processing as ambiguity interval assignment. Therefore, a column shaped change in the appearance of an object is symptomatic of a particular mistake, namely an ambiguity interval assignment error. Thus, in this example, the program concludes that an ambiguity error has possibly occurred. The TraX system retains information about this error as indicated by the following portion of program output:

```
...
Starting view 157 ...
  (OBJECT-TRACKER-MBCE <OT ID:22 :GENERIC-OBJECT-4> AMBIGUITY-INTERVAL-PROBLEM-WITH-BLOB
   <TRAX-RANGE-BLOB B:25/157> BLOB-HAS-THE-EXTRA-PIECE 1-TO-1 CASE-6)
...
  Creating OT <OT ID:41 :GENERIC-OBJECT-23> for region <TRAX-RANGE-BLOB B:25/157>.
...
```

This fragment indicates two things: First :GENERIC-OBJECT-4 (referred to as object 1 in Figure 6) was missed (not seen as expected) but could be explained (MBCE) by an ambiguity interval problem in the processing of blob 25 in image 157. Second, the system also starts a new object tracker (OT for :GENERIC-OBJECT-23) as a competing hypothesis that needs to be resolved in later processing. Because the original object :GENERIC-OBJECT-4 was seen again in images 158 and 159, and because :GENERIC-OBJECT-23 is not matched, the newly created object is quickly eliminated, with the explanation being that its creation was indeed an artifact.

Object 2 (the thistle bush to the left of object 1 in Figure 6) is an example of a single object splitting (third image) and then merging again. In order to build a robust model of the environment the program must be able to handle situations such as these. Again, the TraX system handles these ambiguities by generating competing hypothesis and resolving them with the processing of additional data. The density of these events in this short sequence is higher than usual, but they are typical of the events that occur in our analysis of hundreds of images.

In the future we plan to expand the list of possible explanations. As we understand more of the fundamental properties of objects and more about the behavior of the analysis procedures we can implement more explanations, increasing the competence of the system.

3.2 Stability in integration

Once we have determined an object is real, we have a set of techniques for generating 3-dimensional descriptions. The simplest uses a "3D-BLOB" analysis which describes an object according to its position, size, and, potentially, surface texture. Initial 3-dimensional analysis of a blob establishes an object-centered coordinate system and then computes a 3-dimensional scene location for the object. The object centered frame allows for the integration of information about the shape of the object to be decoupled from the compiling

of information about the location of the object. The actual representation of this new "3D-BLOB" is an ellipsoid whose parameters of position and size are updated using standard Kalman filtering techniques. Critical to this integration is knowledge of the noise characteristics of the sensor.

When more precise range data are available, we can compute 3-dimensional part models using one of two representations: superquadrics and "sticks." These representations support a class of inferences about objects that are not supported by the blob descriptions, namely those requiring a structured shape description.

Superquadrics are well suited to describing shapes composed of compact parts [Pentland, 1986a; Pentland and Bolles, 1988]. The technique we use to compute superquadric models of objects is a modified version of the algorithm described in [Pentland, 1986b]. We first compute a "minimal cost covering" of the range pixels by executing a coarse global search over the superquadric parameter space, and then optimize the model by gradient descent. We have found this technique adequate for modeling simple objects such as rocks, but have not exercised the algorithm enough to evaluate it fully.

When objects are composed of thin pieces, as are fence posts and thistle bushes, the response of the range sensor tends to "fatten" the parts by generating mixed pixels along the sides. This blurring prevents the superquadric algorithm from finding the true stick-like description. To model these thin objects we have designed a special representation we call "sticks." By definition sticks appear as one-pixel wide lines in range images. Thus, to compute a stick model of an object, we first thin the range image of the object, and then compute a minimal covering in a manner analogous to superquadrics. The stick model representation is used in the bush example presented later in this paper.

Figure 7 displays the results of applying the stick-fitting procedure to a detected object. Each model is computed independently making no use of the previous solution. Note that most of the resulting models capture some structure of the bush. However, except for the last one, none captures all of the structure. The principal problem associated with these fitting techniques is the lack of data to constrain the models. As a result, there are often many descriptions that characterize an object equally well. As with obstacle detection, we rely on processes monitoring the stability of computed descriptions to filter out those that are not valid.

To integrate stick descriptions over time, we employ a method similar to that previously discussed for tracking of 2D-BLOBs. In this case, however, new sticks computed from the data are matched to model sticks that are being refined with each image. Model refinement requires three stages: First, model sticks that are matched by new sticks are reinforced in terms of their stability, and their parameters are updated using standard Kalman filtering techniques; the state variables estimated are the endpoints of each stick [Ayache and Faugeras, 1987] and we use our model of the sensor and its noise characteristics

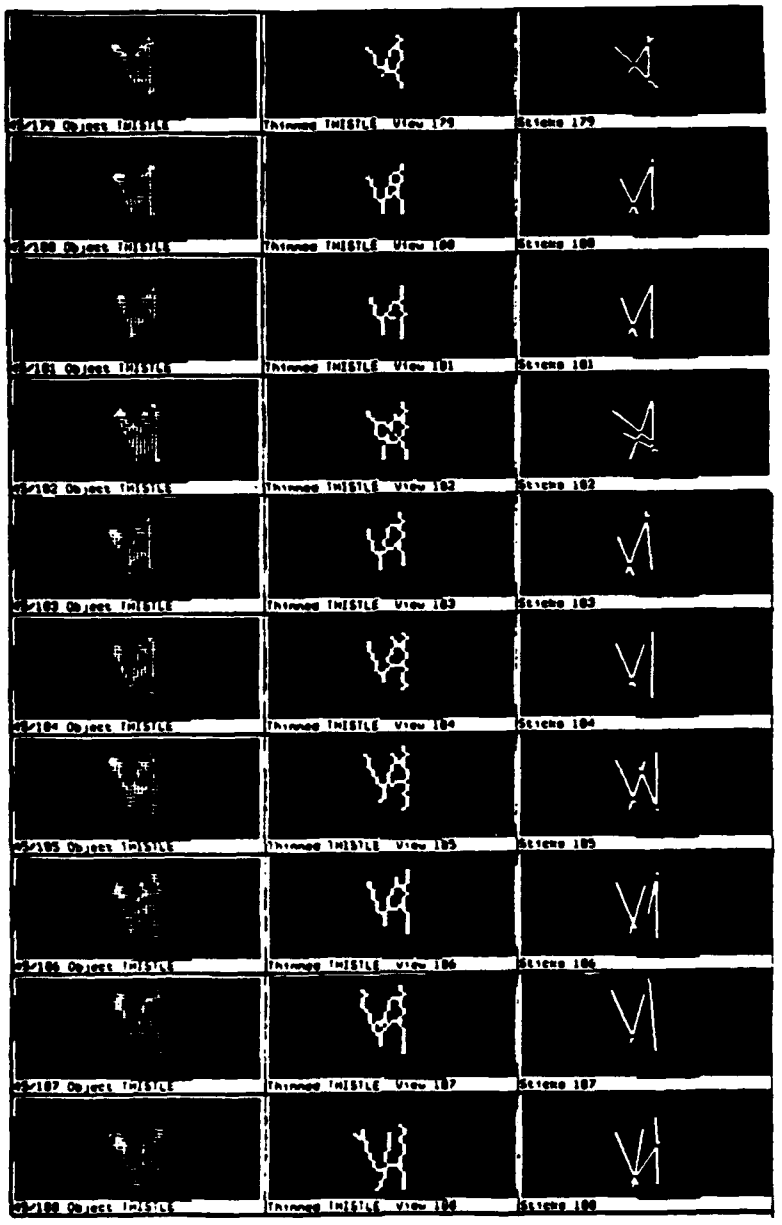


Figure 7: Computing stick descriptions of a thistle bush. The column on the left displays the silhouette of the object as determined by the obstacle detection procedure; the middle column is the thinned version of these objects. The right column displays the best "stick" model of the the thinned bush as computed independently for that one image. Note that some of the sticks are quite robust, such as the vertical stick on the right. Others are less stable, while some are artifacts. Though the fitting technique can be improved, our goal is to use temporal stability to compute a more robust model.

