Image Understanding Research at Colorado State University *

Bruce A. Draper J. Ross Beveridge
Computer Science Department
Colorado State University
Fort Collins, CO 80523
draper/ross@cs.colostate.edu

Abstract

Colorado State University is initiating a new project on learning control strategies for object recognition. It is our belief that the current library of IU algorithms is sufficient for solving many practical tasks if we can only learn to sequence them properly. We are investigating the use of open-loop and closed-loop control policies for sequencing IU algorithms, emphasizing the use of Markov decision models and reinforcement learning to derive closed-loop object recognition policies. This work is being conducted in the context of the Automatic Population of Geospatial Databases (APGD) project, where it will be used to learn object recognition strategies for finding buildings, roads and other objects of interest in aerial images.

1 Introduction

Image understanding (IU) research at Colorado State University (CSU) is based on two simple premises:

- IU researchers have made tremendous progress in developing algorithms for many aspects of the computer vision problem, including feature extraction, shape reconstruction and model matching.
- Despite this progress, there are very few practical IU systems because they are too difficult to build and too brittle with respect to changes in the domain or task statement.

Based on these observations, researchers at CSU are developing recognition strategies for the DARPA Automatic Population of Geospatial Databases (APGD) program based on a new approach. Instead of hand developing new algorithms for specific IU tasks, we approach IU as a control problem. Our hypothesis is that many practical IU problems can be solved through existing techniques, provided we can learn to select the proper sequence of algorithms. Our goal is to develop the technology for learning to recognize objects from examples by training control strategies that select sequences of IU algorithms based on the object to be recognized and the domain context.

2 Background

Over the last twenty years, computer vision researchers have divided the field into ten or twenty (or more) topics, each with a narrowly-defined problem focus. Within these subfields, theories have been developed and specific algorithms have been proposed. As a result, there are now good and improving algorithms for feature extraction (including points, edges, lines, curves and regions), stereo analysis, multi-frame feature tracking, depth from motion (two-frame and multi-frame), shape matching, color matching, pixel matching (a.k.a. appearance matching), and 3D pose determination, and new algo-

*This work was sponsored by the Defense Advanced Research Projects Agency (DARPA) Image Understanding Program under contract 96-14-201 monitored by the Army Topographic Engineering Laboratory (TEC), and contracts DAAH04-93-G-422 and DAAH04-95-1-0447, monitored by the U. S. Army Research Office

rithms are being developed as this is written.

At the same time, other researchers (particularly within the DARPA community) have concentrated on building end-to-end systems that solve specific IU tasks. For example, in recent years competing systems have been developed for recognizing and reconstructing buildings from aerial images [McGlone & Shufelt, 1993; Lin *et al.*, 1994; Jaynes *et al.*, 1996], while other systems have been built for recognizing military targets, road networks and terrain features. These systems can be viewed as the intellectual descendents of earlier knowledge-based systems that exploited knowledge about objects and domains to create special-purpose object recognition strategies (e.g. [McKeown *et al.*, 1985; Hwang *et al.*, 1986; Draper *et al.*, 1989]).

If we take a close look at the recent work on building reconstruction [McGlone & Shufelt, 1993; Lin *et al.*, 1994; Jaynes *et al.*, 1996], we can draw several conclusions:

- These systems were built by sequencing standard IU algorithms for line extraction, line grouping, shadow analysis and graph matching/traversal. (Some of these systems refined previous algorithms in these areas.)
- These systems are special-purpose to the extent that they reconstruct a single type of object within a specific domain. They will not work if the target object class or domain is changed significantly.
- It is difficult to analyze these systems. There is no underlying theory by which to gauge their performance, nor is there an analytical method for predicting at what point they will break. It is difficult even to compare them, since each system makes slightly different assumptions about the imagery.
- These systems are difficult to build. Each system took months or even years of highly skilled labor to construct, and even so most of these systems could be improved given more time.

