An Attention Mechanism for Top-Down Recognition of 3-D Objects

Sven J. Dickinson
Department of Computer Science
University of Toronto
6 King's College Rd.
Toronto, Ontario, Canada M5S 1A4
TEL: (416) 978-7564
email: sven@vis.toronto.edu

Henrik I. Christensen
Laboratory of Image Analysis, IES
Aalborg University
DK-9220 Aalborg, Denmark
TEL: 45 98 15 85 22
email: hic@vision.auc.dk

Abstract

We present an approach to the top-down recognition of volumetric part-based 3-D objects from a single 2-D image. In previous work [11, 10], we focused on the unexpected object recognition problem and required a complete and consistent interpretation of the image, in terms of a set of 3-D volumetric parts, before matching recovered parts to object models. This requirement was both time consuming and ambitious, particularly with less constrained images of scenes containing less constrained objects. In this paper, we exploit knowledge of the target object being searched for to drive an attentional mechanism based on image regions. Specifically, perform a depth-first search of a tree whose levels correspond to objects, volumetric parts, part aspects, and aspect faces, respectively. At any node in the tree, we can rank the node's subtrees according to the probabilities in an augmented version of the aspect hierarchy first introduced in [8]. We present the approach in detail and demonstrate its application to real images.

area: computer vision

This paper is neither currently under review for a journal or another conference, nor will it be submitted during IJCAI's review period.

1 Introduction

In previous work, we presented a bottom-up approach to the recovery of qualitative 3-D volumetric primitives from a single 2-D image [11]. The approach is based on a hybrid object representation in which objects are composed of a set of chosen 3-D object-centered volumetric primitives; the primitives, in turn, are mapped to a set of 2-D viewer-centered aspects. The primitive recovery problem was formulated as a heuristically guided search through the various groupings of image regions into aspects, each representing a view of a volumetric part. A system called OPTICA (**O**bject recognition using **P**robabilistic **T**hree-dimensional **I**nterpretation of **C**omponent **A**spects) was built to demonstrate the approach, and it was successfully applied to the problem of unexpected object recognition from real images [10].

A major limitation of the approach was its dependency on a complete and consistent covering of the image regions in terms of a set of aspects. This assumes that all objects visible in the image are made up of the chosen primitives. And what about the case where the target object is known (expected object recognition)? Is it really necessary to completely recover all high-order shape information in the scene in order to locate a particular object? OPTICA was first extended to the problem of top-down, expected object recognition, by using knowledge of the target object to focus the various search procedures inherent in OPTICA's unexpected object recognition paradigm [6]. However, as a starting point, it still required a complete aspect covering of the image. Although some heuristics were introduced to accommodate over- and undersegmentation [11], lighting conditions had to be favorable for a successful interpretation to be generated. When dealing with noisy images of less constrained scenes, along with shadows and poor lighting, an aspect covering of the entire scene is not only very costly, but very ambitious.

In this paper, we propose a new technique for expected object recognition that requires neither a high quality region segmentation of the scene nor a complete aspect covering of the scene. Specifically, we map knowledge of the target object into a set of ranked candidate search region shapes. A preprocessing step recovers only a set of image regions, each characterized by a set of possible shape labels ordered according to a goodness-of-fit. The labeled regions offer an attention mechanism whereby the search for an object (through one of its component primitives) can be focused at key points in the image. We will begin by reviewing the hybrid object representation and proceed to detail the search algorithm. We then apply the algorithm to example images and conclude by evaluating its performance.

2 Review of the Object Representation

2.1 Object-Centered Modeling

Given a database of object models representing the domain of a recognition task, we seek a set of three-dimensional volumetric primitives that, when assembled together, can be used to construct the object models. Many 3-D object recognition systems have successfully employed 3-D volumetric primitives to construct objects. Commonly used classes of volumetric primitives include polyhedra (e.g., Lowe [25]), generalized cylinders (e.g., Brooks [4]), and superquadrics (e.g., Pentland [27]). Whichever set of volumetric modeling primitives is chosen, they will be mapped to a set of viewer-centered aspects.