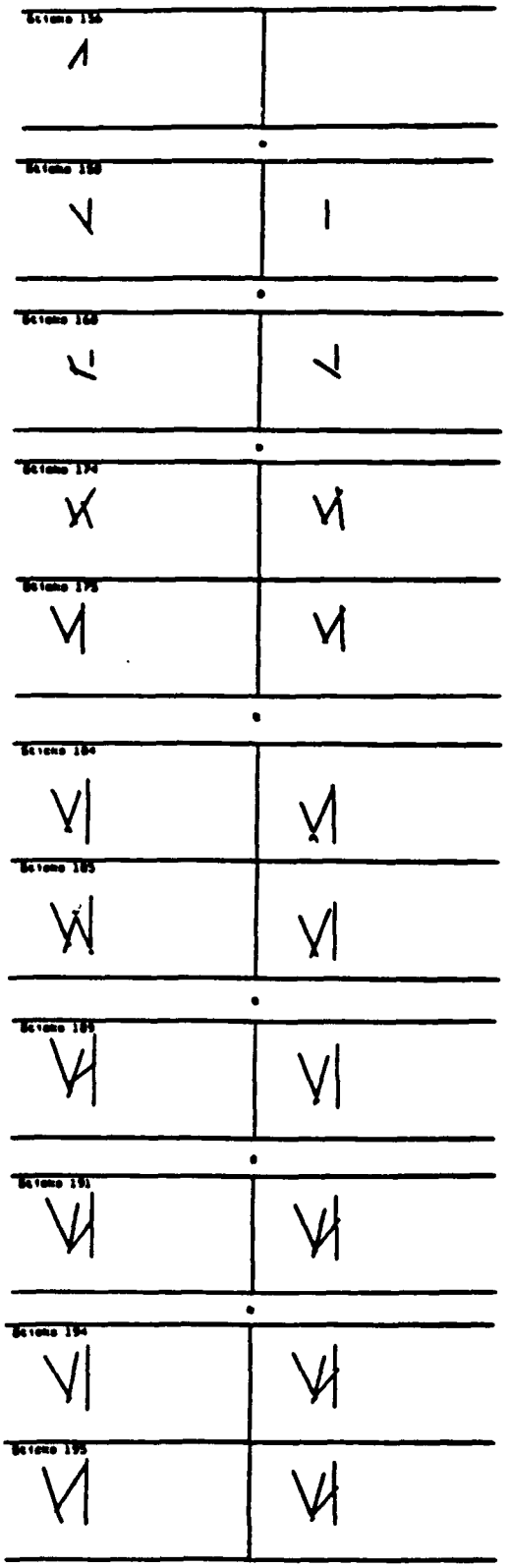


Figure 8: Sticks computed independently (left column) and tracked over time (right column). As a stick becomes stable it is added to the model on the right.

to estimate the variance of the new measurements. Second, unmatched new sticks cause the formation of model sticks that are initialized from the data; these sticks are searched for in subsequent images. Finally, sticks that were initially detected but not matched again are eventually discarded as an artifact of the stick-fitting procedure, unless there exists some explanation as to why the stick should not be matched. Currently, we include only one explanation that allows an unmatched stick to remain as a viable part of the representation: the vehicle has backed away from the object and the individual stick may no longer be detectable by the sensor.

Figure 8 shows an example of the stability analysis applied to sticks. On the left is the stick description computed independently using the single range image as input. On the right is the set of stable sticks tracked over time. A new stick is added to the model on the right only after it has been deemed stable. Note that the stable description converges to (what is known to be) an accurate model of the bush.

4 Hard Problems

4.1 Quasi-stability

In our approach to temporal integration, temporal stability of a description is the primary indicator of reliability. The basic assumption of this strategy is that for each appropriate representation there is a unique, correct description of an object and that commonality across independently computed descriptions reflects valid aspects of those descriptions. However, the appropriateness of this assumption depends upon the match between the objects in the domain and the representations. Consider, for example, the image of an object shown in Figure 9. In this case, describing the object as a two-stick 'X' or as a three-stick 'H' is equally correct. And, stability cannot disambiguate between these models because they may each occur periodically; such an occurrence leads to two competing stick models, each of which is "quasi-stable."

While we do not have a complete solution to this problem, we can avoid most of these difficulties in the TraX system by keeping track of competing hypotheses. Thus, the system could maintain two or more descriptions of an object in one representation, and clearly mark them as alternatives. If asked to assert or predict a property of this object (such as appearance from some viewing direction) the system would have to decide which description or which combination of descriptions it should use, just as it does now when answering a task-level request about an object that has multiple descriptions derived from different representations. While this remedy may be adequate for many problems, it is clearly unsatisfactory for situations in which there are many quasi-stable descriptions of an object.

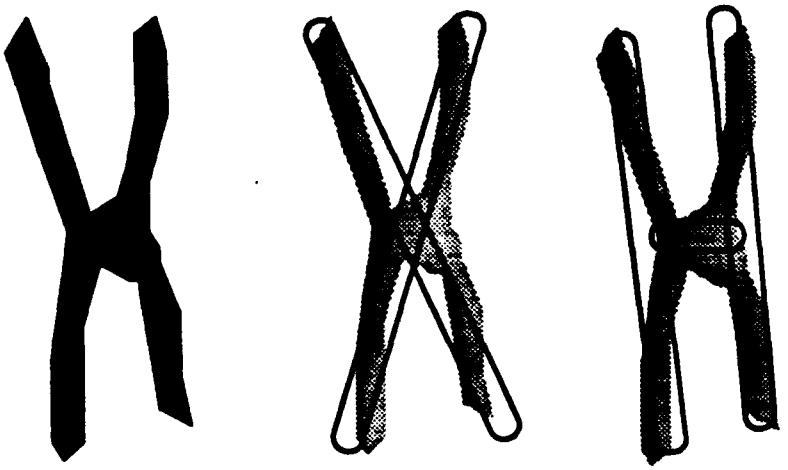


Figure 9: A thin object equally well described as a as a two stick 'X' or as a three stick 'H'. Stability analysis cannot be used to resolve the ambiguity.

4.2 Dynamic worlds

In the TraX system, the scene is assumed to be static: other than the vehicle, objects do not change their location, orientation, or shape. What are the issues in extending our approach to a dynamic environment?