Our project is an effort to generalize from earlier object recognition efforts. Like these previous systems, our goal is to sequence existing IU algorithms in order to recognize specific objects within limited contexts. Unlike previous efforts, we will neither construct these algorithm sequences by hand nor build a "knowledge base" with rules for selecting algorithms (such knowledge bases have proven error-prone and difficult to build in the past; see [Draper & Hanson, 1991]). Instead, we will model IU as a control problem, in which the goal is to select the best sequence of algorithms for recognizing a given object class. More specifically, we model the IU control problem as a Markov decision process, in which the goal is to train a control policy that selects algorithms so as to maximize the expected reward over time, where the reward is a weighted function of cost versus accuracy. Our aim is a system that is general-purpose and robust in the sense that it can be retrained to recognize a wide variety of objects in various domains, and theoretically sound in the sense that it will converge to the optimal control policy as the training set size increases.

3 Relevance to APGD

To the extent that we are successful, our technology for learning object recognition strategies could be used within the APGD context for context-based algorithm control. We imagine a system in which image analysts begin to populate a database by outlining and labeling objects of interest in images. As the analysts work, the system uses these labeled object instances as training samples for learning object recognition policies. When the strategy for an object class is trained, the system will take over for the analyst and automatically label any remaining instances of the object class and enter these instances into the geospatial database. Moreover, as new images are acquired the system will automatically label objects instances and update the database.

Through continued training, a system which learns strategies could in principle adapt to recognize any visually distinct object class. Moreover, because the system will be based on a Markov decision process model and reinforcement learning, there are analytical reasons to believe that the control policies it learns will be sound; this is not true of current hand-crafted systems. Finally, because the reinforcement learning process makes a series of predictions about the intermediate data it generates, the system should be able to detect when its control policies are failing (perhaps because of a new variant of an object class or because of a change in the domain) and to ask the image analyst for new training samples, rather than make possibly disastrous mistakes.

4 Image Understanding as a Control Problem

Underlying this work is the notion of IU as a control problem. In casting IU as a control problem, we make the assumption that there is a library of available IU algorithms (such as the ones mentioned in Section 2), and that these algorithms are sufficient to solve many interesting IU problems. We also assume that the system is given an IU task of the form "Find the X in Y", where X is an object to be recognized and Y is a set of images; examples of X in the APGD context include buildings, roads, trees and power lines, while an example of Y might be EO nadir-view images of a constrained geographic area such as Bosnia or Somalia. The challenge for the control system is to find a sequence of algorithms (ideally, the best sequence of algorithms) for finding the target object in the given domain.

Readers should note that we propose to control algorithms, not physical devices such as cameras or platforms. Other researchers have addressed physical camera control in the context of active vision research. Also, there is a harder version of the IU problem in which the object being searched for is unspecified (i.e. "What's in the image, and where is it?"). This form of the problem includes the object indexing problem, which is outside of the scope of this work.

4.1 Open-. vs. Closed-loop Control

Once the decision is made to cast IU as a control problem, several interesting questions emerge. The first of these questions is whether the system is an open-loop or closed-loop control system. In open-loop control, the system selects a fixed sequence of actions for each task. Open-loop control systems have the advantage that recognition strategies can be easily expressed as sequences of algorithms, and open-loop policies can be learned by searching the space of algorithm sequences; Brown & Roberts [1994], for example, use genetic algorithms to search for the best sequence of algorithms for a specific automatic target recognition (ATR) task.

On the other hand, open-loop control systems have the disadvantage that they are unable to adjust to unexpected events. For example, one strategy for finding buildings in aerial images might be to first extract building corners, where the camera viewpoint determines the expected image angle. Unfortunately, if the corner detector fails in an open-loop system (perhaps because of an error in the estimated viewpoint) the open-loop strategy will continue to apply the remaining algorithms in the sequence, even though the data produced by the first step was erroneous. Closed-loop systems, on the other hand, do not produce explicit sequences of actions. Instead, they select an algorithm at each stage of the process based on the results of the previous processing step. This gives closed-loop systems the ability to react to unexpected events during processing, for example by backtracking and selecting another algorithm.