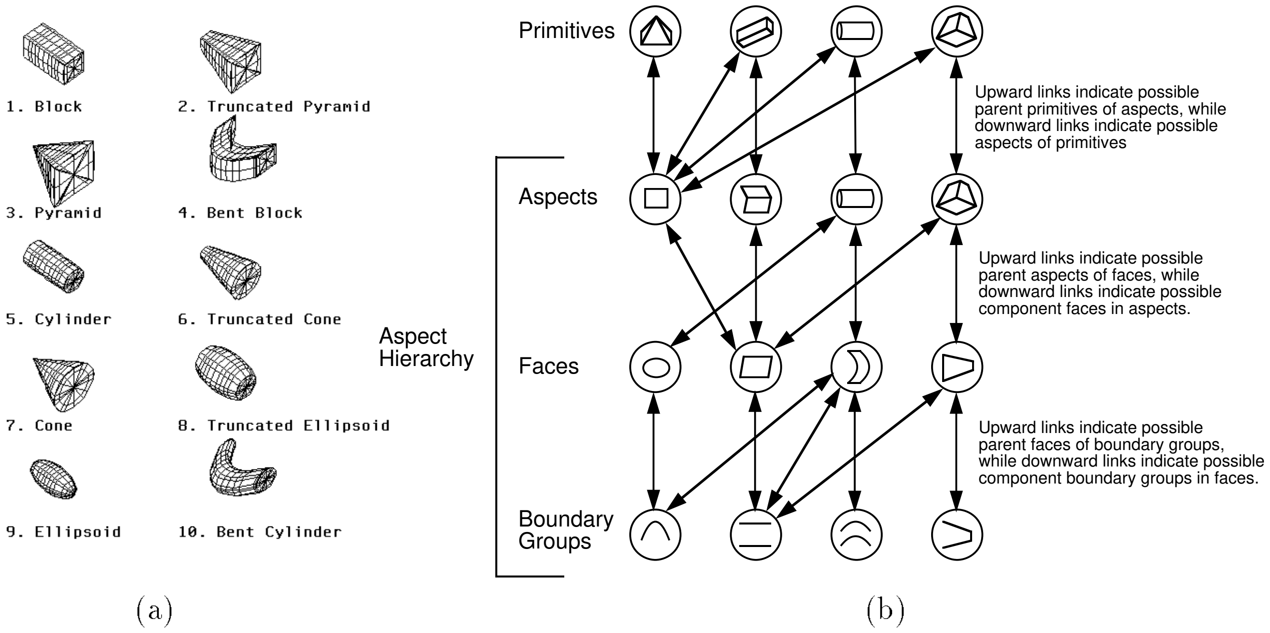

Figure 1: (a) The Ten Modeling Primitives, (b) The Augmented Aspect Hierarchy

To demonstrate our approach to object recognition, we have selected an object representation similar to that used by Biederman [2], in which the Cartesian product of contrastive shape properties gives rise to a set of volumetric primitives called geons. Since the introduction of Biederman's geons to the vision community, a number of other researchers have developed computational models for geon recovery resulting in a number of geon-based recognition systems [1, 3, 13, 18, 29]. However, unlike these approaches, which are typically applied to manually segmented line drawings, our approach is applied to real images and is not dependent on the choice of geons as modeling primitives. For our investigation, we have chosen three properties including cross-section shape, axis shape, and cross-section size variation (Dickinson et al. [8]). The cartesian product of the dichotomous and trichotomous values of these properties give rise to a set of ten primitives (a subset of Biederman's geons), modeled using Pentland's SuperSketch 3-D modeling tool [27], and illustrated in Figure 1a. To construct objects, the primitives are attached to one another with the restriction that any junction of two primitives involves exactly one distinct surface from each primitive.

2.2 Viewer-Centered Modeling

Traditional aspect graph representations of 3-D objects model an entire object with a set of aspects, each defining a topologically distinct view of an object in terms of its visible surfaces [5, 12, 14, 16, 15, 17, 21, 22, 28, 32, 33, 34, 35]. Our approach differs in that we use aspects to represent a (typically small) set of volumetric primitives from which each object in our database is constructed, rather than representing an entire object directly. Consequently, our goal is to use aspects to recover the 3-D primitives that make up the object in order to carry out a recognition-by-parts procedure, rather than attempting to use aspects to recognize entire objects. The advantage of this approach is that since the number of qualitatively different primitives is generally small, the number of possible

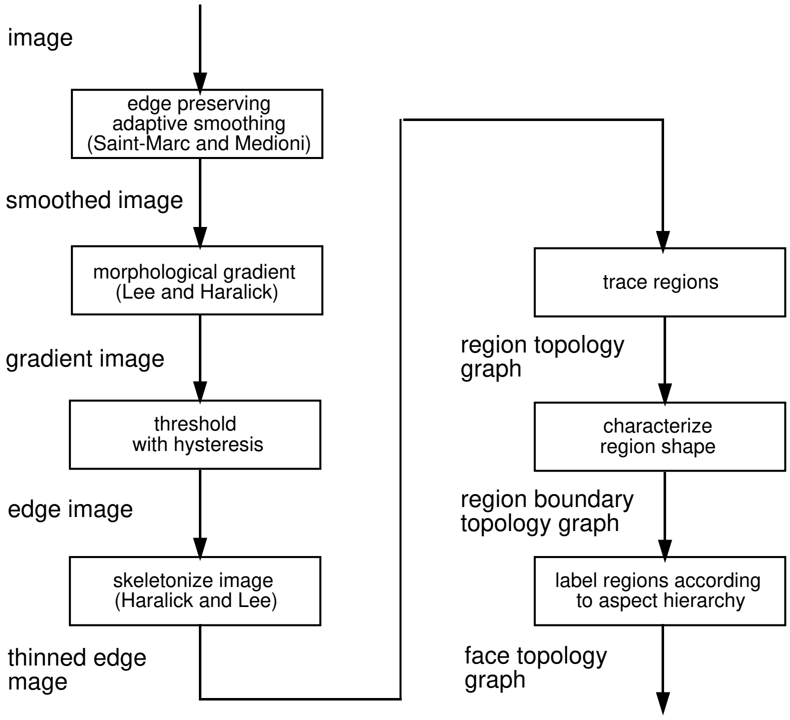

Figure 2: Face Recovery

aspects is limited and, more important, *independent* of the number of objects in the database. The disadvantage is that if a primitive is occluded from a given 3-D viewpoint, its projected aspect in the image will also be occluded. Thus we must accommodate the matching of occluded aspects, which we accomplish by use of a hierarchical representation we call the *aspect hierarchy*.

The aspect hierarchy consists of three levels, consisting of the set of *aspects* that model the chosen primitives, the set of component *faces* of the aspects, and the set of *boundary groups* representing all subsets of contours bounding the faces. The ambiguous mappings between the levels of the aspect hierarchy were originally captured in a set of upward conditional probabilities (Dickinson et al. [7, 9]), mapping boundary groups to faces, faces to aspects, and aspects to primitives. However, for the approach described in this paper, the aspect hierarchy was augmented to include the downward conditional probabilities mapping primitives to aspects, aspects to faces, and faces to boundary groups.[1] Figure 1b illustrates a portion of the augmented aspect hierarchy.

3 Face Recovery

The process of recovering faces from an image is outlined in Figure 2, and consists of first extracting a set of regions from an image followed by a classification of the regions' shapes. In the following sections, we discuss these two processes in greater detail.

