RECOGNIZING THREE-DIMENSIONAL OBJECTS FROM SINGLE PERSPECTIVE VIEWS USING GEOMETRIC AND RELATIONAL REASONING

by

Prasanna G. Mulgaonkar
Linda G. Shapiro
Robert M. Haralick

Department of Computer Science,
Virginia Polytechnic Institute and State University,
Blacksburg, VA 24061

ABSTRACT

In this paper, a scene analysis system, which ecognizes three-dimensional objects from single wo-dimensional perspective projections, is escribed. Given a database of rough relational escriptions of three-dimensional objects and an mage containing a view of one of the objects, the ystem uses geometric information extracted from he image to determine which of the given objects ould have given rise to the observed view. It also roduces an estimate of the camera geometry equired to produce the view along with a numerical elational distance which quantifies how close the bject in the database is to the one in the image. xperiments performed using this system are escribed.

eywords: scene analysis, computer vision, erspective projections, inexact matching, elational description, geometric constraints, onsistent labeling, constraint satisfaction.

Introduction

The problem of recognizing three-dimensional bjects from single perspective views is a non-rivial one. Yet human beings have the capacity to dentify and recognize complex three-dimensional bjects from single photographs. Often, even when he image is degraded with noise, or when much of he structure of the object is obscured, acceptable nterpretations can be found. As an extreme xample, sillhouettes, which contain only the nformation about the outer boundary of an object, an be recognized without much confusion.

Several explanations of this ability have been iscussed from time to time including suggestions hat we possess a remarkable geometric reasoning power that allows us to infer the inverse of three-dimensional projections. A more plausible explanation is that we work with certain expectations about the scene and that we have a database of possible objects from which we choose the most likely candidate. This seems to be a reasonable alternative as evidenced by the fact that we sometimes cannot make judgments from single views of objects we have never seen before. We are most likely to "identify" an unusual view as being caused by something that we have seen or experienced before.

In order to be able to recognize objects from single views, we therefore need the ability to perform several basic operations. First, we need the ability to describe and characterize the three-dimensional objects. Next, we must be able to extract relevant information from the scene. We also need the ability to perform geometric processing on the extracted information in order to compare the view with the three-dimensional objects. In this paper, we describe a system which combines geometric and relational reasoning in order to recognize three-dimensional objects from the outlines of objects in single perspective views.

Description of Three-Dimensional Objects

There exists a large volume of work which has been reported in the field of three-dimensional object modeling and representation. Most of the current techniques build up descriptions of objects from simpler primitives in various ways. Constructive solid geometry (VOE77) systems use set-theoretic "additions" and "subtractions" of solid primitives to assemble objects. Binford (BIN71) first proposed a scheme of decomposing objects into "generalized cylinders" which consisted of cross sections and an axis along which the cross-section was translated. A transformation rule could also be specified which modified the cross section as a function of the translation along the axis. The generalized cylinder modeling was incorporated into a system for performing scene

his research was supported by the National Science oundation under grants MCS-7923827, MCS-7919741 nd MCS-8102872.

82CH1761-6/82/0000/0479$00.75 © 1982 IEEE

analysis experiments by Nevatia (NEV77). This technique was extended to a hierarchic system by Marr and Nishihara (MAR75). The modeling scheme used in our work is the "generalized blob" model proposed by Shapiro et.al. (SHA80a).

The generalized blob model describes a three-dimensional object in terms of the interrelationships among its constituent primitive parts. There are three types of primitives: sticks, plates and blobs. Sticks are predominantly linear features, and plates are planar features which can be idealized as circles in three-space. Blobs are parts which have sizable volume. The model decomposes objects into parts of these three types and describes their relative sizes and the geometric relations that define the connections between them.

Models constructed in this scheme are considered "rough" descriptions of the objects and there is an implied tolerance on every numerical value in the description. This tolerance is not explicitly specified for every part in the structure (c.f. the scheme used by Brookes (BRO81)).