More formally, closed-loop control strategies are defined by *policies* that map states of the system onto actions (i.e. algorithms), where system states are determined by measuring feature attributes. Closed-loop control is more powerful than open-loop control, and it is our belief that variations among images within a domain and the inherent unreliability of many IU algorithms imply that closed-loop policies will be needed for robust control. This is still a hypothesis, however, and one of our tasks in this project will be to compare closed-loop strategies learned through reinforcement learning with open-loop strategies learned through search.

4.2 Expert Knowledge vs. Machine Learning

Another question is whether control decisions come from reasoning about expert knowledge or whether they are learned automatically from experience. We believe that although many variations on expert systems have been tried for IU (e.g. SPAM [McKeown *et al.*, 1985], SIGMA [Hwang *et al.*, 1986], PSEIKI [Andress & Kak, 1988], the Schema System [Draper *et al.*, 1989] and more recently OCAPI [Clement & Thonnat, 1993]), they have always proven to be difficult to build and harder to extend. Moreover, even when they produce acceptable results on a limited set of images, it is difficult to tell whether the control system has performed well or not. We base our conclusions in part on our own past work. In [Draper & Hanson, 1991] we used the Schema System to illustrate problems inherent in the hand-construction of expert knowledge bases for IU, while in [Draper *et al.*, 1996] we give a broader scope to these issues.

The alternative is to build a system that learns control strategies based on examples. Brown & Roberts [1994] is one example of a system that learns open-loop control policies bases on examples ([Chen & Mulgaonkar, 1992] is another). We propose to build on our earlier work [Draper, 1996a; Draper, 1996b] by building a system that uses reinforcement learning to automatically acquire closed-loop control policies from examples. (In related but different work, [Peng & Bhanu, 1996] used reinforcement learning to select parameters for IU algorithms.).

It is our contention that in the long run machine learning is the best source of control strategies. A fielded APGD system, for example, must be able to adapt to new object classes in new domains. A system requiring expert modification and recertification for each new object or domain is not acceptable to the military (or indeed to any user). Instead, it is our goal to show that robust closed-loop control policies for object recognition can be learned by observing an expert.

4.3 General-purpose vs. Object-specific Attributes

A critical issue for closed-loop control systems is generating object-specification attribute measures for intermediate data representations. In a closed-loop system, a control policy selects actions (i.e. algorithms) based on the current state of the system. The system state in turn is a reflection of attributes that can be measured for the features that have been extracted up to that point. For example, in the hypothetical closed-loop control policy for recognizing buildings mentioned in Section 4.1, the first action was to extract corners from the image data. The second action was then selected based on the number and quality of corners found in the first step.

Clearly, closed-loop control policies can only outperform open-loop action sequences if the attributes of the image features provide meaningful feedback. In earlier experiments on learning to recognize buildings we provided a system with routines for computing sophisticated feature attributes, including an algorithm which measured how much of a shadow a feature cast (based on the known camera viewpoint). This attribute proved to be critical; as reported in [Draper, 1996a] (page 1453), the number of false alarms detected by the trained system dropped significantly when the shadow attribute was introduced.

In this project, we intend to have the system develop meaningful attributes on its own. Some of these attributes will be learned, while others will be deduced from *a priori* models. Attributes may be derived from many levels of representation, including image properties, such as color and texture, and object geometry. For instance, we will extend our earlier work with linear machine decision trees to learn the apparent color of objects in outdoor imagery [Buluswar & Draper, 1994] to train attributes that match image features to the expected textures (and if available, colors) of objects. We will also build on current work for matching features derived from geometric object models. Prior examples include our past work on matching horizons derived from terrain maps to imagery [Beveridge & Balasubramanian, 1997] and our multi-sensor system for predicting observable features of CAD vehicle models [Stevens & Beveridge, 1997]. The specific feature sets developed in these examples do not carry over directly to the APGD domain. However, the principles employed by the algorithms generating these features are applicable to APGD. We will also be expanding our work on a set of algorithms for learning probe sets and/or eigenvalue measures from example features extracted from images [Stevens *et al.*, 1997].