[1] For the probabilistic search process described in Section 4, the augmented aspect hierarchy is actually represented by two acyclic graphs, one capturing the upward conditional probabilities and the other capturing the downward conditional probabilities.

3.1 Region Segmentation

The first step in recovering a set of faces is a region segmentation of the input image. We begin by applying Saint-Marc and Medioni's edge-preserving adaptive smoothing filter to the image [31].[2] Next, we apply a morphological gradient operator [23] which computes the difference between the maximum and minimum pixel values in a neighborhood. A hysteresis thresholding operation[3] is then applied to produce a binary image, which is is then skeletonized [19]. The closed contours are traced and the resulting regions are captured in a *region topology graph* in which nodes represent regions and arcs specify region adjacency. The region segmentation algorithm has been selected to satisfy our goal of near real-time performance. With the exception of the boundary following, each of the smoothing, gradient, thresholding, and thinning steps can be performed at frame rates on a Datacube image processor.

3.2 Face Labeling

Once the faces have been extracted, we must classify each region according to the faces in the aspect hierarchy. Both an image region and an aspect hierarchy face are represented by a graph in which nodes represent the face's bounding (partitioned) contours, and arcs represent relations between contours. We apply Saint-Marc and Medioni's scale-space adaptive smoothing algorithm [31] to partition the bounding contour of each region at low, medium, and high scales. Each of the resulting contours is then characterized as straight or curved depending on how well a straight line can be fitted to it; furthermore, curves are characterized as convex or concave. Two non-coterminating lines are considered parallel if the angle between their fitted lines is small, while two non-coterminating curves are considered parallel if one is convex, one is concave, and the angle between their directions is small.[4] Two non-coterminating, non-parallel lines are considered symmetric if there is sufficient overlap when one line is projected onto the other.[5]

Each of the three scales in the curve partitioning step gives rise to a graph representing an image region, called a *region boundary graph*; the region boundary graphs representing all the image regions are captured in a *region boundary topology graph*. Consequently, the classification of an image region consists of comparing its three region boundary graphs to those graphs representing the faces in the aspect hierarchy. We begin with the region boundary graph corresponding to the low scale in the curve partitioning step, often representing an oversegmentation of the region's bounding contour. If there is an exact match, as shown in Figure 3, then we immediately generate a *face hypothesis* for that image region, identifying the label of the face. If for any reason (e.g., occlusion, segmentation errors, noise, etc.), there is no match, we must descend to the boundary group level of the aspect hierarchy, as shown in Figure 4. We then compare *subgraphs* of the region boundary graph representing the image region to those graphs at the boundary group level of the aspect hierarchy. For each subgraph that matches, we generate a face hypothesis with a probability determined by

[2]We perform 5 iterations of the algorithm, using a k value of 15.

[3]High and low thresholds were 35 and 6 respectively.

[4]The direction of a curve is computed as the vector whose head is defined by the midpoint of the line joining the two endpoints of the curve, and whose tail is defined by the point on the curve whose distance to the line joining the endpoints is greatest.

[5]Two non-parallel vectors will have an intersection point. When one vector is rotated about that point it can be brought into correspondence with the other. If the resulting overlap of the two lines is a large portion of the smaller of the two lines, the lines are said to be symmetric.

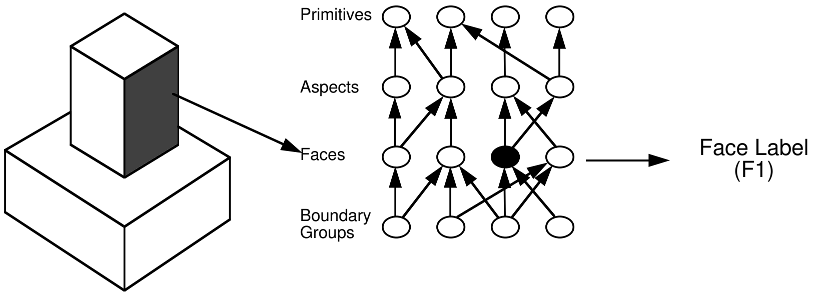

Figure 3: Labeling an unoccluded region

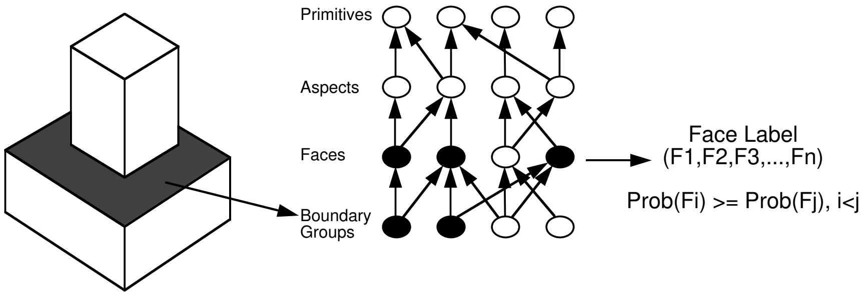

Figure 4: Labeling an occluded region

the appropriate entry in the conditional probability matrix mapping boundary groups to faces.

Matching proceeds from the lowest to the highest scales. As soon as an exact match is found, only the single matched face hypothesis is retained, ignoring higher scales and discarding all partial hypotheses for efficiency. If no exact match is found at any scale, all partial face interpretations recovered from the three scales are retained. The labeled face hypotheses for all regions in the image are captured in a *face topology graph*.

