Qualitative 3-D Shape Reconstruction using Distributed Aspect Graph Matching*

Sven J. Dickinson[1] Alex P. Pentland[2]
Azriel Rosenfeld[1]

[1]Center for Automation Research
University of Maryland
College Park, MD 20742-3411

[2]Vision and Modeling Group
Media Laboratory, Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

We present an approach to 3-D primitive reconstruction that is independent of the selection of volumetric primitives used to model objects. The approach first takes an arbitrary set of 3-D volumetric primitives and generates a hierarchical aspect representation based on the projected surfaces of the primitives; conditional probabilities capture the ambiguity of mappings between levels of the hierarchy. The integration of object-centered and viewer-centered representations provides the indexing power of 3-D volumetric primitives, while supporting a 2-D matching paradigm for primitive reconstruction [DIC90a]. Moreover, the prohibitive number of aspects needed to model large databases of objects is avoided.

The reconstruction algorithm exploits all levels of the aspect hierarchy and fully accommodates primitive occlusion. We present a novel formulation of the problem based on grouping the image regions according to aspect. No domain dependent heuristics are used; we exploit only the probabilities inherent in the aspect hierarchy. For a given selection of primitives, the success of the heuristic depends on the likelihood of the various aspects; best results are achieved when certain aspects are more likely, and fewer primitives project to a given aspect. For a common set of 3-D volumetric modeling primitives, the algorithm yields favorable results.

*The support of the Air Force Office of Scientific Research under Contract AFOSR-86-0092 is gratefully acknowledged.

1 Introduction

Many approaches to 3-D object recognition, e.g., [LOW85, HUT87, THO87, LAM88], limit the bottom-up feature extraction process to simple 2-D primitives such as line segments, corners, zeros of curvature, and 2-D perceptual structures. These features are appealing due to their viewpoint invariance. However, because of the simplicity of these 2-D features, a typical 3-D model contains a large number of features. Consequently, the process of searching a database to recognize a model becomes inefficient. Furthermore, the simplicity of the features makes recognition unreliable, and detailed verification of the model's pose is required. Such verification is not only expensive, but restricts the recognition system to models whose exact geometry is known beforehand.

Our approach is to use more complex features and primitives, so that indexing for recognition is efficient, and only qualitative (topological) verification is required. This approach shifts the burden from top-down verification to the bottom-up extraction and grouping of features into complex primitives. This process would normally entail a high search cost due to the complexity of the features and primitives. We have been able to avoid this problem while making use of more complex features, by first assessing and then taking advantage of the *statistical properties* of whatever set of modeling primitives the user has chosen.

In our system, the user first inputs a set of volumetric modeling primitives (polyhedra, generalized cylinders, superquadrics, etc.). The system then defines a set of image features based on these primitives. Such features include the qualitative shape of the primitives' faces, connectivity between subsets of the edges that bound the faces, and groups of faces. The statistical relations between these various features are then assessed from *all* viewpoints, thus generating a table of conditional probabilities for each feature and primitive as a function of all other features or primitives. For instance, one entry in this table might be the conditional probability that we are viewing a cylinder given that we have found a rectangle in the image.

Given an image of a scene, this table of conditional probabilities is then used to guide a combinatorial search that eventually yields a full and consistent interpretation of the viewed scene. The key idea, then, is that the statistical properties of the set of user-defined primitives are used to avoid a combinatorial explosion in the search process. Thus knowledge about how each primitive and feature looks from all angles makes for a "smart" search, and allows the use of much more complex features than can otherwise be employed. The use of such complex features and primitives establishes the foundation for a more robust recognition system, one that can accommodate unexpected objects [ROS86].

2 Building the Search Tables

2.1 Choosing the 3-D Primitives

The goal of the object-centered modeling component is to define a set of three-dimensional volumetric primitives that, when assembled together, comprise a large set of concrete objects in the world. The primitives, in turn, will be mapped into a set of viewer-centered aspects. It is important to note that our approach to primitive reconstruction is *independent* of the selection of modeling primitives; any selection of 3-D volumetric primitives will suffice. To demonstrate our approach, we have selected an object representation based on Biederman's Recognition by Components (RBC) theory [BIE85]; our ten primitives are shown in Figure 1. To construct objects, the primitives are simply attached to one another with the restriction that any junction of two primitives involves exactly one attachment surface from each primitive; details can be found in [DIC90a].