One response is to simply model the dynamics of the environment. In this case variables such as velocity and acceleration become additional parameters of the representation; aside from incorporating these new variables into the prediction mechanisms, the approach to temporal integration remains the same. In this case however, stability becomes much harder to assess. If an object is moving (e.g., a rotating windmill) how does one determine that the shape description computed is stable, implying that the description is valid? Presumably, an understanding of the dynamics would need to be included in the model itself.

Another issue raised by a dynamic environment concerns the matching of known objects

to objects detected in the data. If an object can change in location, orientation, and shape, how does one determine correspondence? Without digressing into a philosophical discussion of ontology or Lincoln's axe,[4] we must consider how to insure that one is integrating information about the *same* object. One insight to this problem is derived from the work on motion analysis developed by one of the authors [Baker and Bolles, 1989] where the temporal sampling rate is great enough that all transitions over time are smooth respect to the data. Thus, for example, tracking the movement of an object, such as a person's arm, is simplified because the object can be easily tracked from frame to frame. That approach requires that the data sampling rate be high enough to smoothly sample the dynamics of the domain.

4.3 Explanations and Hallucinations

In this paper we have made the claim that increasing the number of explanations that the system may invoke to explain why the actual sensory data deviated from predicted data increases the overall competence of the system. The intuitive argument supporting this claim is clear: the greater the number of important events that the program can diagnose, the less likely the data are to confuse the model construction process.

However, one must be aware that if the system has enough explanations, then the system can find an explanation for anything. As an extreme example, suppose the explanation "The sensor is completely broken and the incoming signal is independent of the world." is part of the knowledge base of the system. Then any data may be explained by such a statement, and no useful modeling occurs.

To avoid such confusions we have to resort to the idea of *best* explanation, where best means most likely according to some a priori model of the world. This approach is the same as adopted by researchers employing minimal encoding strategies to select the best description of a scene; examples include segmentation [LeClerc, 1990] and part descriptions [as done here and Pentland and Bolles, 1988].

To date, we have avoided addressing this problem by placing stringent preconditions on the invocation of most explanations. These conditions are strong enough that if they are satisfied, we are willing to state categorically that the explanation is appropriate. In the few instances of shallow model explanations where strong preconditions do not exist, the requirement that the weak conditions remain true over an extended period of time prevents the explanations from becoming too widely applied.

[4] Old joke: A farmer displays an axe over his mantle with a sign that read "Abe Lincoln's Axe." When a skeptical visitor enquired about its authenticity the farmer replied: "Yup, it sure is Lincoln's axe. I've had to replace the handle twice and blade once, but it's old Abe's."

5 Summary

We have described a new representation paradigm that supports concurrent evolving descriptions of an object. Our rationale for developing this paradigm is as follows:

- Multiple, concurrent descriptions are required for two reasons: (1) to describe the wide variety of objects that occur in complex domains, such as the outdoor world, and (2) to efficiently support the inferences required by a collection of task modules, including object tracking, path planning, obstacle detection, and landmark recognition.
- Not all representations are appropriate for every detected object. Sometimes the data are not sufficient to support the representations. And sometimes the representations are simply not appropriate for the object, such as a fractal model of a hippopotamus. Therefore, to restrict the application of representations to appropriate objects, we introduce the idea of a representation space, which imposes a partial ordering on the set of available representations.
- For applications in which a continuous stream of data is available, the descriptions of an object can evolve in two ways. First, the parameters of a representation can be refined by filtering techniques as new data are acquired. And second, if the data improve over time, new descriptions can be added, when they are supported by the data.
- Temporal consistency across independently computed descriptions of an object is a strong indication of the validity of the descriptions. If the same description is computed from several images in a row, there is a high probability that the description captures a real structural aspect of the object.
- Since there are many reasons for a description to change from one image to the next, the idea of temporal stability can be significantly enhanced by the addition of explanations that account for the problems and special cases that invariably arise in the processing of real imagery. The sources of explanations range from deep models, such as the physics of the sensor, to shallow models, such as the probability that a low-level procedure makes a mistake.

The ability to change an object's description incrementally and to build a temporally persistent, yet consistent model of the environment is crucial in autonomous navigation tasks: objects are viewed many times, from different viewpoints, and with different resolutions. By continually updating the objects' descriptions, a robot is in a position to base its decisions on the most current information at all levels.

Acknowledgments

We gratefully acknowledge the comments of the SRI vision group and the assistance of Tom Strat, Andy Hanson, Steve Barnard, Helen Wolf, and Lynn Quam. The range data was gathered at Martin Marietta, Denver CO with the assistance of Steve Seida, Jim Allison, Terry Dunlay, and Laura Haber. This research was supported in part by DARPA contracts MDA 903-86-C-0084 and DACA 76-85-C-0004.

References

Agin, G.J. and T.O. Binford [1973], "Computer Description of Curved Objects," *Proc. of the Third IJCAI*, Stanford, CA, August, 1973, 629-640.

Ayache, N. and O. D. Faugeras [1987], "Maintaining Representations of the Environment of a Mobile Robot," *Proc. of International Symposium on Robotics Research*, Santa Cruz, CA.

Baker, H. and R. Bolles [1989], "Generalizing Epipolar-Plane Image Analysis on the Spatiotemporal Surface," *Int'l J. of Computer Vision*, **3**, 33-49.

Brooks, R.A. [1981],"Symbolic Reasoning Among 3-D Models and 2-D Images," *J. of Artificial Intelligence*, **17**, 285-348.

Crowley, J. L. [1989], "World Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging," *Proc. of Conference on Robotics and Automation*, Scottsdale, AZ, 674-680.

Duda, R. and P. Hart [1973], *Pattern Classification and Scene Analysis*, J. Wiley & Sons, New York.

Hanson, A.R. and E.M. Riseman [1987], "The VISIONS Image Understanding System," in *Advances in Computer Vision*, C. Brown (Ed.), Erlbaum Press.

Jain, R. and T. Binford [1991], "Ignorance, Myopia, and Naivete in Computer Vision Systems," *CVGIP*, **53**, 1, 112-117.

Marr, D. and H. K. Nishihara [1978], "Representation and recognition of the spatial organization of three-dimensional shapes," *Proc. R. Soc. Lond. B*, **200**, 269-294.

Matthies, L. and T. Kanade [1987], "The Cycle of Uncertainty and Constraint in Robot Perception," *Proc. of International Symposium on Robotics Research*, Santa Cruz, CA.

Nishihara, H. K. [1981], "Intensity, Visible-Surface, and Volumetric Representations," *J. of Artificial Intelligence*, **17**, 265-284.

Oshima, M. and Y. Shirai [1978], "A Scene Description Method Using Three-Dimensional Information," *Pattern Recognition*, **11**, 9-17.

Pentland, A. P. [1986a], "Perceptual Organization and Representation of Natural Form," *J. of Artificial Intelligence*, 28, 293 - 331.

Pentland, A. P. [1986b], "Recognition by Parts," SRI Technical Report 406.

Pentland, A. P. and R. C. Bolles [1988], "Learning and Recognition in Natural Environments," to appear in the *Proceedings of the SDF Benchmark Symposium on Robotics Research*, MIT Press.