4 Attention Mechanism

The goal of the attention mechanism is to decide where in the image we should begin searching for the target object. However, before deciding this, we must decide what features in the image we are searching for. There is a very important trade-off here which is critical to the problem of choosing features to attend to. On one hand, we wish to expend as little effort as possible in recovering a set of features that might belong to the object and hence give us a place to initiate the search for the object. However, if the recovered features are not discriminative enough, we will be faced with an abundance of features located throughout the image, all attempting to draw the attention mechanism to their location. Thus we have a trade-off between uniqueness, or indexing power, and cost of recovery.

In analyzing the upward conditional probabilities used in OPTICA's unexpected object recognition paradigm [11], we concluded that for 3-D modeling primitives which resemble the commonly used generalized cylinders, superquadrics, or geons, the most appropriate image features for recognition appear to be image regions, or faces. Moreover, the utility of a face description can be improved by grouping the faces into the more complex aspects, thus obtaining a less ambiguous mapping to the primitives and further constraining their orientation. Only when a face's shape is altered due to primitive occlusion or intersection should we descend to analysis at the contour or boundary group level. For our expected object recognition problem, we will essentially reverse this process. Starting from the object, we will generate predictions down to the level of labeled faces. Since our face recovery preprocessing step recovers labeled faces, our attention can be drawn to those recovered labeled faces which match predictions. Furthermore, since each face label has a corresponding probability, that probability can be used to rank-order candidate image faces for search.

In selecting which recovered face to focus our attention on, we utilize a decision theoretic approach using a Bayesian network. A similar approach was reported by Levitt et al. [24], who use Bayesian networks for both model representation and description of recovered image features. Specifically, they use Bayesian networks for both data aggregation and selection of actions and feature detectors based on expected utility. The approach is thus centered around the use of a Bayesian approach to integration and control. Similar techniques have also been reported by Rimey and Brown [30], and Jensen et al. [20], where both regions of interest and feature detectors are selected according to utility/cost strategies.

To select a region of interest, i.e., attend to a particular face, the augmented aspect hierarchy may be considered as a Bayesian network, allowing us to utilize decision theory as described, for example, by Pearl [26]. To apply such a strategy, it is necessary to define both utility and cost measures. The utility function, U, specifies the power of a given feature at one level of the augmented aspect hierarchy, e.g., primitives, aspect, faces, and boundary groups, to discriminate a feature at a higher level. The cost function, C, specifies the cost of extracting a particular feature. The subsequent planning is then aimed at optimizing the benefit, $\max B(U, C)$; profit, e.g., $utility - cost$, is often maximized in this step.

For the system described in this paper, the face recovery algorithm was chosen to support a simple implementation on a pre-attentive, real-time, pipeline architecture. Thus, the cost of face recovery is constant and, in fact, equal for all types of faces. Therefore, given such an implementation, the selection of which face to consider next should simply optimize the utility function.

The process of mapping an object to a candidate face is outlined in Figure 5. Given a target object, $object_T$, the first step is to choose a target primitive, $primitive_T$, to search for. Next, given a target primitive, $primitive_T$, we choose a target aspect, $aspect_T$, to search for. Finally, given a target aspect, $aspect_T$, we choose a target face, $face_T$, to search for. Given a target face, $face_T$, we then examine the face topology graph for labeled faces which match $face_T$. If there is more than one, they can be ranked in descending order according to their probabilities.

The above top-down sequence of predictions represents a best-first search of a tree defined by each object; the root of the tree represents the target object, while the leaf nodes of the tree represent target faces. The target primitive subtrees for each object tree are independent of the object database and can be specified at compile time. The branching factor at a given node in any object tree can be reduced by specifying a probability (or utility) threshold on a prediction.

The heuristic we use to guide the search is based on the power of an object's features, e.g., primitives, aspects, and faces, to identify the object. For example, to determine how discriminative a particular primitive, $primitive_i$, is in identifying the target object, $object_T$, we use the following

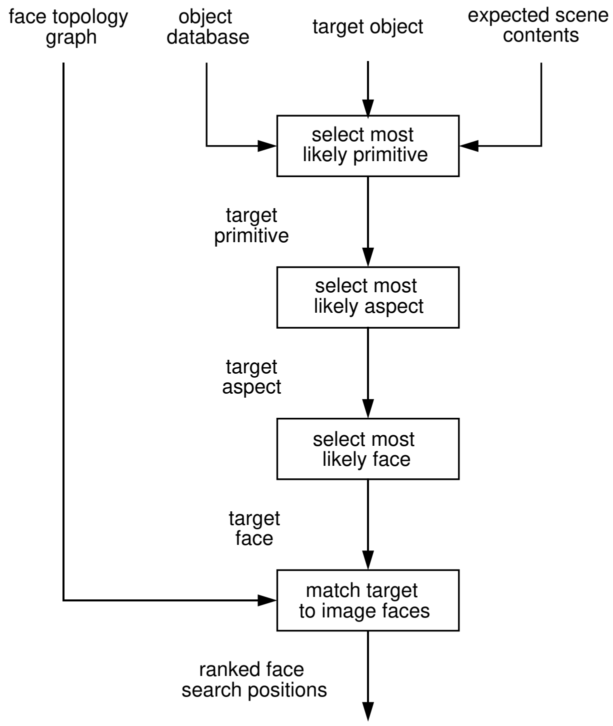

Figure 5: Attention Mechanism

function:

$$D(primitive_i, object_T) = \frac{p(object_T|primitive_i)}{\sum_j p(object_j|primitive_i)} * p(primitive_i) \quad (1)$$

The numerator specifies how discriminative $primitive_i$ is for $object_T$, while the ratio specifies the "voting power" of $primitive_i$ for the object of interest. $p(object_i|primitive_j)$, for any given i and j, is computed directly from the contents of the object database. The last term specifies the likelihood of finding the primitive, and is included to discourage the selection of a primitive which is highly discriminative but very unlikely. The $p(primitive)$ may be calculated as follows:

$$p(primitive_i) = \sum_k (p(primitive_i|object_k) * p(object_k)) \quad (2)$$

where $p(primitive_i|object_k)$, for any given i and k, is computed directly from the object database, and $p(object_k)$ represents a priori knowledge of scene content. In a similar fashion, we can define $D(primitive_T, aspect_i)$ and $D(aspect_T, face_i)$.