2.2 Defining the 2-D Aspects

For each of the ten primitives, we define a set of 2-D characteristic views, or aspects. Each aspect represents a set of topologically equivalent views of the primitive. Unlike traditional aspect-based recognition systems, e.g., [CHA82, IKE88, FAN88], which model each entire object using a distinct set of aspects, we

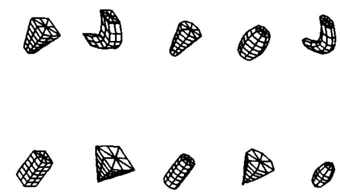

Figure 1: The ten object modeling primitives

use aspects to model the the finite set of primitive classes from which objects are constructed. The size of the resulting set of aspects is fixed and, more important, *independent* of the size of the object database. To minimize the number of aspects representing model primitives, we constrain the set of aspects to be invariant to minor changes in primitive shape. This, in turn, constrains our primitives to possess a qualitative nature, capturing only the gross shape characteristics of the object.

Because of occlusion and errors in region/line finding, the entire aspect of a primitive will often not be available. To overcome this problem, we define a hierarchical aspect representation. This *aspect hierarchy* consists of three levels:

- **face structures**: the set of all aspects of the 3-D primitives, where each aspect consists of a collection of 2-D faces representing the 3-D surfaces of a primitive visible from one viewpoint
- **faces**: the set of all 2-D faces comprising the face structures
- **face features**: the set of all subsets of 2-D contours bounding the faces

Figure 2 illustrates a portion of the aspect hierarchy.

2.3 Relating the 2-D Aspects to the 3-D Primitives

A given face feature may be common to a number of faces. Similarly, a given face may be a component of a number of face structures, while a given face structure may be the projection of a number of primitives. To capture these ambiguities, a table maps face features to faces, while another table maps faces to face structures. To tie together the viewer-centered and object-centered representations, we define a third table mapping the top level of the aspect hierarchy, the

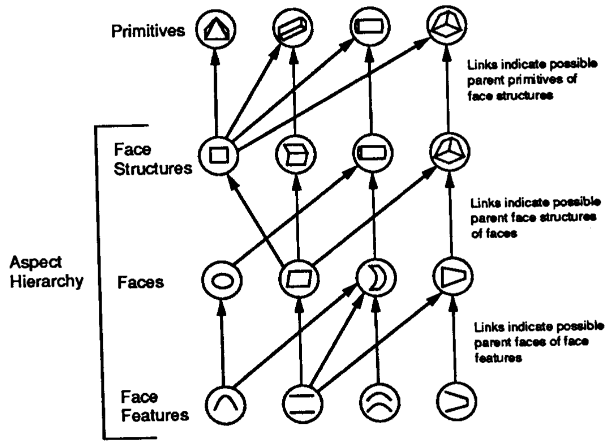

Figure 2: The aspect hierarchy

face structure level, to the primitives. The entries in the three tables represent the probabilities of the mappings.

To generate the probabilities in the tables mapping face features to faces, faces to face structures, and face structures to primitives, we first model our 3-D volumetric primitives using the SuperSketch modeling tool. SuperSketch models each primitive with a superquadric surface subject to deformation. The next step in generating the probability tables involves rotating each primitive about its internal x, y, and z axes in 10° intervals. The resulting quantization of the viewing sphere gives rise to 648 different views per primitive. However, we can exploit symmetries of the primitives to significantly reduce the number of views (688 views for all ten primitives). For each view, we orthographically project the primitive onto the image plane. The final step (currently accomplished manually) involves noting each feature (face feature, face, and face-structure) and its parent; the resulting frequency distribution gives rise to the three probability tables.

3 Primitive Reconstruction

3.1 Extracting Faces

An analysis of the aspect hierarchy probabilities reveals that the face-structure-to-primitive mapping is the least ambiguous mapping to the primitives. In addition, we conclude that the best mapping to the face structures is from the faces rather than from the face features. This suggests that faces are an appropriate starting point in the primitive reconstruction process. Therefore, given an input image, our first task is to segment the regions; the result is a *face graph* in which nodes represent faces, or regions, and arcs represent face adjacency. In turn, each face is represented by a *face contour graph* in which nodes represent straight, convex, or concave bounding contours, and arcs represent relations between the contours, including intersection, parallelism, and symmetry.

The classification of an image face consists of comparing its face contour graph to those face contour graphs representing the faces in the aspect hierarchy. If, for a face contour graph representing an image face, there is a matching face in the aspect hierarchy, we generate a *face hypothesis* for that image face, with probability 1.0. If, however, due to occlusion, there is no match, we must descend to the face feature level of the aspect hierarchy. In this case, we compare subgraphs of the face contour graph to *face feature contour graphs* in the aspect hierarchy. For each subgraph that matches an aspect hierarchy face feature, we use the face-feature-to-face table to hypothesize a face with corresponding probability.