Popplestone, R.J.,C.M. Brown, A.P. Ambler, and G.F. Crawford [1975], "Forming Models of Plane-and-Cylinder-Faceted Bodies from Light Stripes," *Proc. of the Fourth IJCAI*, Tbilisi, Georgia, USSR, 664-668.

Smith, R., M. Self, and P. Cheeseman [1987], "A Stochastic Map for Uncertain Spatial Relationships," *Proc. of International Symposium on Robotics Research*, Santa Cruz, CA

Witkin, A. [1983], "Scale Space Filtering," *Proceedings of IJCAI*, 1017-1022.

Appendix C

SRI Image Understanding Research in Cartographic Feature Extraction,
Lynn H. Quam and Thomas M. Strat

to appear in ISPRS-II,
Munich, Germany, September 1991.

SRI IMAGE UNDERSTANDING RESEARCH IN CARTOGRAPHIC FEATURE EXTRACTION

Lynn H. Quam and Thomas M. Strat
Artificial Intelligence Center
SRI International
333 Ravenswood Avenue
Menlo Park, California 94025

Abstract

This paper describes image understanding research at SRI International in the area of computer assisted extraction of cartographic features. Several techniques that are well-suited to semi-automated photo-interpretation are described. These algorithms and others have been implemented within the framework of the Cartographic Modeling Environment, a highly capable, interactive programming environment that has been designed to facilitate the construction of image understanding systems.

1 Introduction

It is generally accepted that human levels of performance in image understanding will require the use of massive computational resources as well as the representation and use of massive amounts of domain knowledge. We believe that continuing advances in computing technonology are likely to provide adequate computational resources, but that progress will be limited by our ability to apply knowledge of the domain effectively. There are many kinds of knowledge that must be exploited to automate feature extraction, including:

- Physics of the imaging process. This includes the geometry and photometry of illumination source, surface materials, atmospheric effects, and sensors.
- Geometry and photometry of specific objects. The shape and appearance of objects that are to be identified and measured must be modeled, including allowance for variations in imaging conditions and seasons of the year. This must include the capability to distinguish objects of interest from other objects in the scene.
- Spatial relationships and constraints between objects. The classification of many objects depends upon contextual constraints, which must be exploited by automated feature extraction algorithms. For example, the facts that rivers flow downhill and that roads (particularly railroads) generally have steepness constraints are vital to the development of reliable recognition techniques.

It is our belief that totally automated cartographic feature extraction is so far beyond the current state of the art that near term payoffs must be based on approaches involving man/machine cooperation. In our research, we endavour to achieve a range of intermediate goals, starting with computer aids to improve the productivity and/or accuracy of manual techniques, and extending to operator-guided application of specialized semi-automated and automated techniques.

In this paper we present a few techniques for extracting specific classes of features, discuss the kinds of knowledge they exploit, and discuss their limitations. We then present a computational framework we have developed which allows one to build comprehensive feature extraction systems that exploit geometric knowledge easily and effectively.

2 Semi-Automated Feature Extraction Examples

In this section we provide a sampling of research results aimed at automating cartographic feature extraction, drawing primarily from those that have been developed through the years at our own laboratory. The techniques we have chosen to include span the range from those that exploit almost no domain knowledge, but have a wide range of potential application, to techniques that exploit increasingly more specific domain knowledge, but have a rather limited range of application.

2.1 Scene Partitioning

It is often desirable to group pixels into coherent regions as a precursor to providing a semantic description of features in a scene. *Partitioning* is the process of segmenting an image into regions that are homogeneous in some set of local attributes, such as intensity, texture, or color.

The scene partitioning task can be formulated as a Minimum Description Length (MDL) optimization problem [8]. In this formulation, the quantity to be minimized is the complexity of describing an image in a given language, where complexity is defined as the number of bits in the description. The choice of language reflects prior knowledge of the class of scenes and noise processes in question, and how they are combined into a final image.

One such language was designed for a simple yet very general class of scenes (including, e.g., aerial images of urban areas) in which all objects are approximated by piecewise-smooth surfaces with piecewise-smooth coloration, and for which the noise process is modeled as additive uncorrelated Gaussian noise with piecewise-constant variance. The resulting language partitions an image into regions whose intensity is described by a low-order two-dimensional polynomial and whose boundaries are described using a chain-code.

The search for the simplest description uses a finite-element grid to represent the underlying image (i.e., the two-dimensional polynomials and their boundaries).

Figure 1: Example of Leclerc's partitioning results.

Each element of this grid represents a polynomial within a unit square.

With this representation, the objective function (overall complexity) can be written as the sum of spatially local terms that involve only the polynomial coefficients of an element, those of its four neighbors, and the pixels of the given image. Finding the vector of coefficients that minimizes this objective function is difficult because the objective function has exponentially many local minima, so that standard optimization techniques cannot be used. Leclerc has devised an approximate solution technique based on a general approach called a "continuation method." An example of a partitioning produced by this approach is shown in Figure 1. The formulation has no specific knowledge of the characteristics of features in the scene, except the assumption that features of interest appear as piecewise-smooth regions in the image. The generality of this model allows the approach to be used to extract many different features in widely varying imagery.

Intuitively, the partitioning process greatly reduces the amount of information needed to describe the image. The remaining challenge is to devise techniques to extract the desired features from the resulting partitions. One near term approach is to use a human to guide the interpretation process.

2.2 Low-Resolution Road Tracking

At low resolution (defined here to be 1 or 2 pixels for the entire width of the road), roads are often indistinguishable from other linear features appearing in the image including artifacts, such as scratches. Thus, the low-resolution road tracking problem largely reduces to the general problem of line (as opposed to edge) following. Nevertheless, there are still some weak semantics that can be invoked to specifically

