Control in a 3D Reconstruction System using Selective Perception[*]

Maurício Marengoni, Allen Hanson, Shlomo Zilberstein, and Edward Riseman
Computer Science Department
University of Massachusetts
Amherst, MA 01003
marengon,hanson,shlomo,riseman@cs.umass.edu

Abstract

This paper presents a control structure for general purpose image understanding that addresses both the high level of uncertainty in local hypotheses and the computational complexity of image interpretation. The control of vision algorithms is performed by an independent subsystem that uses Bayesian networks and utility theory to compute the marginal value of information provided by alternative operators and selects the ones with the highest value. We have implemented and tested this control structure with several aerial image datasets. The results show that the knowledge base used by the system can be acquired using standard learning techniques and that the value-driven approach to the selection of vision algorithms leads to performance gains. Moreover, the modular system architecture simplifies the addition of both control knowledge and new vision algorithms.

1 Introduction

An Image Understanding (IU) system should be able to identify objects in 2D images and to build 3D relationships between objects in the scene and the viewer. A large number of image understanding systems developed so far are dedicated to aerial image interpretation. One of the problems with aerial image interpretation systems is the management of uncertainty. Uncertainty in this case arises from a variety of sources, such as the type of sensor, weather conditions, illumination conditions, season, random objects in the scene, and the inherent uncertainty in the definition of common objects.

Object recognition in aerial images is one important step towards 3D reconstruction of a scene, but automating the recognition process in a real world application is not an easy task. Consider the image tiles from aerial images presented in Figure 1. The tile on top contains a building, which is easy to identify by its door and rooftop. The recognition of the three objects marked in the bottom tile is not as simple, and more detailed comparisons and measurements may be required to identify them correctly.

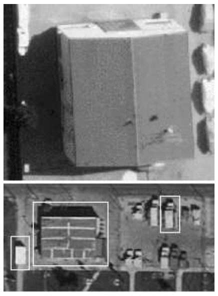

Figure 1: Different types of regions extracted from aerial images

Since an interpretation of an image can be viewed as a correspondence between image features and the identifying object classes, it is clear that the descriptive vocabulary of the system must be reflected in the set of features extractable from the image. Thus the image features must form the primitive descriptions of the objects in the knowledge base. Since every feature has at least one operator for measuring it, the control problem we address in this paper is this: given a general purpose system and a specific interpretation problem within the domain of the system, how do we effectively select the features to measure or, more generally, which algorithms to apply, and in what order. Furthermore, because there is a significant amount of inherent ambiguity in the interpretation process, an interpretation system must include a sufficiently rich set of relations among features as well as flexible mechanisms for manipulating uncertain hypotheses until there is a convergence of evidence.

In this paper we show how to use Bayesian net-

[*] Funded by the National Council for Scientific Research-CNPq, Brazil grant number 260185/92.2, by the APGD-DARPA project contract number DACA76-97-K-0005, and by Army Research Office, contract number DAAG55-97-1-0188

works and utility theory to build a control structure for a general purpose image understanding system. We also address the knowledge engineering issue by demonstrating that it is possible to learn the Bayesian network structures from fairly coarse training information. Ascender II, an IU system for fully automated Aerial Image Interpretation, is used as a testbed to address these questions:

- How can the results of a visual operators and their associated uncertainties be combined in order to classify a particular image region?
- How can the hierarchical structure of objects be exploited in order to construct an incremental classification process?
- Can the construction of the knowledge base be simplified (or fully automated) for a particular application using both human expertise and machine learning techniques?
- Can performance be improved by using a disciplined approach to operator selection?

The next section presents an abbreviated summary of related work previous work. Section 3 introduces the Ascender II system and presents its control structures, specifically how operators are ordered given the current knowledge. Section 4 shows how to learn the structures used for control. Experimental results are presented in Section 5 and conclusions plus future direction of this work are outlined in Section 6.

2 Background

One popular approach in the 1980's to the general Image Understanding problem was knowledge-directed vision systems. A typical knowledge-directed approach to image interpretation seeks to identify objects in unconstrained two-dimensional images and to determine the three-dimensional relationships between these objects and the camera by applying object- and domain-specific knowledge to the interpretation problem. A survey of this line of research in computer vision can be found in [6], [5], and [4].