To determine the best target primitive to search for in order to recognize the target object, the following utility function is evaluatd:

$$U(primitive_T, object_T) = \max_a\left(\frac{p(object_T|primitive_a)}{\sum_j p(object_j|primitive_a)} * p(primitive_a)\right) \quad (3)$$

Again, similar utility functions can be defined for $U(aspect_T, primitive_T)$ and $U(face_T, aspect_T)$.

When we descend the search tree to a given target face, we search for matching face candidates in the face topology graph. We focus our attention on the best face matching the target face, and proceed to recognize the object, as described in the next section.

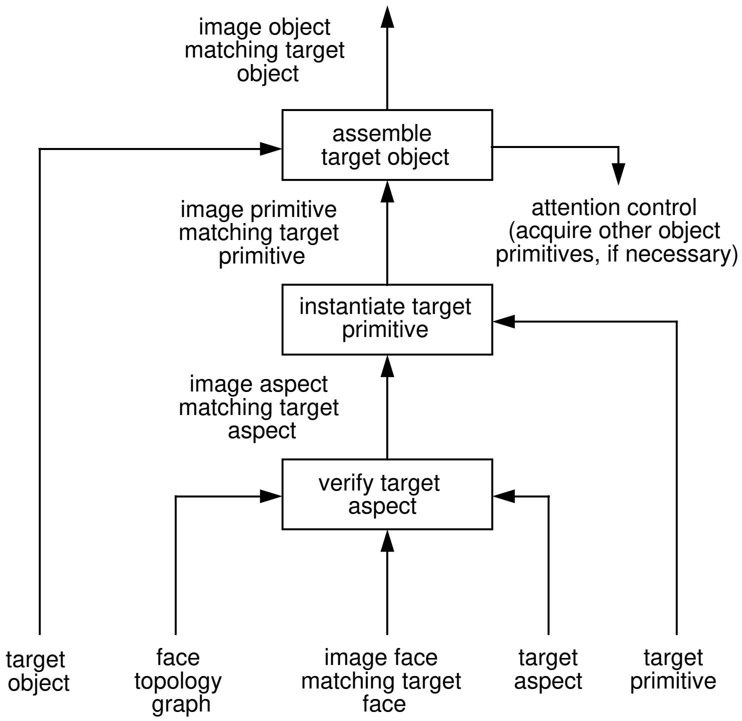

Figure 6: Object Recognition

5 Object Recognition

From a target face in the image which matches a target face in the object tree, we next proceed to recognize the object using the process shown in Figure 6. Recognition is the process by which we move from a matched target face node in the search tree back up to an object. Once we have a matched face leaf node, our next step is to verify its parent (target) aspect. This entails searching the vicinity of the target face for faces whose labels and configuration match the target aspect. Note that the resulting verified aspect has a score associated with it which can be compared to a score threshold to terminate the search from a particular target face. Details on the aspect verification process can be found in [11].

Once a target aspect is found, we then proceed up the tree one level to the target primitive, defining a mapping between the faces in the target aspect and the surfaces on the target primitive. Moving back one level to the object, we must then decide whether or not we have enough information confirming the target object. If so, the recognition process is complete. If not, we must then decide which primitive to search for next. If we choose a primitive which is connected to a primitive we have already verified, we can move back down its branch in the tree and, when matching its target faces to image faces, consider only those image faces that are in the vicinity of the faces belonging to the verified primitive.

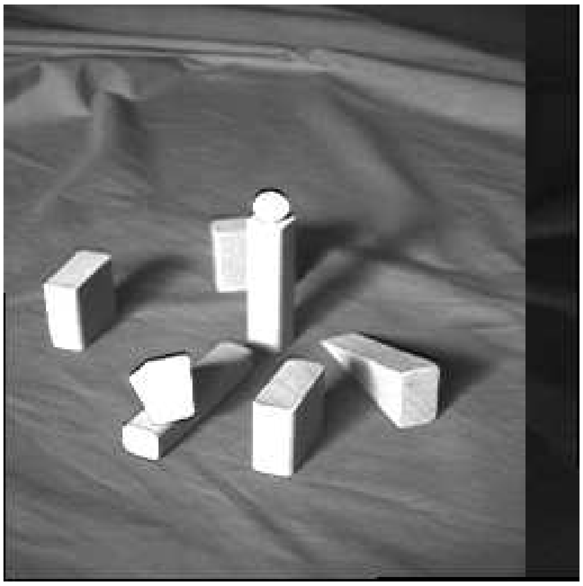

Figure 7: Image of Blocks on a Table

6 Results

To illustrate the shape recovery approach, we will begin by searching for instances of single primitives in a noisy image of cluttered scene; at this time, the top level of the search, i.e., selecting a target primitive given an object, has not yet been implemented. In the first example, we search for the best three instances of a block primitive in the image shown in Figure 7; the results are shown in Figure 8. At the top of the figure, the image window contains the contours extracted from the image, along with the face numbers. To the left is a window describing the recovered primitives found during the search. The mnemonic PL refers to the primitive label (see Figure 1(a)). The mnemonics AL, AP, and AS refer to the aspect label (see [11]), aspect probability, and aspect score (how well aspect was verified), respectively. The mnemonics FN, FL, FP, and PS refer to face number (in image window), face label (see [11]), face probability, and corresponding primitive attachment surface (see [11]), respectively, for each component face of the aspect.