5 Evaluating Recognition Strategies

In order to evaluate our ability to learn closed-loop object recognition policies, we will apply our system to the APGD Fort Hood dataset[1] and test its ability to recognize objects of strategic interest. In particular, we will begin by training recognition policies to find buildings and roads. Then we will test how easily the system can adapt to new tasks by training it to recognize two more object classes, to be determined jointly by ourselves and representatives of the Army Topographic Engineering Center (TEC). Finally, we will adopt a new dataset from a different domain to see how easily the system adapts from one setting to another.

In general, when evaluating our system, the quality of a control policy will be measured by a utility function that balances accuracy and cost. (The relative weight of accuracy vs. cost is determined by the user prior to training.) To measure the effectiveness of the learning system, however, we must seperate the performance of the control policy from the quality of the underlying IU algorithms. To do this, we will compare control policies against two standards. The first standard is the result of an exhaustive search of the space of open-loop strategies. (We can compute this because the space of open-loop strategies is much smaller than the space of closed-loop policies.) Closed-loop policies trained through reinforcement learning will then be compared to the optimal open-loop strategy according to the user-defined utility function. Second, we will compare closed-loop policies to each other. Although there is no way to know what the true optimal closed-loop policy for a given task may be, if we train multiple closed-loop policies we can compare them to each other, determining which are the best (and by how much).

[1]For readers unfamiliar with the Fort Hood data, it is a collection of approximately twenty high-resolution black-and-white aerial images of Fort Hood, TX, including approximate camera parameters for each image.

6 Practicum: Khoros and the IUE

At a more mundane level, work on IU as a control problem can only proceed if libraries of IU algorithms are accessible. One of the goals of the Image Understanding Environment (IUE) [Mundy *et al.*, 1992] is to disseminate libraries of IU "task" objects, which are implementations of IU algorithms. To this end, we have been actively contributing to the IUE effort. Our most recent contribution is a target detection algorithm for use in IR imagery. This algorithm was selected as an archetypical ATR algorithm and it is based loosely on the concepts of a sliding window detector set out by Nguyen [Nguyen, 1990]; it is of practical interest because it is used as the first phase of a two phase target detection algorithm on the Unmanned Ground Vehicle Program's Semi-Autonomous Scout Vehicles. We are also developing a version of our optimal line segment matching system for release with the IUE.

At the same time, we are forced to recognize the limited state of the current IUE task library. We will therefore be developing our learning system within the Khoros image processing environment [Rasure & Kubica, 1994] which at the moment has a more extensive library of (mostly low-level) computer vision algorithms, and we will be extending this library as necessary. Fortunately, long-term plans call for the IUE to become compatible with Khoros, and it is our hope to be able to access both the IUE and Khoros task libraries in the relatively near future.

7 Recent Accomplishments

Although the main thrust of this paper is to look forward to our project on learning object recognition policies, we thought it would be useful to briefly summarize some of our recent accomplishments, and in particular to expose some underlying intellectual connections between our previous work and our new project.

7.1 Automatic Target Recognition

Over the past three years we have devoted considerable attention to the development of

Table 1: Confusion matrix for Multisensor Target Identification. Correct identification rate is 27/35 (77%). The two entries marked with '*' are cases where hypothesis generation failed to suggest the correct target type.

		Multisensor System ID			
		M113	M901	M60	Pickup
True Target ID	M113	7		1	1
	M901	1*	5	2	1*
	M60		1	7	1
	Pickup				8

new model-based ATR techniques for multisensor imagery. This work was supported by the DARPA IU Program as part of the Unmanned Ground Vehicle (UGV) program's RSTA (Reconnaissance, Surveillance and Target Acquisition) activity. A detailed report on this effort appears in [Beveridge *et al.*, 1997a]; Here we discuss only those portions of the work that are relevant to the APGD project.

The CSU ATR system was a three-stage target detection and recognition system that performed well on the difficult Ft. Carson data set [Beveridge *et al.*, 1994]. On 35 target identification tasks involving 4 targets, the CSU system correctly identified 27 out of 35 (77%) of the targets. If we neglect difficult cases, such as distant and occluded targets, the correct identification rate is over 90%. The confusion matrix summarizing this result is presented in Table 1.