Typically, a knowledge-based vision system contains a knowledge base, a controller, and knowledge sources (or visual operators). In most of these systems the controller and the vision algorithms are combined into a single system. Problems common to most of the knowledge-directed vision systems include: control for vision procedures was never properly addressed as an independent problem [5], the system's structure did not facilitate entry of new knowledge [4], and the knowledge engineering task was formidable [5]. These are some of the issues that are addressed in this paper.

Bayesian networks have been successfully used in systems required to combine and propagate evidence for and against a particular hypothesis. Vision systems have been developed using Bayesian networks for knowledge representation and as a basis for information integration, e.g. Rimey [15], Binford [13] and Krebs [10] (for indoor applications), and Kumar [11] (for aerial image interpretation).

3 Value-driven control of a vision algorithms

The Ascender II system was designed for aerial image interpretation, particularly for the 3D reconstruction of urban areas. The system is divided into two independent parts - the reasoning subsystem and the visual subsystem - running under different operating systems on different machines, as shown in Figure 2. One advantage of this design is that changes in the reasoning subsystem, or in the visual subsystem, can be made independently.

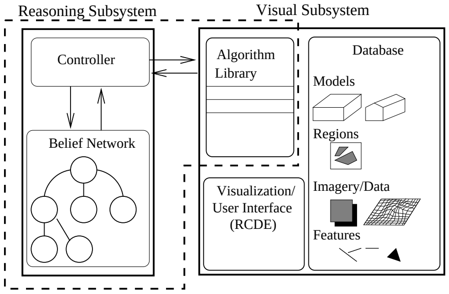

Figure 2: Process overview. Decisions are based on current knowledge about the site. Vision algorithms, stored in the visual subsystem, gather evidence about the site through focus of attention regions (FOAs), update the knowledge base, and produce geometric models.

Although the initial effort has focused primarily on recognizing and reconstructing buildings from aerial images, Ascender II has been designed as a general purpose vision system. The system has a set of focus-of-attention regions as input. These regions can be extracted from aerial images automatically (using a system such as Ascender I [3]), manually, or interactively (using cues from other sources such as maps or other classified images). The system's goal is to automatically select vision algorithms, recognize objects in the scene, and reconstruct these objects in 3D.

The system's knowledge base is composed of a set of Bayesian networks organized hierarchically. The networks are used to integrate information from different sources, and to label a region based on information provided by the visual operators. Each level of the hierarchy represents object classes at a specific scale [9]. The hierarchy leads to a system capable of performing incremental classification. The classification process is refined until the hierarchy reaches its finest level, or until the system exhausts all resources available. The Bayesian networks were developed using the HUGIN system [1].

The first set of networks were developed manually; two of the five networks used in the system are presented in Figure 3 and 4. The root node corresponds to the focus of attention region at a specific level of detail. All leaf nodes correspond to visual operators, and all internal nodes correspond to features that can be measured in the image. The probability table associated with the links between a feature node and an operator node reflects the reliability of the operator in retrieving the value of the feature; a link between the root node and the internal nodes represent relationships between object classes and feature values. The probability tables related to these links reflect the probability that a feature has a certain value given that the region is a certain object class, or:

$$P(Feature = k|Region = Object\_1)$$

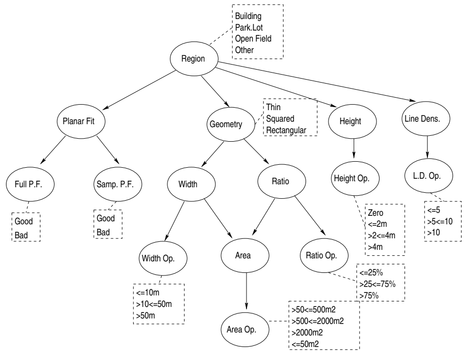

Figure 3: The level 0 hand-crafted network determines if a region belongs to one of the possible object classes (Building, Parking Lot, Open Field, or Other).

A set of experiments have been performed to compare alternative evaluation measures for operator selection. The first of these, called uncertainty distance [14], represents the difference between the value of the maximum belief in a node and the value of the belief if the node had a uniform distribution. Given a network, the system computes the uncertainty distance for all nodes that have a correspondent IU process and selects the node with the minimum uncertainty distance. This was shown empirically to be equivalent to entropy as an evaluation measure. We have also shown that using uncertainty distance leads to a system which uses significantly less resources (operators) than an exhaustive strategy yet achieves comparable performance [14].