If we examine the first primitive found during the search, we see that in Figure 7, it is actually a wedge-shaped object and not a block. The reason for this misinterpretation is due to a shadow edge that results in the top (triangular) face of the wedge being oversegmented in the image. The resulting region, numbered 12 in Figure 8, matches one of the component faces, FL9, in aspect AL29, of the block. Despite undersegmentation, oversegmentation, and occlusion of image regions, we are still able to recover instances of the target object.

In the next example, we search for a cylinder primitive in the stereo pair shown in Figure 9; the results are shown in Figure 10. Despite oversegmentation, the cylinder was correctly identified in each image. In the final example, we search for the bent cylinder in the image of a toy padlock shown in Figure 11; the results are shown in Figure 12. Searching for a block in the same image yields the results also shown in Figure 12.

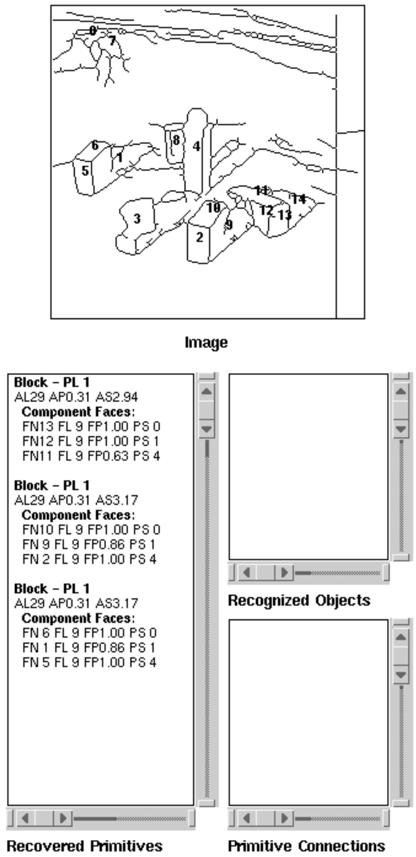

Figure 8: The Best Three Instances of a Qualitative Block

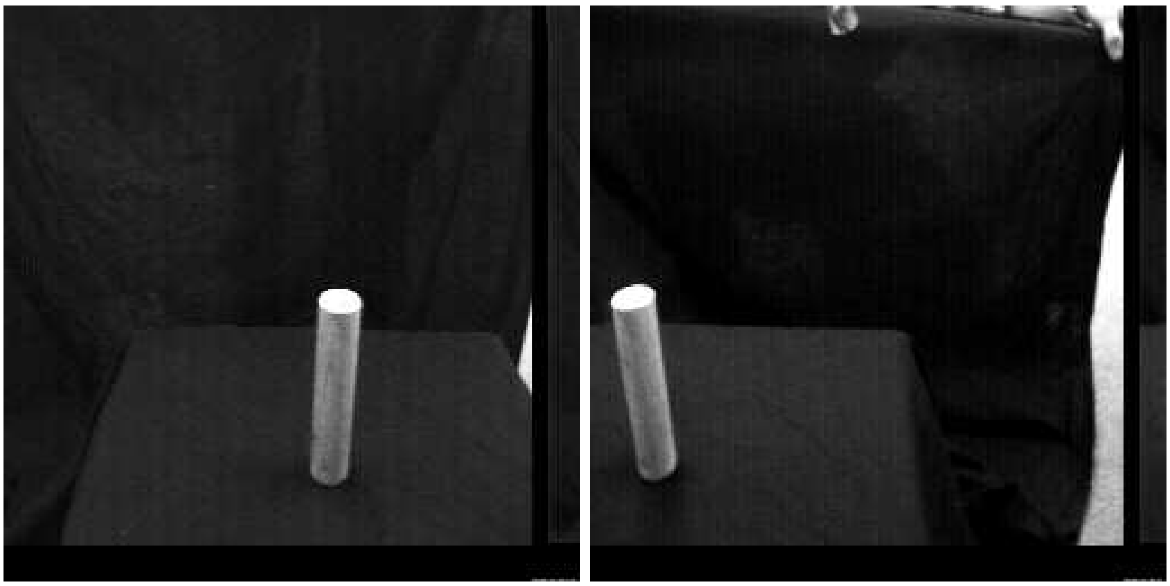

Figure 9: Left and Right Stereo Images of a Cylinder

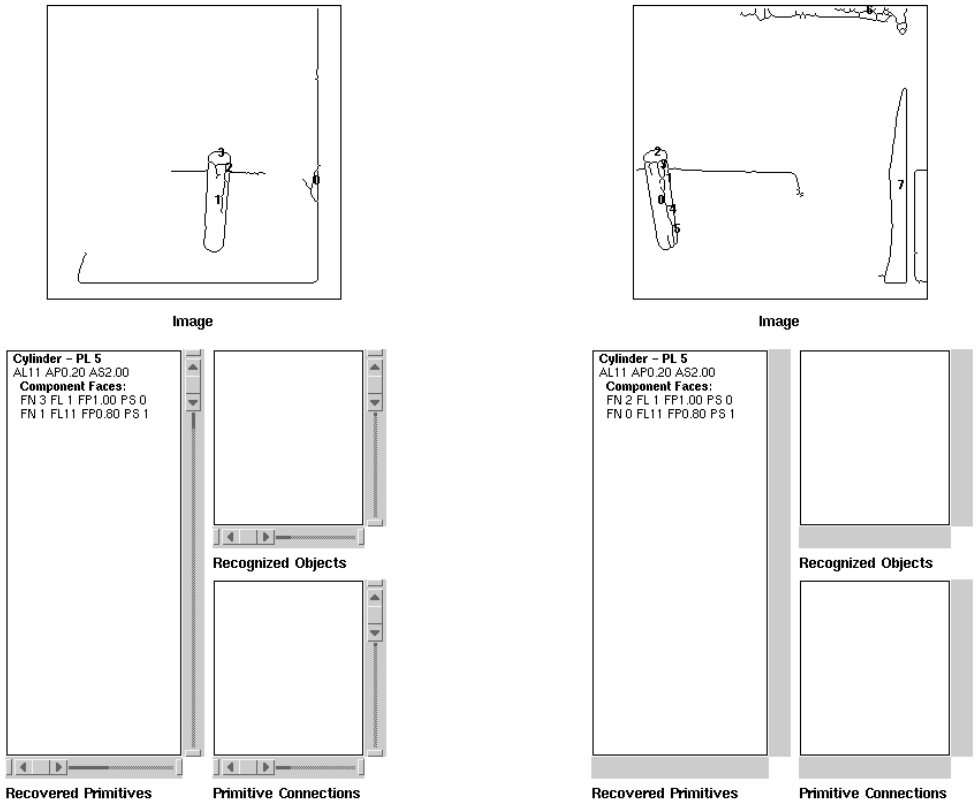

Figure 10: The Best Instance of a Cylinder in a Stereo Pair

Figure 11: Image of a Lock (256 x 256)

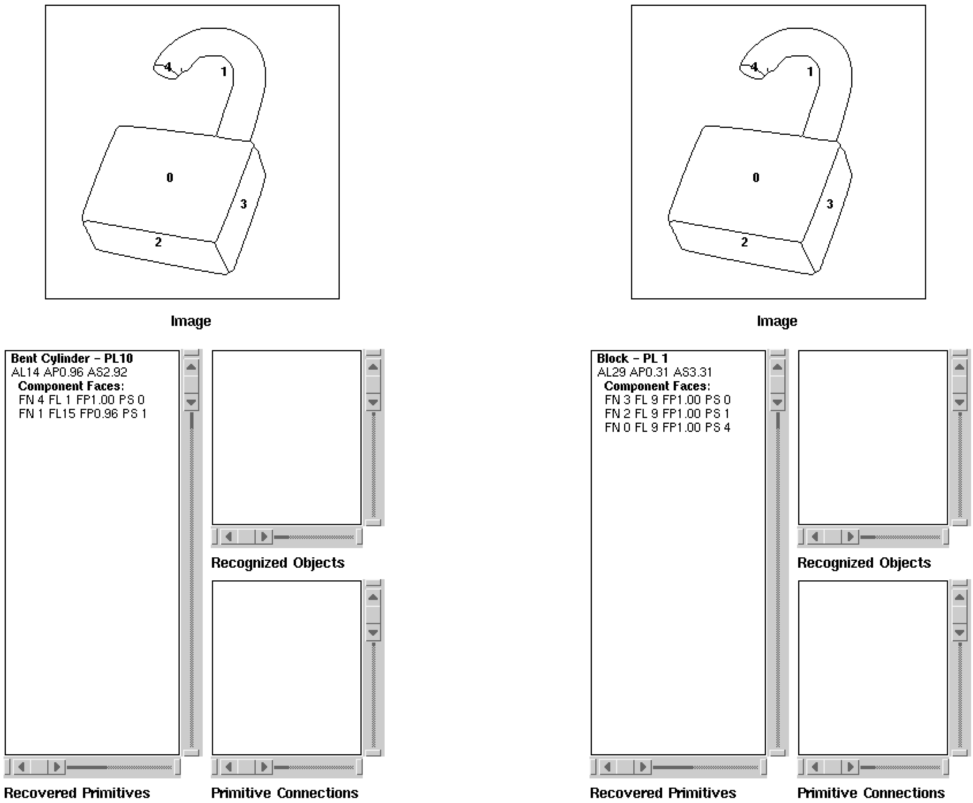

Figure 12: The Best Instances of a Bent Cylinder and a Block

7 Conclusions

Our earlier approaches to 3-D object recognition from a single 2-D image tried to shift the burden of recognition from the top-down verification of simple geometric image features, requiring accurate pose verification, to the bottom-up recovery of qualitative 3-D volumetric primitives. Although effective given no a priori knowledge of scene contents, there was too much effort expended in bottom-up shape