3.2 Extracting Face Structures

Given a face graph with each face having one or more face hypotheses ranked in decreasing order of probability, we can formulate the problem of extracting face structures as follows: Given a face graph and a set of face hypotheses at each face, find a covering of the face graph using face structures in the aspect hierarchy, such that no face is left uncovered and each face is covered by only one face structure. Or, more formally: Given an input face graph, FG, partition the vertices (faces) of FG into disjoint sets, $S_1, S_2, S_3, ..., S_k$, such that each set, S_i, is isomorphic to some face structure graph, FS_j, from a fixed set of face structure graphs, $FS_1, FS_2, FS_3, ..., FS_n$.

3.2.1 Problem Complexity

The *Partition into Isomorphic Subgraphs* problem [GAR79] is stated as follows: Given two graphs, $G = (V, E)$ and $H = (V', E')$, can the vertices of G be partitioned into q disjoint sets, $V_1, V_2, V_3, ..., V_q$, such that, for $1 \leq i \leq q$, the subgraph of G induced by V_i is isomorphic to H? Kirpatrick and Hell [KIR78] prove that this problem remains NP-complete for any fixed H, $|H| \geq 3$. Whether or not this latter problem is NP-complete for planar G and H remains an open problem. However, Berman et al. [BER90] have recently shown that the problem is NP-complete for any connected outerplanar H, $|H| \geq 4$, and that the problem is solvable in linear time for any triangulated H, $|H| \geq 4$. Since our face structure face graphs are neither outerplanar nor triangulated, the complexity of our problem remains open. If the ***Partition into Isomorphic Subgraphs*** problem is NP-complete, a simple reduction reveals that our problem is also NP-complete. Since there is no known polynomial time solution to our problem, we must explore heuristic techniques.

Each face in the face graph representing an image has a number of associated face hypotheses. For each face hypothesis, we can use the face-to-face-structure table to generate the possible *face structure hypotheses* that might encompass that face; the set of face structure hypotheses for a given face in the face graph can be ranked in decreasing order of probability. We can now reformulate our problem as a search through the space of face structure labelings of the faces in our face graph. Equivalently, we wish to choose one face structure hypothesis from the list at each face, such that the verified face structures completely cover the face graph. In all likelihood, there will be many solutions due to perceptual ambiguities in the image. Since we cannot guarantee that a given solution represents a correct interpretation, we must be able to enumerate in decreasing order of likelihood all solutions until the objects in the image are recognized.

3.2.2 Algorithm for Enumerating Face Structure Coverings

For our search through the possible face structure labelings of the face graph, we employ Algorithm A [NIL80] with a heuristic based on the probabilities of the face structure hypotheses. The different labelings are ordered in the open list according to a value determined by the heuristic function. At each iteration, a labeling, or state, is removed from the open list and checked to see if it represents a solution. The successor states are then generated, evaluated, and added to the open list. The heuristic function has been designed to meet three objectives. First, we wish to favor selections containing high probability face structure hypotheses. Second, we wish to favor selections of face structure hypotheses from which face structures can be verified; this effectively constrains the search to portions of the face graph that have not been covered. Finally, we favor those face structure hypotheses covering more faces; we seek the minimal face structure covering of the face graph.

3.3 Extracting Primitives

Once a face structure covering of the face graph has been found, the next step is to map the face structures in the covering to a set of primitives and determine their connectivity. From the face graph and its associated face structure covering, we construct a *face structure graph* in which nodes represent face structures in the covering and arcs represent face structure adjacency. For each face structure in the face structure graph, we can use the face-structure-to-primitive table to generate a set of *primitive hypotheses*, ranked in decreasing order of probability. Our goal is to choose one primitive hypothesis from the list at each face structure, such that the primitives represent a correct interpretation of the face structure graph; we call such a selection a *primitive covering*. Since there may be multiple primitive coverings, we must enumerate them in decreasing order of likelihood. To enumerate the selections, we employ a variation on the algorithm used to enumerate the face structure coverings.

Given a primitive covering of the face structure graph, our next task is to determine the connectivity relations between the primitives; the resulting *primitive graph*, in which nodes represent primitives and arcs represent primitive interconnectivity, is then compared to the object database during the recognition process. If two face structures are disconnected in the face structure graph, their corresponding primitives are disconnected in the primitive graph. However, if two face structures are connected in the face structure graph, their corresponding primitives are connected in the primitive graph according to the visibility of the connection; a visible connection constitutes a *strong* connection, while an invisible connection constitutes a *weak* connection. The subgraph consisting of strong connections represents the index into the model database.