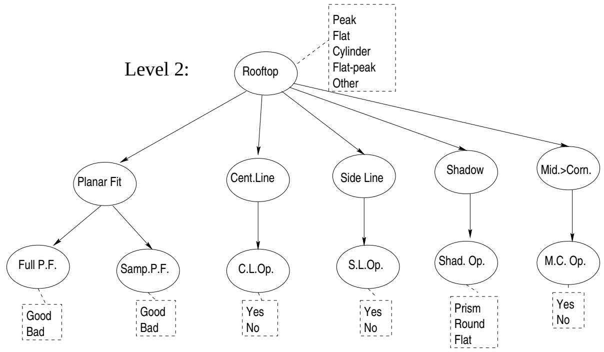

Figure 4: The level 2 hand-crafted network used to determine the type of rooftop (Peaked, Flat, Flat-Peak, Cylinder, or Other), once a single building is detected.

The work presented here uses the same system architecture, but it employs utility theory for selecting the operators to apply[12]. Utility theory is a probabilistic technique for decision making and it fits well in a Bayesian network system. Utility theory selects the decision that has the highest expected utility. In the discussion that follows, we use the following notation:

- $R_j \overset{\text{def}}{=}$ region R belongs to Class j.
- $DR_j \overset{\text{def}}{=}$ the decision that region R is identified as Class j.
- $E \overset{\text{def}}{=}$ all the evidence collected so far.
- $F_m \overset{\text{def}}{=}$ feature F is discretized in m states.

The expected utility (EU) of each decision is computed using the probability that a region belongs to a class j, $P(R_j|E)$, and the utility of deciding that a region is in class i given that the region belongs to class j, $U(DR_i|R_j)$, [12]:

$$EU(DR_i|E) = \sum_{j=1}^{N} U(DR_i|R_j) * P(R_j|E)$$

The current utility of the decision is defined as the maximum value among each of the expected utilities:

$$max(EU(DR_i|E))$$

Table 1: The table shows all utilities for the level 0 network in the Ascender II system.

Decide	*Class*			
	Building	Park. Lot	Open Field	Other
Building	1	0	0	0
Parking Lot	0	1	0	0
Open Field	0	0	1	0
Other	0	0	0	1

The best decision is defined as the decision α which gives the maximum expected utility:

$$\alpha = argmax_i(EU(DR_i|E))$$

In our problem domain the system has to decide the most likely identity (e.g. label) of a region. Assume that there are K features that can be measured in the region, the measurements are not completely reliable, and the measurements help in deciding about the region's label.

The region's prior probabilities and the conditional probability tables relating features with labels are stored in the Bayesian networks. The utility tables storing the values $U(DR_i|R_j)$ are not hard to define and can be adjusted by the user of the system to reflect specific goals for the classification process [12]. The utility tables used here are all similar, with ones on the diagonal and zeros in all other entries (see Table 1). In this case, only the correct labels are accepted.

Features are selected based on the value of information [8] associated with each feature. This value is computed as follows: for each feature currently available compute the expected utility of the system given that information about the feature is known.

$$EU(DR_i|E, F_m) = \sum_M P(F_m)*max_i(EU(DR_i|E, F_m))$$

Now, compute the value of information of each feature as follows:

$$VI(F_m) = EU(DR_{\alpha'}|E, F_m) - EU(DR_\alpha|E) \qquad (1)$$

and select the feature with the highest value of information. Intuitively, the value of information measures the expected improvement in the utility of the best decision, once the result of an operator becomes available.

Figure 5 shows a generic Bayesian network that will be used to illustrate how feature selection is performed in the Ascender II system. The first step is to compute the system's utility before extracting any information about the features. Each decision has an expected utility $U(Dec_i) = EU(DR_i|E)$; the expected utilities of the decisions can be calculated by multiplying the matrix of utilities by the column vector of beliefs from the root node, as shown in Figure 5. The system's utility is the maximum value among the utilities of the decisions.