As a point of comparison, the group from MIT Lincoln Laboratory has used the Ft. Carson dataset in part of the evaluation of their own range-based ATR system [Verly & Lacoss, 1997]. Based upon their performance modeling work, they conclude their approach is only applicable to four of the highest resolution Ft. Carson range images.

Although some aspects of the CSU ATR system are tailored to ATR, some of the technology is applicable to the APGD task. The first stage of the ATR system used a linear machine decision tree to learn the range of apparent colors exhibited by an object under outdoor lighting conditions [Buluswar & Draper, 1994]. Although the Ft. Hood dataset used for the current APGD work does not include color data, this same learning technique can be used to learn combinations of texture measures extracted from black and white imagery. How well an image feature matches the expected texture of an object then becomes an attribute that a closed-loop control system can use for feedback.

The second stage of the CSU ATR system used probing techniques to suggest possible targets and target orientations. The probing techniques derived probe sets from BRL/CAD models of possible targets, and we developed new techniques based on neural networks for efficiently selecting the most relevant probesets [Stevens *et al.*, 1997]. Once again, although the objects being searched for in the APGD domain will be different, probing techniques such as these can be used to develop object-specific feature attributes whenever either object models or substantial training imagery are available. We have also begun to explore the intellectual connections between probing and eigenspace analysis [Nayar *et al.*, 1996; Kirby & Sirovich, 1990; Turk & Pentland, 1991], and are looking for ways to train both probe sets and eigenspace representations from the APGD training samples.

Finally, the third stage of the CSU ATR system performed the final target identification and target pose determination by matching 3D target models to the multisensor image data. By exploiting an iterative predict-and-match cycle between the 3D object model and the multisensor image data, we have demonstrated what we consider to be several significant advances in the state-of-the-art for ground-based multisensor ATR. We have demonstrated an ability to take a rough target pose estimate, i.e. off by as much as 30°, and generate a more reliable estimate accurate to within about 5° [Stevens & Beveridge, 1997]. Further, this is done in the presence of errors in the initial registration mappings between sensors: our algorithm refines sensor registration and 3D target pose as part of the matching process.

Although this work might at first seem unrelated to the current APGD effort, this work implies that our learning systems will have access to state-of-the-art algorithms for matching geometric models to data. Moreover, it demonstrates an ability to propagate evidence for terrain occlusion between sensors and accordingly modify range, IR and color target features during the matching cycle. This means that the CSU ATR system does not try to find features which it can infer are occluded by foreground terrain. Such an ability to reason about why certain features might not be seen will be critical to control systems that must distinguish between features that are missing (or may have changed), and those which simply cannot be seen from the current viewpoint.

8 Conclusion

The current focus of IU research at Colorado State University is on understanding the implications of modeling IU as a control problem, and on building practical systems that learn object recognition strategies from examples. This work is being conducted in the context of the Automatic Population of Geospatial Databases (APGD) project, where it will be used to learn object recognition strategies for finding buildings, roads and other objects of interest in aerial images.

References

[Andress & Kak, 1988] Andress, K.M. and Kak, A.C. "Evidence Accumulation and Flow of Control in a Hierarchical Reasoning System," *AI Magazine* 9(2):75–94 (1988).

[Beveridge & Balasubramanian, 1997] R. Beveridge and K. Balasubramanian. "Camera Orientation Refinement by Matching of Terrain Map Horizons to Ground-looking Imagery," IUW 1997..

[Beveridge *et al.*, 1994] R. Beveridge, D. Panda, and T. Yachik. November 1993 Fort Carson RSTA Data Collection Final Report. Technical Report CSS-94-118, Colorado State University, Fort Collins, CO, January 1994.

[Beveridge *et al.*, 1996] R. Beveridge, C. Graves and C. Lesher. Local Search as a Tool for Horizon Line Matching. In *Proceedings: Image Understanding Workshop*, pages 683 – 686, Los Altos, CA, February 1996. ARPA, Morgan Kaufmann.