4 Results

We have built a system to demonstrate our approach to shape reconstruction; the system has been implemented in LISP on a Symbolics^{TM} 3600. We are currently integrating an image region segmentation algorithm due to [MEE90] into our system. However, for the following examples, the input to the system is a manually segmented contour image; all face contour graphs and face feature contour graphs are entered manually. A more detailed discussion of the results can be found in [DIC90b].

The first example presents an application to a simple two-primitive object; the results are shown in Figure 3. The correct face structure covering was found after three iterations of the algorithm for enumerating face structure coverings, while the correct primitive covering was found after one iteration of the algorithm for enumerating primitive coverings. Two of the faces in the image were altered due to the intersection of the two primitives; consequently, face features were used to hypothesize face labels for those faces. The two primitives were found to be strongly connected since their junction is visible; the end of the block is connected to the side of the cylinder.

For the second example, we rotate the first object so that it is viewed degenerately; the results are shown in Figure 4. The correct face structure covering was found after seven iterations, while the correct primitive covering was found after one iteration. The reason that the algorithm did not find the face structure covering sooner is due to the fact that the most likely face structure hypotheses for the two faces belonging to the block primitive could not be verified. The two primitives were again found to be strongly connected; however, the surfaces involved in the connection could not be uniquely determined. Either the end of the block is attached to the side of the cylinder, or the end of the cylinder is attached to the side of the block.

The final example is more complex, containing a variety of primitives; the results are shown in Figure 5. The correct face structure covering was found after six iterations, while the correct primitive covering was found after one iteration. The tapered truncated cone primitive is broken into two primitives since no 3-D collinearity grouping is performed. Strong connections were found between the end of the cylinder and the end of the block, and between the side of the block and the large end of the tapered block.

Given a set of 3-D primitives, the number of iterations the reconstruction algorithm requires to converge on a solution depends on the likelihoods of the views of the primitives comprising the object; the more likely the views, the quicker the convergence. In general, our approach will be particularly successful for those sets of primitives giving rise to nonuniform mapping distributions, i.e. certain features (face features, faces, or face structures) are more likely to appear in the image than others.

5 Acknowledgements

The authors would like to thank Suzanne Stevenson, Larry Davis, Peter Cucka, and Nick Leibovic for insightful discussions and for their comments on earlier drafts of this paper.

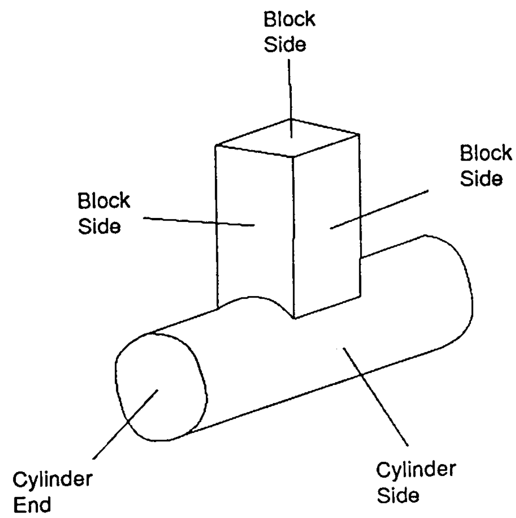

Figure 3: Example 1: three iterations for face structure covering and one iteration for primitive covering

References

[BER90] Berman, Fran, David Johnson, Tom Leighton, Peter W. Shor and Larry Snyder, "Generalized Planar Matching", Journal of Algorithms, Vol. 11, No. 2, 1990, pp 153-184.

[BIE85] Biederman, Irving, "Human Image Understanding: Recent Research and a Theory", Computer Vision, Graphics, and Image Processing, Vol. 32, 1985, pp 29-73.

[CHA82] Chakravarty, Indranil and Herbert Freeman, "Characteristic Views as a Basis for Three-Dimensional Object Recognition", Proceedings, SPIE Conference on Robot Vision, Arlington VA, 1982, pp 37-45.

[DIC90a] Dickinson, Sven J., Alex P. Pentland, and Azriel Rosenfeld, "A Representation for Qualitative 3-D Object Recognition Integrating Object-Centered and Viewer-Centered Models", in: K.N. Leibovic (ed.), *Vision: A Convergence of Disciplines*, Springer Verlag, New York (to appear in Fall 1990).