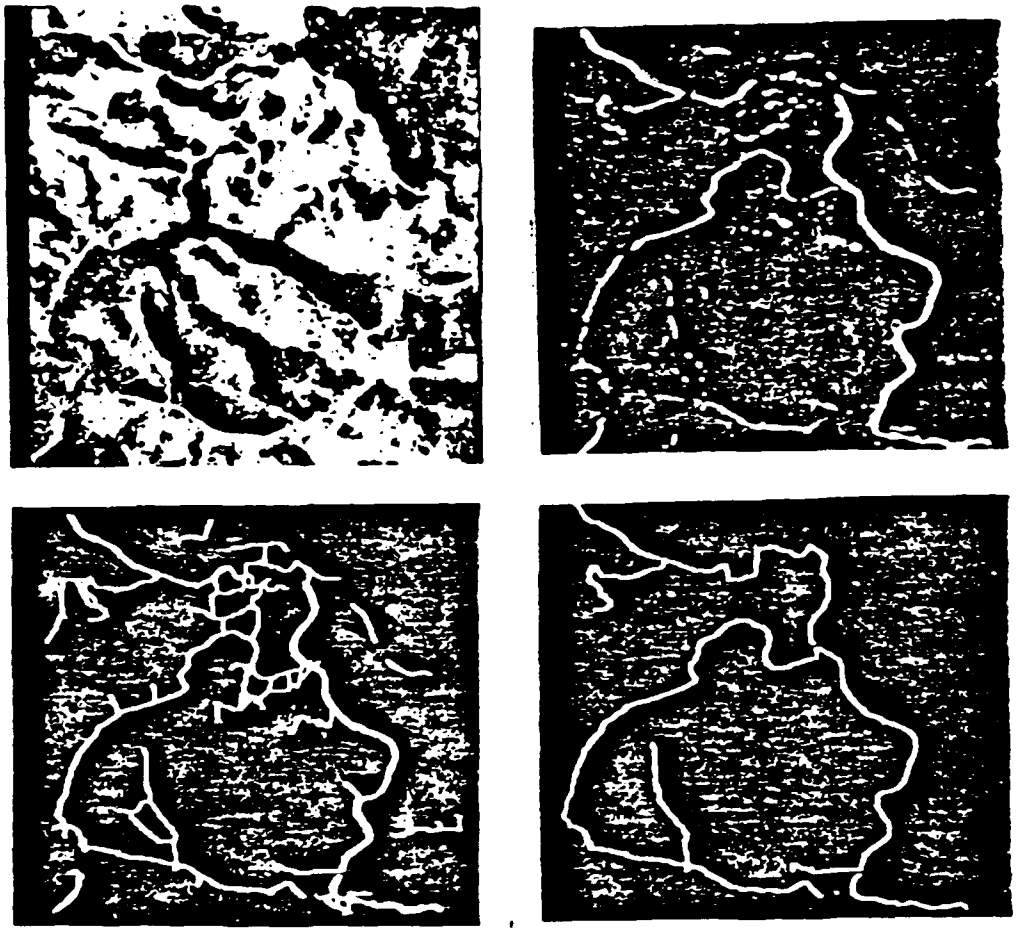


Figure 2: Low-resolution road segment extraction. (a) Intensity image of road scene. (b) Perfect road score mask (PRS) of image (derived from scores of local operators described in [1]. (c) Minimum spanning trees for all clusters of PRS. (d) Maximum paths through minimum spanning trees with length greater than 60 points for all clusters.

tailor a system for road tracking, trading some generality for significant increases in performance.

The basic paradigm employed by Fischler *et.al.*, [1] is to first evaluate all local evidence for the presence of a road at every location in the search area, and then find a single track which, while satisfying imposed constraints (such as continuity), optimizes the sum of the local evaluation scores (costs) associated with every point along the track. Figure 2 illustrates the approach. The results of this automated approach can be edited interactively to improve the quality of the product. Such a semi-automated process promises to be more efficient and more accurate than a purely manual extraction.

A major component of the approach, the F* algorithm described in Fischler *et.al.*, [1], iteratively finds an optimal path in an image from a starting pixel to

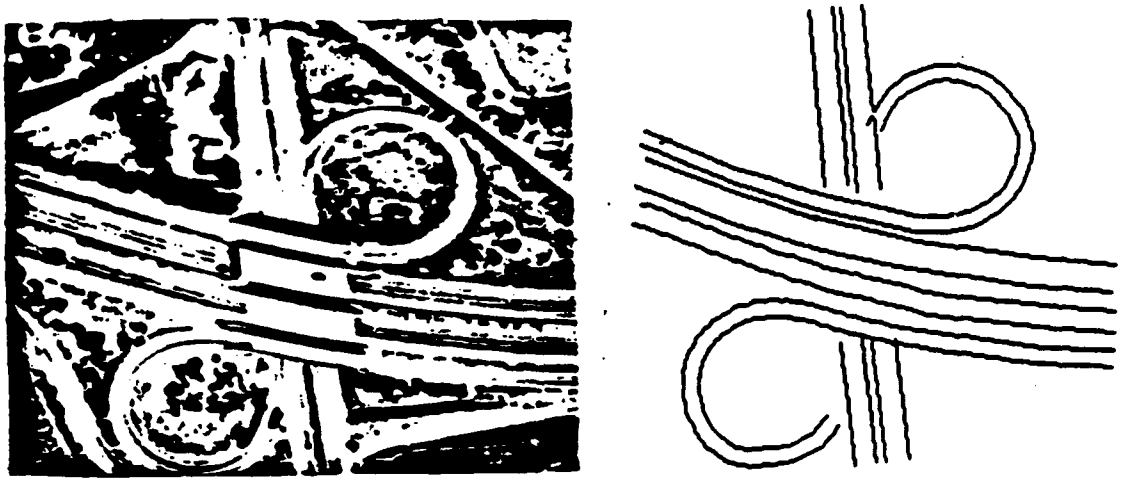


Figure 3: Example of Quam's correlation-based road tracker results.

a terminating pixel. The 2-D image array is considered as a graph in which each pixel is connected by a directed weighted arc to its eight immediately adjacent array neighbors. The pixels and arcs have an associated cost that reflects their local likelihood of belonging to the optimal path, i.e., the path with minimum cost. The track with the highest normalized ranking is selected as the primary road track through the given region. The starting pixel and terminating pixel, as well as a search region, can be selected interactively or from a map data base.

2.3 High-Resolution Road Tracking

In Quam's procedure [9] for tracking roads and detecting potential vehicles in aerial images, a context-adapting heuristic search method is used to support a dynamically changing model of road photometry. Figure 3 shows an example.

Successive road intensity cross-sections (RCS) taken perpendicular to the direction of the road show a high degree of correlation, which suggests that road tracking can be accomplished by using cross-correlation (template matching between the road cross-section intensities, and a road cross-section model). Deviations from the model indicate anomalous pixels such as road patches, road markings, occlusions, and vehicles. The location of the correlation peak was used to maintain alignment with the road center and to generate a model for the road trajectory. However, this approach turned out to be suboptimal because anomalies perturb the correlation peak causing small, cumulative alignment errors.

To overcome these problems, four refinements were introduced:

- Cumulative road cross-section model
- Trajectory extrapolation
- Anomaly detection

- Masked correlation

Instead of aligning consecutive RCSs, each RCS is aligned with a cumulative RCS model, based on an exponentially weighted history of previously aligned RCSs. Parabolic extrapolation of past correlation peaks is used to predict the future road trajectory. The predicted trajectory is used to guide the tracker past areas where the correlation peak is unsatisfactory. Anomalies are detected by comparing the aligned RCS with the RCS model. Corresponding pixels that significantly disagree are marked as potential anomalies. The cross-correlation is then repeated, masking out the anomalous pixels to obtain a more accurate alignment.

This algorithm uses a more complex model of road appearance than that employed by the low-resolution road tracker described in Section 2.2. This model allows it to successfully extract roads in high resolution photography in which low-resolution road trackers usually fail.

2.4 Optimization-Based Feature Extraction

In most images, object boundaries cannot be detected solely on the basis of their photometry because of the lack of a precise object model, the presence of unknown objects, and the presence of various photometric anomalies. Thus, all methods for finding boundaries based on purely local statistical criteria are bound to make mistakes, finding either too many or too few edges (usually depending on the choice of arbitrary thresholds).