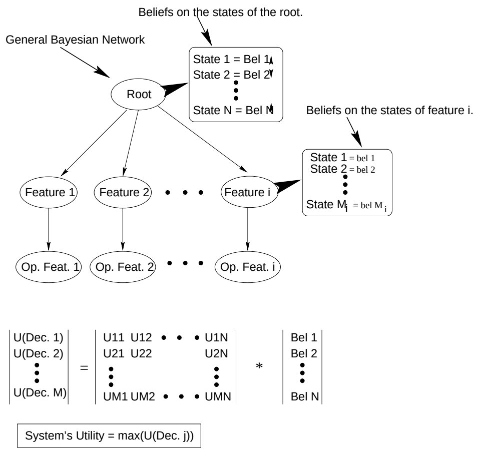

Figure 5: A generic Bayesian network for the Ascender II system

The next step is to compute the value of information of each feature. This is performed by computing the expected utility of each feature as follows: assume feature "i" has "M" states, $state_1, state_2, \cdots, state_M$; each state in feature "i" has a corresponding belief, $bel_1, bel_2, \cdots, bel_M$. These beliefs correspond to the current expectation about the outcome of feature "i". Set the outcome of feature "i" to $state_1$ (make the belief of $state_1 = 1$ and the belief of all other states equal to 0), and propagate the information through the network. This will change the beliefs in the states of the root node. Use this new set of beliefs in the root node to compute the new utility of the system. When completed, the value of information is found from equation 1.

4 Learning the models for the control structure

The knowledge engineering necessary to design a efficient Bayesian network (structure and probability tables) is a time consuming task, even for small networks such as those currently used in the Ascender II system. This has been one of the main criticisms of Bayesian networks.

Algorithms for learning Bayesian networks from data have been developed [7, 2]. Cheng's algorithms [2] are based on statistical measures over pairs of random variables. The algorithms perform conditional independent tests using mutual information, and conditional mutual information given a third variable, and use these tests to define causality. Cheng's algorithms were used to learn the structure and the probability tables for the networks in the Ascender II system.

The data used for learning was collected from 3 different well-known data sets (Ft. Hood, Ft. Benning and Avenches); overall, 79 regions were selected representing a mix of objects drawn from buildings, parking lots, grassy fields, etc. All regions were presented to a set of 6 human subjects, and the subjects were asked to estimate the state of each feature in the feature set (features were coarsely quantized to facilitate the human task). This information was compiled and used to learn a Bayesian network representing the task domain.

Note that the structures as learned contain only the node representing the region plus the nodes representing all the features. The operator nodes (along with their reliability tables) were added manually after the learning phase was complete. If the true value of each feature is known, the tables representing the operator's reliability can also be learned from the data.

The learned networks corresponding to Figures 3 and 4 are shown in Figure 6 and 7. The general structure is completely different, although some of the substructures were preserved. Also, the learned networks are generally more densely connected.

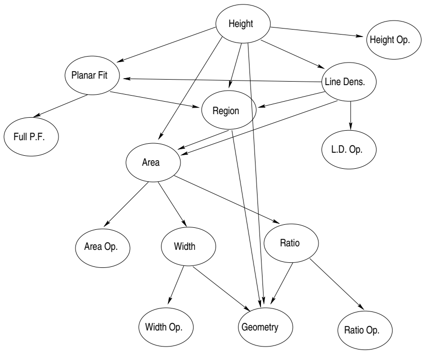

Figure 6: The level 0 learned network determines if a region belongs to one of the possible object classes: Building, Parking Lot, Open Field, or Other.

The networks learned from data are limited to the objects present in the training data. For instance, the data used to learn the networks had only peak- and flat-roofed buildings. Thus the feature *Rooftop* in Figure 7 has only states for *Peak* and *Flat* roofs, and not the more general structure as in the hand-crafted networks presented in Figure 4.

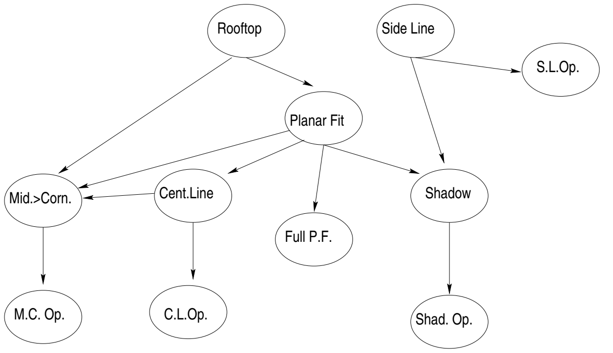

Figure 7: This level 2 learned network is called after a single building is detected. It is used to determine the building's rooftop type (Peak or Flat).