recovery for the task of top-down target recognition. In this paper, we have proposed a middle ground in terms of target features, namely labeled regions, or faces, that are not only recoverable from an image in the presence of noise and occlusion, they form the building blocks of our recognition-by-parts paradigm. Furthermore, the face recovery process assigns a confidence value to each face label which can be exploited by an attention mechanism which focuses on not only the most discriminating components of an object's parts, but those with the highest quality or confidence. The technique does not assume a perfect region segmentation of the image, but does assume that some of the regions are intact and can be used to focus the search for the target object. We are currently investigating ways of coping with more severe cases of region under- and oversegmentation.

References

[1] R. Bergevin and M. Levine. Generic object recognition: Building coarse 3D descriptions from line drawings. In *Proceedings, IEEE Workshop on Interpretation of 3D Scenes*, pages 68–74,

Austin, TX, 1989.

[2] I. Biederman. Human image understanding: Recent research and a theory. *Computer Vision, Graphics, and Image Processing*, 32:29–73, 1985.

[3] I. Biederman, J. Hummel, P. Gerhardstein, and E. Cooper. From images edges to geons to viewpoint invariant object models: A neural net implementation. In *Proceedings, SPIE Applications of Artificial Intelligence X: Machine Vision and Robotics*, pages 570–578, Orlando, FL, 1992.

[4] R. Brooks. Model-based 3-D interpretations of 2-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):140–150, 1983.

[5] I. Chakravarty and H. Freeman. Characteristic views as a basis for three-dimensional object recognition. In *Proceedings, SPIE Conference on Robot Vision*, pages 37–45, Arlington VA, 1982.