To supplement the weak and noisy local information, Fua and Leclerc consider the geometrical constraints that object models can provide [2]. A boundary is described as an elastic curve with a deformation energy derived from the geometrical constraints, as suggested by Kass *et.al.* [6]. Photometric constraints are incorporated by defining photometric energy as the average of the edge strengths along the curve. Local minima in this energy correspond to boundaries that best match the photometric model. A candidate boundary can be found by deforming the curve in such a way as to minimize its total energy, which is the sum of the deformation and photometric energies. Once a curve has been optimized, i.e., once it has settled in a local minimum of the total energy (which is, in effect, a compromise between the two constraints) more detailed object models can be used to determine if the curve actually corresponds to an object boundary.

Such energy minimizing curves, sometimes called "snakes," have two key advantages:

- The geometric constraints are used directly to guide the search for a boundary.
- The edge information is integrated along the entire length of the curve providing a large support basis without including the irrelevant information off the curve.

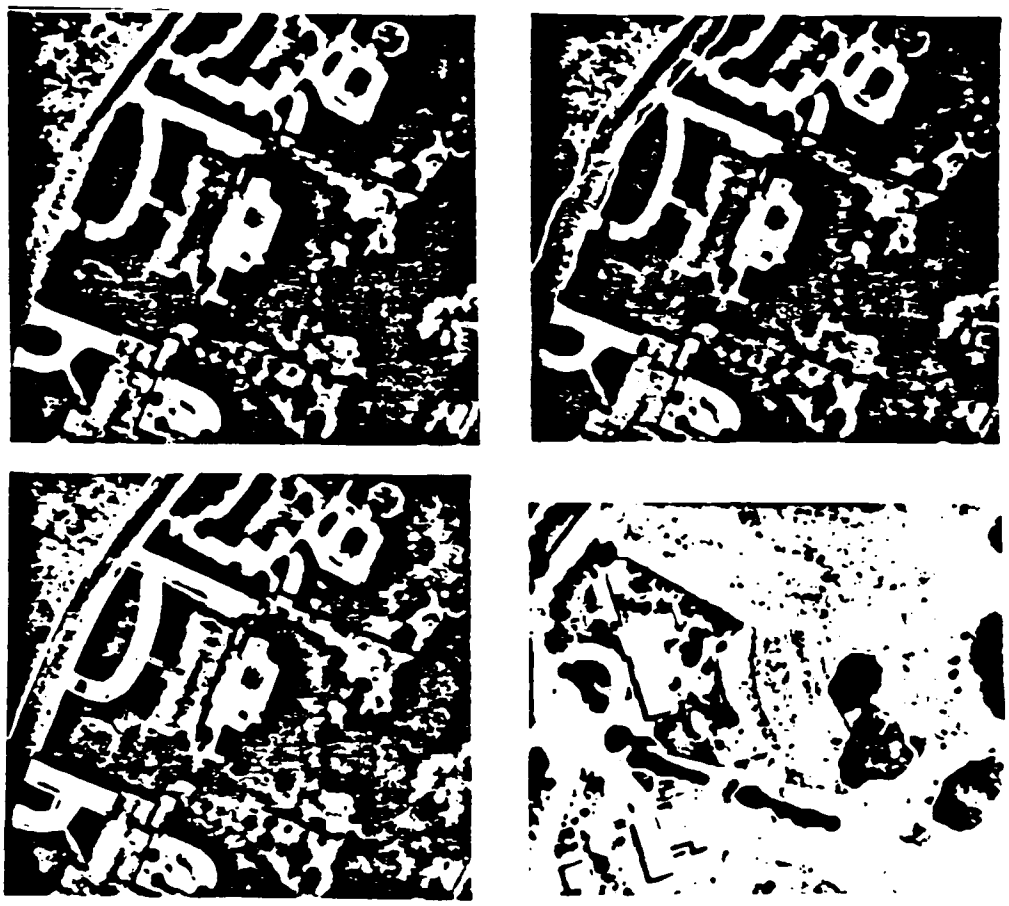


Figure 4: Example of the use of energy minimizing curves. (a) Aerial photo of a suburban scene. (b) Road boundary computed by Quam's correlation-based road tracker. (c) Optimized road boundaries using (b) as one of the initial conditions. (d) Vegetation boundaries extracted by closed-curve snakes.

Taken together, these advantages allow energy minimizing snakes to find photometrically weak boundaries that local edge detectors simply could not find without also finding many irrelevant boundaries.

This approach is used to extract a variety of curvilinear features, including roads (as illustrated in Figure 4c), sidewalks, rivers, and skylines. A variant of the approach utilizing closed curves can be employed to extract closed regions, such as vegetation boundaries (as in Figure 4d), lakes, and rooftops.

Techniques of this type can be effectively used to improve the precision and reduce the tedium associated with manual boundary extraction, by allowing the human to coarsely delimit a boundary using a minimum number of points, and allowing the optimization procedure to refine it.

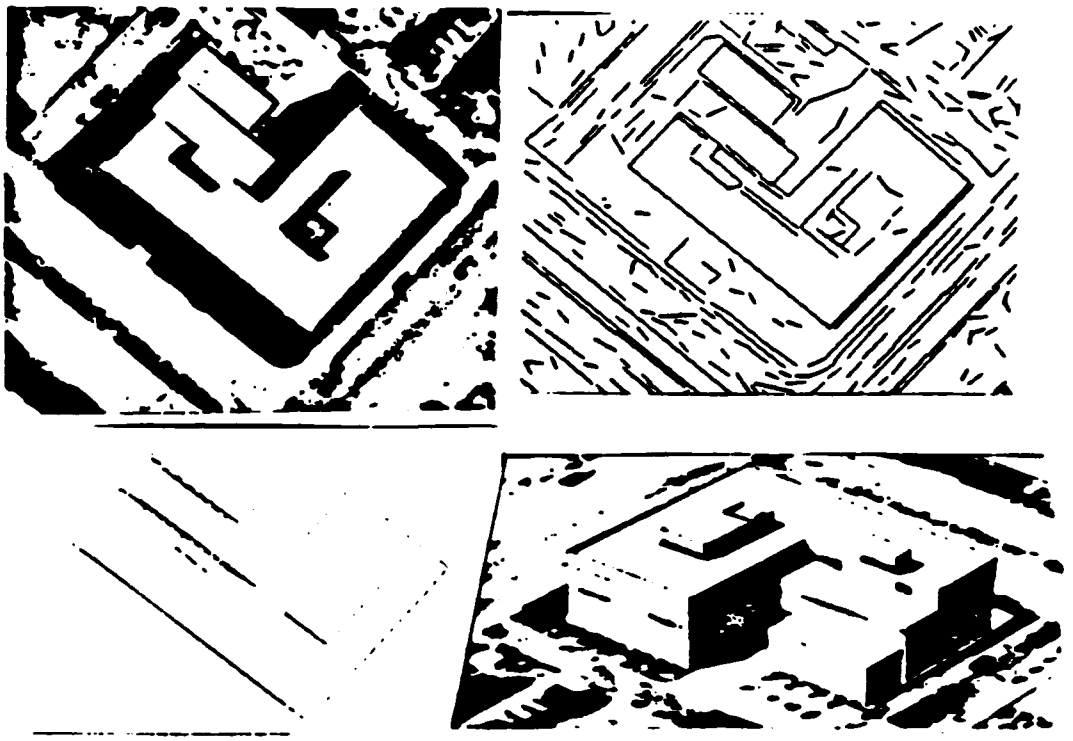


Figure 5: Example of building extraction. (a) Aerial photo of an urban scene. (b) Edges have either of two principle orientations. (c) Wire-frame model constructed semi-automatically. (d) Synthetic scene generated from (c).