5 Results

A set of experiments were performed on the Fort Hood data set (7 views with known camera parameters and corresponding digital elevation map DEM) shown in Figure 8, on the Avenches data set (1 view and a DEM) shown in Figure 9, on the Fort Benning data set (2 views and a DEM) shown in Figure 10, and on the ISPRS Flat data set (2 views and a DEM) show in Figure 11. These data sets are an effective test suite because they have different numbers of images, different resolutions and different numbers of objects in each class.

The first experiment was designed to show that a more disciplined approach to feature selection leads to a more efficient system. The experiment provides a comparison between the system using uncertainty distance (Basic System) and the system using utility theory (System A). Both systems used the hand-crafted networks. The results in terms of classification and number of operators used are presented in Tables 2 and 3.

Table 3 shows that the overall classification obtained by the two selection processes is about the same. Table 2 shows that the selection of operators is more efficient using utility theory (10% fewer operators). This result confirms the intuition that a selection methodology using utility theory would choose more effective operators, thus classifying regions faster.

The second set of experiments was designed to demonstrate the performance of the system using the learned networks on the same data sets used for training. Although the regions used in the these experiments are the same as the ones used for learning, there are two major differences that have to be considered:

Table 2: Total number of calls to visual operators for all data sets for all classes.

Decision process	Number of Operators
Utility Theory	430
Uncertainty Distance	475

1. During the experimental phase the features were computed algorithmically from the image data by a visual operator. The results do not necessarily correspond to the outcome given by humans in the learning phase.

2. The values of the features computed by the visual operator were entered into the operator's node and were attenuated by the operator's reliability during the propagation.

First, the networks and probability tables (including prior probabilities) as learned from the data (System B) was applied in the 3 data sets (Ft. Hood, Avenches and Ft. Benning). Because the prior probabilities learned from data reflect the exact frequency of each object class, the system should react faster to feature values retrieved and it would not be a fair comparison to System A. So a second test was performed where the prior beliefs for each object class were changed in the networks to reflect the same prior probabilities used in the hand-crafted networks (System C). The results obtained for these two experiments are shown in Tables 4 and 5.

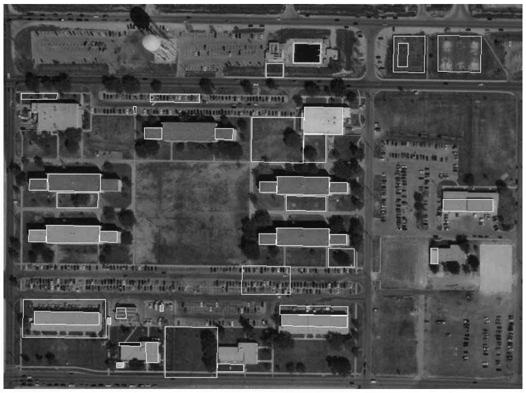

Figure 8: The input regions from the Fort Hood data set. These regions were obtained by running the original Ascender I system constrained to detect two-dimensional building footprints.

The numbers shown in Table 5 are similar to the numbers presented in Table 3. Thus, the system using Bayesian networks learned from data generates classifications very similar to the system using the hand-

Table 3: Summary of the recognition process for different data sets using the hand-crafted networks. In each case the number of objects correctly identified is shown, followed by the total number of objects evaluated by the system.

Uncertainty Distance - Basic System				
Data set	Overall	Level 0	Level 1	Level 2
Fort Hood	34/42	36/42	22/24	21/21
Avenches	12/18	15/18	12/13	5/7
Fort Benning	17/19	18/19	17/18	17/18

Utility Theory - System A				
Data set	Overall	Level 0	Level 1	Level 2
Fort Hood	35/42	37/42	23/25	21/21
Avenches	13/18	16/18	12/13	5/7
Fort Benning	16/19	18/19	17/18	16/17

Table 4: Total number of calls to visual operators for all data sets for all classes.

Decision process	Number of Operators
Learned Networks	322
Learned + Modified Priors	400

Table 5: Summary of the recognition process for different data sets using the learned networks.