[Beveridge *et al.*, 1997a] R. Beveridge, B. Draper, M. Stevens, K. Siejko and A. Hanson. A Coregistration Approach to Multisensor Target Recognition with Extensions to Exploit Digital Elevation Map Data. In Oscar Firschein, editor, *Reconnaisance, Surveilance, and Target Acquisition for the Unmanned Ground Vehicle*, page (to appear). Morgan Kaufmann, 1997.

[Beveridge *et al.*, 1997b] R. Beveridge, C. Graves and J. Steinborn. Comparing Random-Starts Local Search with Key-Feature matching. In *Proc. 1997 International Joint Conference on Artificial Intelligence*, page (submitted), August 1997.

[Bienstock *et al.*, 1990] E. Bienstock, D. Geman, S. Geman, and D. McClure. Development of laser radar atr algorithms: Phase II - Military Objects. Technical report, Mathematical Technologies Inc., Providence, Rhode Island, October 1990. Prepared under Harry Diamond Laboratories Contract No. DAAL02-89-C-0081.

[Brown & Roberts, 1994] C. Brown and B. Roberts. "Adaptive Configuration and Control in an ATR System," *DARPA Image Understanding Workshop,* Monterey, CA. 1994, pp. 467-480.

[Buluswar & Draper, 1994] S. Buluswar and B. Draper. "Non-parametric Classification of Pixels of Varying Outdoor Illumination," *DARPA Image Understanding Workshop,* Monterey, CA. 1994, pp. 727-732.

[Chen & Mulgaonkar, 1992] C. Chen and P. Mulgaonkar. "Automatic Vision Programming," *CGVIP-IU,* 55(2):170-183, 1992.

[Clement & Thonnat, 1993] V. Clement and M. Thonnat. "A Knowledge-based Approach to Integration of Image Processing Procedures," *CGVIP-IU,* 57:166-184, 1993.

[Delanoy *et al.*, 1993] R. Delanoy, J. Verly and D. Dudgeon. Machine Intelligent Automatic Recognition of Critical Mobile Targets in Laser Radar Imagery. *The Lincoln Laboratory Journal,* 6(1):161–186, Spring 1993.

[Draper *et al.*, 1989] B. Draper, R. Collins, J. Brolio, A. Hanson and E. Riseman. "The Schema System," *International Journal of Computer Vision,* 2:209-250, 1989.

[Draper & Hanson, 1991] B. Draper and A. Hanson. "An Example of Learning in Knowledge-directed Vision," *Scandinavian Conference on Image Analysis,* Aalborg, DK, 1991. pp. 189-201.

[Draper, 1996a] B. Draper. "Learning Grouping Strategies for 2D and 3D Object Recognition," *DARPA Image Understanding Workshop,* Palm SPrings, CA, 1996. pp. 1447-1454.

[Draper, 1996b] B. Draper. "Object Recognition as a Markov Decision Process," *International Conference on Pattern Recognition,* Vienna, 1996. pp. 25-29, Vol 4.

[Draper *et al.*, 1996] B. Draper, A. Hanson and E. Riseman. "Knowledge-directed Vision: Control, Learning and Integration," *Proceedings of the IEEE*, 84(11):1625-1637, 1996.

[Gillberg *et al.*, 1990] J. Gillberg, R. Johnston, K. Siejko, and J. Lee. Laser Radar ATR Algorithms. Technical Report DAABO7-87-C-F109, Honeywell Systems and Research Center, Minnesota, May 1990.

[Goldberg *et al.*, 1993] D. Goldberg, K. Deb, H. Kargupta, and G. Harik. Rapid, accurate optimization of difficult problems using fast messy genetic algorithms. In Stephanie Forrest, editor, *Proc. 5th International Conference on Genetic Algorithms*, pages 56–64. Morgan-Kaufmann, 1993.

[Huttenlocher & Ullman, 1990] D. Huttenlocher and S. Ullman. Recognizing Solid Objects by Alignment with an Image. *International Journal of Computer Vision*, 5(2):195 – 212, November 1990.