2.5 Building Extraction

Automatic recognition and delineation of important cartographic objects, such as man-made structures, from aerial imagery is addressed by Fua and Hanson [3]. The basis for their approach is a theoretical formulation of object delineation as an optimization problem; practical objective measures are introduced that discriminate among a multitude of object candidates using a model language and the minimal encoding principle, MDL. This approach is then applied in two distinct ways to the extraction of buildings from aerial imagery: the first is an operator-guided procedure that uses a massively parallel Connection Machine implementation of the objective measure to discover a building in real time given only a crude sketch [4]. The second is an automated hypothesis generator that employs the objective measure during various steps in the hypothesis-generation procedure.

As described by Suetens, *et.al.*, [11]:

> To generate optimal descriptions, a hierarchical procedure carries out the following steps: (1) Extract edges with the appropriate geometry; (2) Find elementary geometric relationships between edges (such as corners or parallels); (3) Build closed cycles of related edges that enclose areas with acceptable photometric and geometric properties; (4) Invoke a contour completion procedure that generates closed contours, optimizes

their location and computes their elevation. (5) Select the highest scoring contours.

Each parsing step is designed as a filtering process that both enforces some model constraints and limits the size of the search space, thereby preventing combinatorial explosion of the search. Multiple knowledge sources (edge data, interior pixel intensities, stereographic information, shadow information, and other geometric constraints) are combined to build and rank hypotheses for generic objects of arbitrary complexity, such as the one shown in Figure 5.

The building model employed makes use of implicit constraints on the perpendicularity of roof edges, verticality of walls, and the homogeneity of regions corresponding to rooftops. These attributes are best expressed in terms of constraints on the 3D geometry of the building rather than constraints on the 2D geometry in the perspective projection of the building seen in the image. The Cartographic Modeling Environment, with its perspective camera models and accompanying geometric transformations allows these 3D constraints to be enforced with a minimal burden on the programmer.

3 Cartographic Modeling Environment

The SRI Cartographic Modeling Environment (CME) has been developed to support research and software development of interactive, semiautomated, and automated cartographic feature-extraction techniques [5, 10]. By carefully integrating image processing, photogrammetry, 3D computer aided design (CAD) modeling, and 3D computer graphics technologies, CME provides a rich programming environment for experimentation and prototype development. CME supports a variety of interactive facilities for creating, editing, viewing, and rendering three-dimensional models of physical objects, cartographic features, and terrain. These modeling capabilities may be used independently or in conjunction with multiple, photogrammetrically calibrated digital images. Interaction with geometric models is characterized by intuitive simplicity and by innovative techniques for exploiting geometric and data-driven constraints in the manipulation process. Synthetic views of a scene may be constructed from arbitrary viewpoints using terrain and feature models in combination with photo texturing using photogrammetrically registered imagery. Examples of the types of imagery, terrain, and object models represented and manipulated by CME are illustrated in Figure 6.

CME extends the two-dimensional capabilities of the ImagCalc^{TM} image manipulation system to the three-dimensional domain of the real world. The geometric parameters of terrain, cartographic features, and camera models are defined in a local rectangular coordinate system which is usually tied to a reference geoid such as Clarke 1866 or WGS84. Each image has an associated camera model that defines

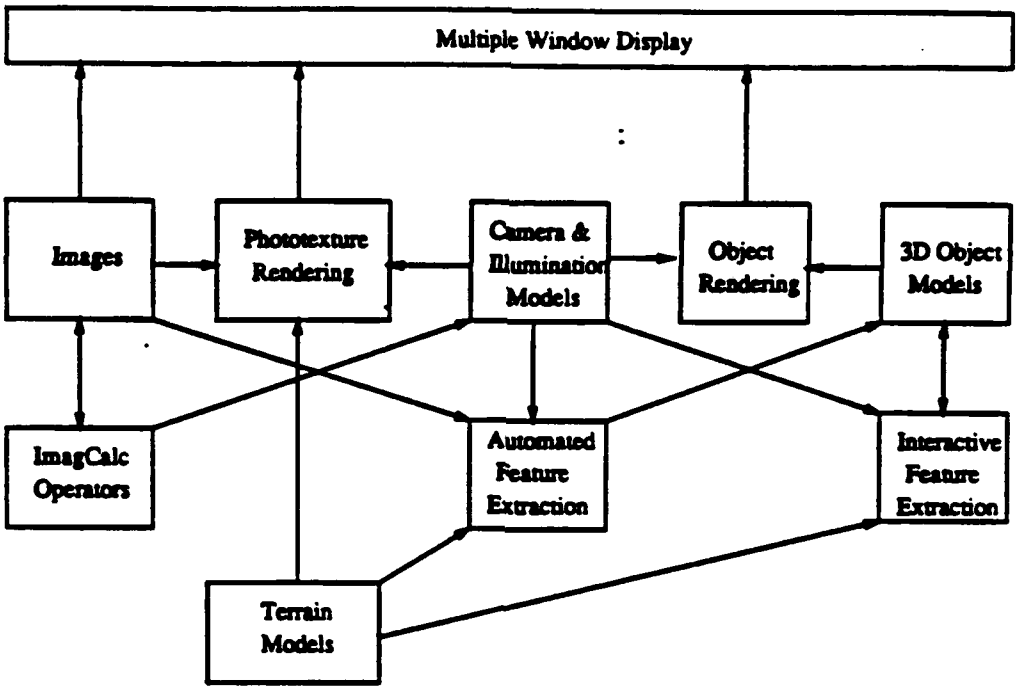


Figure 6: Block diagram of the major components of the SRI prototype cartographic analysis and display system.

the projection from 3D world coordinates to 2D image coordinates. The inverse projection from image coordinates to world coordinates is accomplished by intersecting rays from the camera with 3D terrain and feature models. Conventional stereo triangulation is implemented simply by intersecting the camera rays of corresponding conjugate points. All ImagCalc operators propagate their geometric transformations to the camera models of their result images.