Learned Networks - System B				
Data set	Overall	Level 0	Level 1	Level 2
Fort Hood	33/42	34/42	20/21	20/20
Avenches	16/18	18/18	15/15	7/9
Fort Benning	15/19	18/19	17/18	15/17

Learned Networks + Modified Priors - System C				
Data set	Overall	Level 0	Level 1	Level 2
Fort Hood	34/42	35/42	20/21	20/20
Avenches	13/18	16/18	12/14	6/7
Fort Benning	16/19	18/19	17/18	16/17

Table 6: Summary of the recognition process for the Flat data sets using the hand-crafted and the learned networks with utility theory.

Flat Data Set					
System	Overall	Level 0	Level 1	Level 2	Operators
Hand-crafted	22/30	23/30	21/21	13/14	170
Learned	26/30	27/30	21/21	13/14	162

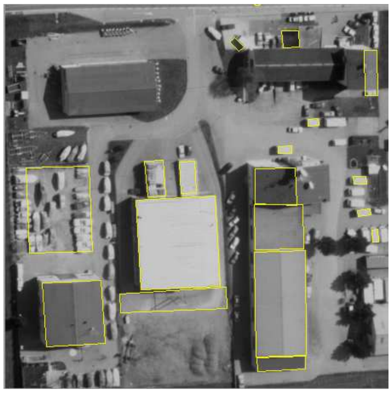

Figure 9: The input regions from the Avenches data set. The regions were obtained by running the Ascender I system.

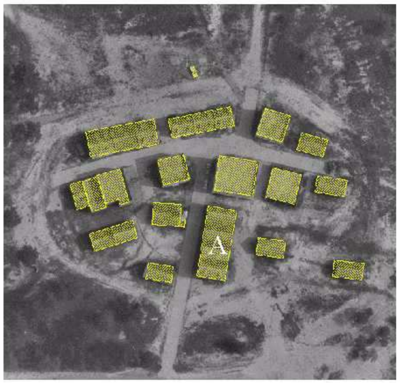

Figure 10: The input regions from the Fort Benning data set. These regions were obtained by a combination of polygons extracted using Ascender I and polygons extracted from SAR data.

crafted networks. However, "System B" was able to classify the regions using 32% fewer operators than the "Basic System". "System C" used 15% fewer operators than the "Basic System". The fact that "System C" used more operators than the "System B" was expected because the distributions of beliefs over the object classes were more uniformly distributed in "System C" than in "System B", thus "System C" requires more exploratory calls before deciding about a region.

The third experiment was designed to show that the structure and relationships among features learned from data is robust enough to be applied to a different data set. In this experiment, the hand-crafted system using utility theory was compared to the learned system applied to the Flat data set. In both systems the prior beliefs were adjusted accordingly. The results over 30 regions are shown in Table 6.

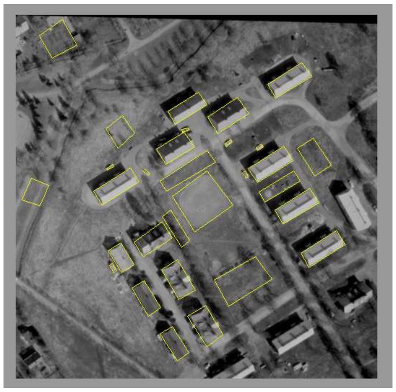

Figure 11: Set of regions extracted by hand from the Flat data set.

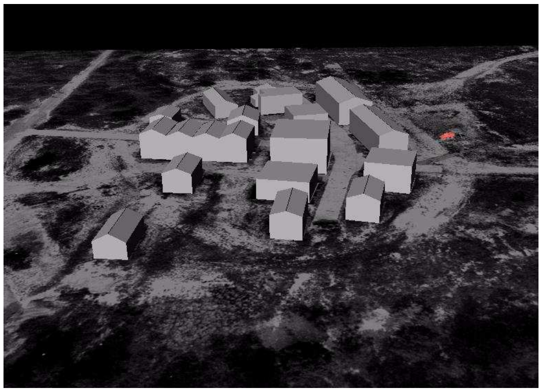

Figure 12: 3D reconstruction on the Fort Benning data set.