[Hwang *et al.*, 1986] Hwang, V.S-S., Davis, L.S., and Matsuyama, T. "Hypothesis Integration in Image Understanding Systems," *Computer Vision, Graphics, and Image Processing,* 36:321–371 (1986).

[Jaynes *et al.*, 1996] C. Jaynes, F. Stolle, H. Schultz, R. Collins, A. Hanson and E. Riseman. Three-Dimensional Grouping and Information Fusion for Site Modeling from Aerial Images. *DARPA Image Understanding Workshop*, Palm Springs, CA 1997, pp. 479-480.

[Kirby & Sirovich, 1990] M. Kirby and L. Sirovich. Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(1):103 – 107, January 1990.

[Koenderink & van Doorn, 1979] J. J. Koenderink and A. J. van Doorn. The Internal Representation of Shape with Respect to Vision. *Biological Cybernetics*, 32:211–216, 1979.

[Lin *et al.*, 1994] C. Lin, A. Huertas and R. Nevatia. Detection of Buildings using Perceptual Groupings and Shadows, *International Conference on Computer Vision and Pattern Recognition*, 1994, pp. 62-69.

[McGlone & Shufelt, 1993] C. McGlone and J. Shufelt. "Incorporating Vanishing Point Geometry into a Building Extraction System," *DARPA Image Understanding Workshop,* Washington D.C., 1993, pp. 437-448.

[McKeown *et al.*, 1985] McKeown, D.M. Jr., Harvey, W.A., and McDermott, J. "Rule-Based Interpretation of Aerial Imagery," *IEEE Trans. on Pattern Analysis and Machine Intelligence,* 7(5):570–585 (1985).

[Mundy *et al.*, 1992] J. Mundy et al. "The Image Understanding Environments Program," *International Conference on Computer Vision and Pattern Recognition,* San Mateo, CA, 1992, pp. 185-214.

[Nayar *et al.*, 1996] S. Nayar, S. Nene and H. Murase. Real-Time 100 Object Recognition System. In *Proceedings of ARPA Image Understanding Workshop*. Morgan Kaufmann, 1996. http://www.cs.columbia.edu/CAVE/rt-sensors-systems.html.

[Nguyen, 1990] D. Nguyen. An iterative Technique for Target Detection and Segmentation in IR Imaging Systems. Technical Report November, (CECOM) Center for Night Vision and Electro-Optics, 1990.

[Olson, 1994] C. Olson. Time and space efficient pose clustering. In *CVPR94*, pages 251–258, 1994.

[Peng & Bhanu, 1996] J. Peng and B. Bhanu. "Closed-loop Object Recognition using Reinforcement Learning," *International Conference on Computer Vision and Pattern Recognition,* 1996, pp. 538-543.

[Rasure & Kubica, 1994] J. Rasure and S. Kubica. "The Khoros Application Development Environment," in *Experimental Environments for Computer Vision and Image Processing,* H. Christensen & J. Crowley (eds.), World Scientific Press, 1994.

[Rimey, 1995] R. Rimey. RSTA Sept94 Data Collection Final Report. Technical report, Martin Marietta Astronautics, Denver, CO, January 1995.

[Stevens & Beveridge, 1997] M. Stevens and R. Beveridge. Precise Matching of 3-D Target Models to Multisensor Data. *IEEE Transactions on Image Processing*, 6(1):126–142, January 1997.

[Stevens *et al.*, 1997] M. Stevens, C. Anderson and R. Beveridge. Efficient Indexing for Object Recognition Using Large Networks. In *Proc. 1997 IEEE International Conference on Neural Networks* , page (to appear), June 1997.

[Turk & Pentland, 1991] M. Turk and A. Pentland. Face Recognition Using Eigenfaces. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 586 – 591, June 1991.

[Verly & Lacoss, 1997] J. Verly and R. Lacoss. Automatic Target Recognition for LADAR imagery Using Functional Templates Derived From 3-D CAD Models. In Oscar Firschein, editor, *Reconnaissance, Surveillance, and Target Acquisition (RSTA) for the Unmanned Ground Vehicle.* Morgan Kaufmann, 1997.