[DIC90b] Dickinson, Sven J., Alex P. Pentland, and Azriel Rosenfeld, "Qualitative 3-D Shape Recovery Using Distributed Aspect Graph

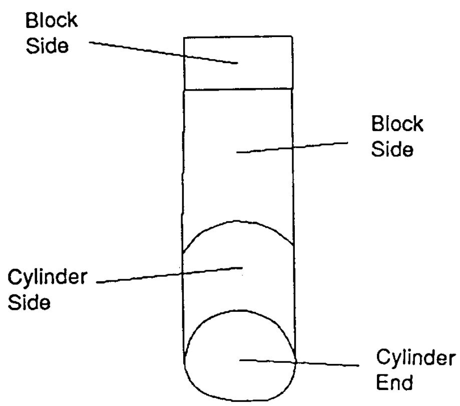

Figure 4: Example 2: seven iterations for face structure covering and one iteration for primitive covering

Matching", Technical Report CAR-TR-505, Center for Automation Research, University of Maryland, June, 1990.

[FAN88] Fan, T.J., G. Medioni, and R. Nevatia, "Recognizing 3-D Objects using Surface Descriptions", Proceedings, Second International Conference on Computer Vision, Tampa, FL, 1988, pp 474-481.

[GAR79] Garey, Michael R. and David S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, New York, 1979, pg 193.

[HUT87] Huttenlocher, Daniel P. and Shimon Ullman, "Object Recognition Using Alignment", Proceedings, First International Conference on Computer Vision, London, UK, 1987, pp 102-111.

[IKE88] Ikeuchi, Katsushi and Takeo Kanade, "Automatic Generation of Object Recognition Programs", Proceedings of the IEEE, Vol. 76, No. 8, pp 1016-1035.

[KIR78] Kirpatrick, D.G. and P. Hell, "On the Complexity of a Generalized Matching Problem", Proceedings, 10th Annual ACM Symposium on Theory of Computing, Association for Computing Machinery, New York, 1978, pp 240-245.

[LAM88] Lamdan, Yehezkel, Jacob T. Schwartz and Haim J. Wolfson, "On Recognition of 3-D

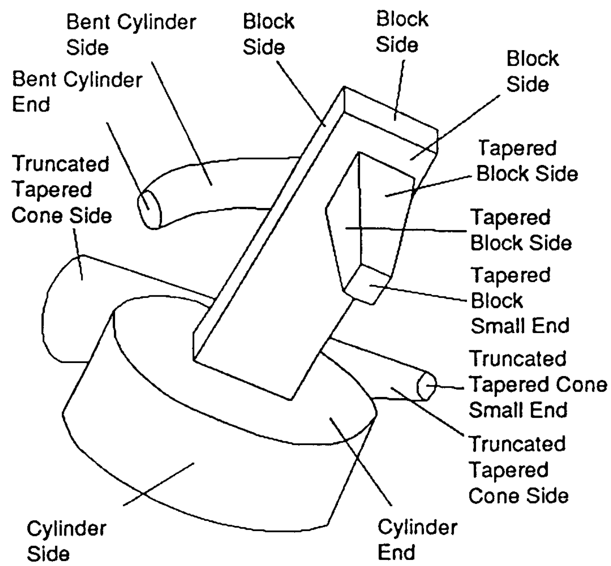

Figure 5: Example 3: six iterations for face structure covering and one iteration for primitive covering

Objects from 2-D Images", Proceedings, IEEE International Conference on Robotics and Automation, Philadelphia, PA, 1988, pp 1407-1413.

[LOW85] Lowe, David G., *Perceptual Organization and Visual Recognition*, Kluwer Academic Publishers, Norwell, MA, 1985.

[MEE90] Meer, Peter, Doron Mintz and Azriel Rosenfeld, "Least Median of Squares Based Robust Analysis of Image Structure", Proceedings, DARPA Image Understanding Workshop, Pittsburgh, PA, 1990.

[NIL80] Nilsson, Nils J., *Principles of Artificial Intelligence*, Chapter 2, Morgan Kaufmann Publishers, Inc., Los Altos, CA, 1980.

[ROS86] Rosenfeld, Azriel, "Recognizing Unexpected Objects: A Proposed Approach", Proceedings, DARPA Image Understanding Workshop, Los Angeles, CA, 1987, pp 620-627.

[THO87] Thompson, D.W. and J.L. Mundy, "Model-Directed Object Recognition on the Connection Machine", Proceedings, DARPA Image Understanding Workshop, Los Angeles, CA, 1987, pp 93-106.