The number of operators used by the system using the learned networks is slightly smaller (5%), but the larger number of relationships between the features in the learned networks allowed better performance of the system on the new data set (87% correct classifications against 73% for the system with the hand-crafted networks).

One example of the 3D reconstruction that can be obtained using the Ascender II system is presented in Figure 12. The maximum error between the reconstructed buildings and the CAD models hand-crafted for the buildings in the Fort Benning data set is less than 1.2 meters.

6 Conclusions and Future Work

The overall performance of the Ascender II system using utility theory or uncertainty distance is above 80% in terms of classification. When utility theory and value of information is used, the system selects operators more efficiently and is able to identify objects faster.

The knowledge base in Ascender II is based on Bayesian networks. Evidence from different sources are combined in the Bayesian networks and each contributes to the region classification.

We have also shown that the networks can be learned from data. The system using the learned networks had a better performance either in terms of the number of operators required to correctly classify the regions, or in terms of the percentage of regions correctly classified. The data used to learn the networks have to be representative of all objects classes desired in the system. The learned networks are robust enough to be applied in a different data set with a simple adjustment of prior beliefs for the object classes.

The hierarchical structure leads to a system capable of performing incremental classification. The current system can be adjusted to behave as an anytime system, where resources, such as number of operators or processing time, can be limited and the overall performance optimized for the resources available.

Another possible extension of this system is related to temporal reasoning. If a 3D reconstruction of a site is available and a new image is obtained for the same area, how can the information previously computed be used to drive the system in order to detect changes and to reconstruct the new site efficiently.

References

[1] Andersen, S., Olesen, K., Jensen, F., and F., J. Hugin - a shell for building bayesian belief universes for expert systems. In *Proceedings of the 11th International Congress on Uncertain Artificial Intelligence* (1989), pp. 1080–1085.

[2] Cheng, J., Bell, D., and Liu, W. Learning bayesian networks from data: An efficient approach based on information theory. Tech. Rep. -, Department of Computer Science - University of Alberta, 1998.

[3] Collins, R., Cheng, Y., Jaynes, C., Stolle, F., Wang, X., Hanson, A., and Riseman, E. Site model acquisition and extension from aerial images. In *Proceedings of the Interantional Conference on Computer Vision* (1995), pp. 888–893.

[4] Crevier, D., and Lepage, R. Knowledge-based image understanding systems: a survey. *Computer Vision and Image Understanding 67(2)* (1997), 161–185.

[5] Draper, B., Hanson, A., and Riseman, E. Knowledge-directed vision: control, learning, and integration. *Proceedings of the IEEE 84(11)* (1996), 1625–1637.

[6] Haralick, R., and Shapiro, L. *Computer and Robot Vision.* Addison-Wesley, 1993.

[7] Heckerman, D. A tutorial on learning with bayesian networks. Tech. Rep. MSR-TR-95-06, Microsoft Research, March 1995.

[8] Howard, R. Information value theory. *IEEE Transactions on Systems, Science and Cybernetics SSC-2(1)* (1966), 22–26.

[9] Jaynes, C., Marengoni, M., Hanson, A., and Riseman, E. 3d model acquisition using a bayesian controller. In *Proceedings of the International Symposium on Engineering of Intelligent Systems, Tenerife, Spain* (1998), pp. 837–845.

[10] Krebs, B., Burkhardt, M., and Korn, B. A task driven 3d object recognition system using bayesian networks. In *Proceedings of the International Conference on Computer Vision, Bombay, India* (1998), pp. 527–532.

[11] Kumar, V., and Desai, U. Image interpretation using bayesian networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence 18(1)* (1996), 74–77.

[12] Lindley, D. *Making Decisions: Second Edition.* John Wiley and Sons, 1985.

[13] Mann, W., and Binford, T. An example of 3-d interpretation of images using bayesian networks. *DARPA Image Understanding Workshop* (1992), 793–801.

[14] Marengoni, M., Jaynes, C., Hanson, A., and Riseman, E. Ascender ii, a visual framework for 3d reconstruction. In *Proceedings of the International Conference on Vision Systems, Las Palmas, Spain* (1999), pp. 469–488.

[15] Rimey, R., and Brown, C. Task-oriented vision with multiple bayes nets. In *Active Vision*, A. Blake and A. Yuille, Eds. The MIT Press, 1992.