[6] S. Dickinson, , and A. Pentland. A unified approach to the recognition of expected and unexpected geon-based objects. In *Proceedings, SPIE Applications of AI X: Machine Vision and Robotics, special session on "Recognition by Components"*, Orlando, FL, April 1992.

[7] S. Dickinson, A. Pentland, and A. Rosenfeld. A representation for qualitative 3-D object recognition integrating object-centered and viewer-centered models. Technical Report CAR-TR-453, Center for Automation Research, University of Maryland, 1989.

[8] S. Dickinson, A. Pentland, and A. Rosenfeld. A representation for qualitative 3-D object recognition integrating object-centered and viewer-centered models. In K. Leibovic, editor, *Vision: A Convergence of Disciplines*. Springer Verlag, New York, 1990.

[9] S. Dickinson, A. Pentland, and A. Rosenfeld. The recovery and recognition of three-dimensional objects using part-based aspect matching. Technical Report CAR-TR-572, Center for Automation Research, University of Maryland, 1991.

[10] S. Dickinson, A. Pentland, and A. Rosenfeld. From volumes to views: An approach to 3-D object recognition. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 55(2), 1992.

[11] S. Dickinson, A. Pentland, and A. Rosenfeld. 3-D shape recovery using distributed aspect matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):174–198, 1992.

[12] D. Eggert and K. Bowyer. Computing the orthographic projection aspect graph of solids of revolution. *Pattern Recognition Letters*, 11:751–763, 1990.

[13] R. Fairwood. Recognition of generic components using logic-program relations of image contours. *Image and Vision Computing*, 9(2):113–122, 1991.

[14] T. Fan, G. Medioni, and R. Nevatia. Recognizing 3-D objects using surface descriptions. In *Proceedings, Second International Conference on Computer Vision*, pages 474–481, Tampa, FL, 1988.

[15] Z. Gigus, J. Canny, and R. Seidel. Efficiently computing and representing aspect graphs of polyhedral objects. In *Proceedings, Second International Conference on Computer Vision*, pages 30–39, 1988.

[16] Z. Gigus and J. Malik. Computing the aspect graph for line drawings of polyhedral objects. In *Proceedings, Computer Vision and Pattern Recognition*, pages 654–661, Ann Arbor, MI, 1988.

[17] K. Ikeuchi and T. Kanade. Automatic generation of object recognition programs. *Proceedings of the IEEE*, 76:1016–1035, 1988.

[18] A. Jacot-Descombes and T. Pun. A probabilistic approach to 3-D inference of geons from a 2-D view. In *Proceedings, SPIE Applications of Artificial Intelligence X: Machine Vision and Robotics*, pages 579–588, Orlando, FL, 1992.

[19] B. Jang and R. Chin. Analysis of thinning algorithms using mathematical morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(6):541–551, 1990.

[20] F. Jensen, H. Christensen, and J. Nielsen. Baysian methods for interpretation and control in multiagent vision systems. In K. Bowyer, editor, *Application of AI X: Machine Vision and Robotics*, volume 1708, pages 536–548, Orlando, FL, April 1992. SPIE, SPIE.

[21] J. Koenderink and A. van Doorn. The internal representation of solid shape with respect to vision. *Biological Cybernetics*, 32:211–216, 1979.

[22] D. Kriegman and J. Ponce. Computing exact aspect graphs of curved objects: Solids of revolution. *International Journal of Computer Vision*, 5(2):119–135, 1990.

[23] J. Lee, R. Haralick, and L Shapiro. Morphologic edge detection. *IEEE Journal of Robotics and Automation*, RA-3(2):142–155, 1987.

[24] T. Levitt, J. Agosta, and T. Binford. Model based influence diagrams for machine vision. In M. Herion, R. Shacter, L. Kanal, and J. lemmer, editors, *Uncertainty in Artificial Intelligence 5*, volume 10 of *Machine Intelligence and Pattern Recognition Series*, pages 371–388. North Holland, 1990.

[25] D. Lowe. *Perceptual Organization and Visual Recognition.* Kluwer Academic Publishers, Norwell, MA, 1985.

[26] J. Pearl. *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann Publishers, Inc., 1987.

[27] A. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28:293–331, 1986.

[28] H. Plantinga and C. Dyer. Visibility, occlusion, and the aspect graph. *International Journal of Computer Vision*, 5(2):137–160, 1990.

[29] N. Raja and A. Jain. Recognizing geons from superquadrics fitted to range data. *Image and Vision Computing*, 10(3):179–190, 1992.

[30] R. Rimey and C. Brown. Where to look next using a bayes net: Incorporating geometric relations. In G. Sandini, editor, *European Conference on Computer Vision (ECCV)*, volume 588 of *Lecture Notes in Computer Science*, pages 542–550. Springer Verlag, May 1992.

[31] P. Saint-Marc and G. Medioni. Adaptive smoothing: A general tool for early vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):514–529, 1991.

[32] L. Shapiro and H. Lu. Accumulator-based inexact matching using relational summaries. *Machine Vision and Applications*, 3:143–158, 1990.

[33] L. Stark, D. Eggert, and K. Bowyer. Aspect graphs and nonlinear optimization. In *Proceedings, Second International Conference on Computer Vision*, pages 501–507, Tampa, FL, 1988.

[34] J. Stewman and K. Bowyer. Creating the perspective projection aspect graph of polyhedral objects. In *Proceedings, IEEE Second International Conference on Computer Vision*, pages 494–500, Tampa, FL, 1988.

[35] J. Stewman and K. Bowyer. Direct construction of the perspective projection aspect graph of convex polyhedra. *Computer Vision, Graphics, and Image Processing*, 51:20–37, 1990.