The distinguishing features that set the Cartographic Modeling Environment apart from more conventional CAD systems include:

- Registration of multiple data sources, including stereographic and non-stereographic images, terrain elevation models, and three-dimensional object models, to the same world coordinate system. This capability is unique in that it permits object model entry to be driven by *sensor* data such as actual images.
- A variety of camera models appropriate to both conventional frame cameras, and to satellite imaging systems such as SPOT.
- Use of lighting models, terrain elevation data, and other geometric knowledge to constrain and facilitate data entry. The exploitation of constraints in the interactive modeling process increases the efficiency of the human operator.
- Registration of local coordinate systems to UTM, latitude-longitude, and other

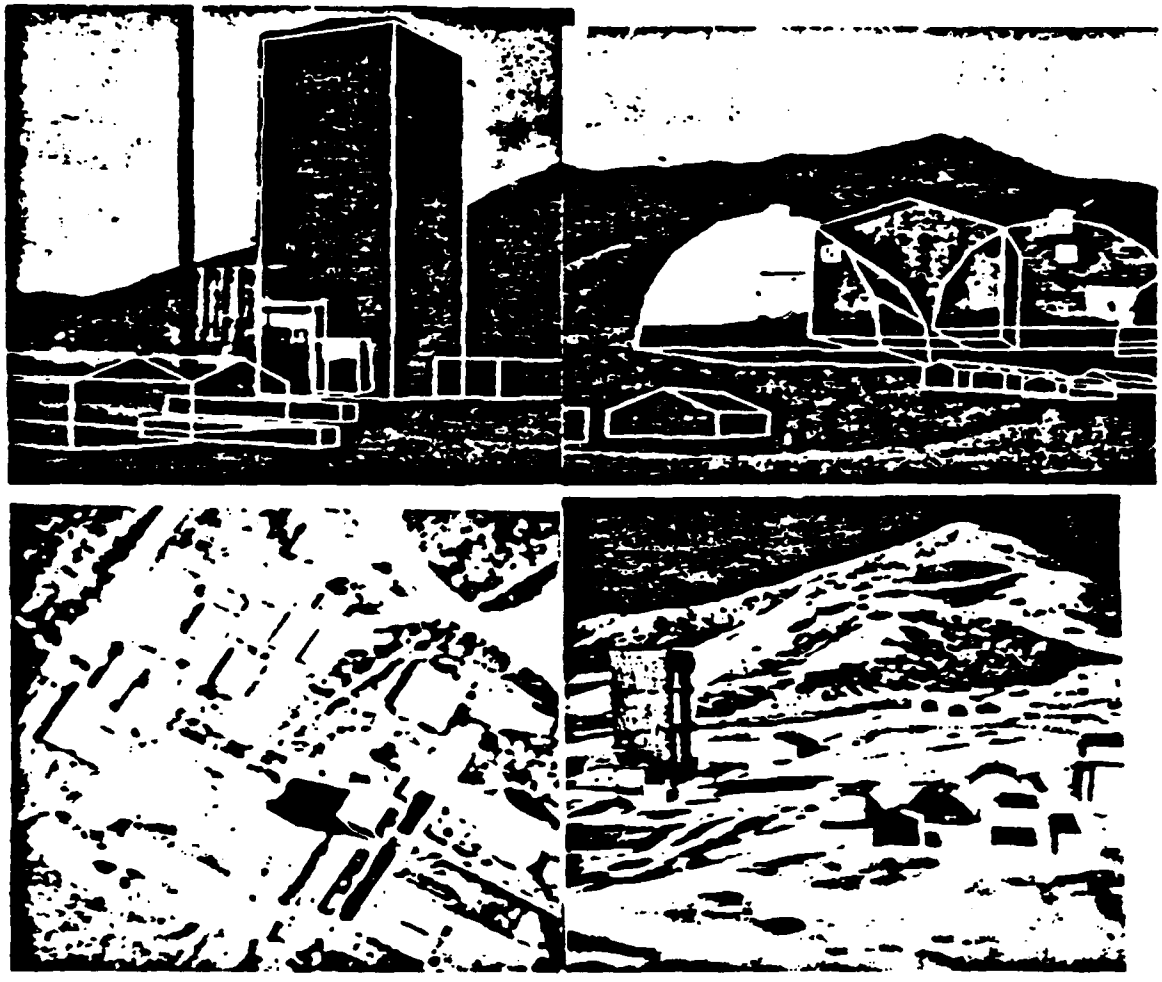


Figure 7: An example of the CME screen showing wire-frame models superimposed on both real and synthetic imagery.

cartographic coordinate representations. The use of real-world coordinate systems enables the system to exploit specific world knowledge, e.g., by computing the sun position for a particular location at a particular time of day.

- Phototexture rendering facilities to synthesize imagery from other perspectives and to generate synthetic image sequences (movies). Synthetic image generation is an essential means for verifying the correctness of extracted features.
- An interactive framework to support computer assisted feature extraction for cartographic applications.

4 Summary

Although fully automated digital cartography is far from reality, semi-automated techniques promise great increases in throughput and accuracy in the construction of cartographic databases from aerial photography. We have described five examples of techniques that appear well-suited for inclusion in an advanced cartographic workstation. They differ to the degree to which they rely upon models of the features they attempt to extract — generally, approaches that employ strong models have limited application, while those employing weaker models are more widely applicable but often less robust. A cooperative partnership in which the human photo-interpreter controls the invocation of suitable algorithms, and in which the advanced workstation provides facilities to evaluate the results through visualization and to edit the results through direct manipulation, offers the greatest payoff in the near term.

The SRI Cartographic Modeling Environment has been designed to support the interactive construction and use of 3D cartographic databases from both aerial and ground-level imagery. Its extensive collection of editing and display tools facilitate experimentation during algorithm development as well as the construction of more specialized photo-interpretation systems.

5 Acknowledgment

This paper includes excerpts of system descriptions provided by other authors which have been reprinted or paraphrased here with their permission. In particular, we thank Marty Fischler, Pascal Fua, Andy Hanson, Yvan Leclerc, Paul Suetens, and Helen Wolf for their contributions.

References

[1] M. A. Fischler, J. M. Tenenbaum, H. C. Wolf, "Detection of Roads and Linear Structures in Low Resolution Aerial Imagery Using a Multisource Knowledge Integration Technique," *Computer Graphics and Image Processing*, Vol 15, pp 201-223, 1981.

[2] P. Fua, Y. G. Leclerc, "Model Driven Edge Detection," *Proc. of the DARPA Image Understanding Workshop*, pp 1016-1021, April 1988.

[3] P. Fua, A. J. Hanson, "Objective Functions for Feature Discrimination: Applications to Semiautomated and Automated Feature Extraction," *Proc. of the DARPA Image Understanding Workshop*, pp 677-690, May 1989.

[4] P. Fua, "Object Delineation as an Optimization Problem: A Connection Machine Implementation," *Proc. of Fourth Int. Conf. of Supercomputing*, Santa Clara, CA, pp. 476-484, 1989.

[5] A. J. Hanson, L. H. Quam, "Overview of the SRI Cartographic Modeling Environment," *Proc. of the DARPA Image Understanding Workshop*, pp 576-582, April 1988.

[6] M. Kass, A. Witkin, D. Terzopolius, "Snakes: Active Contour Models'," *Int. J. Computer Vision*, 1(4), pp. 321-331, 1987.

[7] Y. G. Leclerc, "Constructing Simple Stable Descriptions for Image Partitioning," *International Journal of Computer Vision*, Vol 3, No 1, pp 73-102, May 1989.

[8] Y. G. Leclerc, "Region Grouping using the Minimum Description Length Principal," *Proc. of the DARPA Image Understanding Workshop*, pp. 473-479, Sept 1990.

[9] L. H. Quam, "Road Tracking and Anomaly Detection in Aerial Imagery," SRI International AI Center Technical Note 158, *Proc. of the DARPA Image Understanding Workshop*, pp 51-55, May 1978.

[10] L. H. Quam, "The Terrain Calc System," *Proc. of the DARPA Image Understanding Workshop*, pp 327-330, Dec 1985.

[11] P. Suetens, P. Fua, and A. J. Hanson, "Computational Strategies for Object Recognition," unpublished, Sept 1989.