The interrelationships among the primitive parts (hereafter refered to as "primitives") are described in relational terms. The entire description consists of several types of relations. The Simple-Parts relation identifies each primitive with a unique identifier, defines its type (stick, plate or blob), and specifies its size with respect to other parts in the object. The Connects/Supports relation specifies how these individual parts are put together in pairs. For every pair of primitives that touch, it defines what portions of the primitives are in contact, along with geometric information necessary to reconstruct the connection. This information is encoded as a set of angles (measured in degrees). Up to three angles are necessary to qualify all types of connections that can be obtained. The Triples relation describes subassemblies of three parts in terms of their geometric relationships to the middle part. The generalized blob model also has several other relations which were not used in this research and, consequently, will not be discussed here.

The generalized blob model description of three-dimensional objects is scale independent as well as insensitive to rotations about any global axis. There is no global coordinate system. All descriptions are in terms of the logical coordinates of the primitive parts.

Preprocessing the Image

The object of the preprocessing is to extract from the image, a description of the view of the object. This involves performing some type of image processing operations such as smoothing, thresholding and segmentation in order to isolate the regions of interest. In our experiments, we restricted our images to be simple enough that point operators such as thresholding could extract objects of interest. This simplification was made because our major concern was with the matching.

The image is processed to the point where the object is obtained as an isolated region distinct from the background. We currently have the restriction that there can be at most one object in the view. From this segmented image, the outline of the object is extracted using a one pass border tracking algorithm (LAF81). Normally, a boundary extracted in this way has around 600 points for a typical 256x256 image. The number of points is reduced by identifying and extracting corners in the boundary. This is done by means of Davis's K-curvature algorithm (DAV79). The resulting boundary polygon has on the order of 80 to 100 vertices.

The next step in the processing of the image is the decomposition of this boundary point set into simpler near-convex polygons. Conceptually, the parts that are obtained, should correspond to projections of the three-dimensional primitives used for modelling. Two-dimensional shape decomposition algorithms have been reported by several researchers. Feng and Pavlidis (FEN75) reported algorithms which decomposed polygons based on concavities in the boundary. The "Graph-Theoretic Clustering" algorithm used in our work is based on the visibility of boundary points as seen from other points around the boundary. This algorithm is due to Shapiro and Haralick (SHA79).

The extracted near-convex polygons (called "simple parts") and their interconnections are quantified in a relational manner similar to the description of the three-dimensional models. Several properties are extracted from the simple parts which describe their shape. The properties we use are simple shape measures such as the ratio of the area to the longest dimension and the ratio of the area to the perimeter. The measures used are defined below.

$$AL2 = \frac{4 * A}{\pi * L^2}$$

$$P2AINV = \frac{4 * A * \pi}{P^2}$$

where A is the area of the polygon, L is its longest dimension and P is its perimiter.

Although these measures are highly sensitive to noise on the perimeter, our polygons are known a-priori to be near convex and in the process of corner detection, most crenellation has been eliminated. Further our models are themselves rough models, and so we do not require precise shape descriptors in this phase of the matching.

In addition to describing the various simple parts, the data structure built up also contains a two-dimensional connection relation and a two-dimensional triples relation which can be compared to the three-dimensional relations in the models.

The Matching Strategy

In recognizing three-dimensional objects from single two-dimensional views, there are three alternative strategies. The first method is the construction of plausible two-dimensional views from the model followed by two-dimensional to two-dimensional matching. Comparison of two-dimensional polygons has been tried by examining their Fourier descriptions (RUT70) or chain codes (FRE61). Relational matching of polygonal shapes has been reported by Shapiro (SHA80b). However, the number of possible two-dimensional shapes obtainable from a model is very large and consequently, this technique isn't feasible. Another alternative is to map the view into a set of three-dimensional estimated models and perform the matching in three-dimensions. Shapiro et.al. (SHA81) have defined a relational distance measure for comparing relational models. This approach has the problem that the inverse projective transform is not well defined, leading to a large set of possible interpretations.

Our approach is to perform the matching directly from the view to the model. Obtaining three-dimensional information directly from two-dimensional views has been discussed earlier in the context of analysing orthographic views (LAN76). Analysis of single two-dimensional views such as line drawings has been tackled by Waltz (WAL75), Kanade (KAN79) and Chakravarty (CHA79) among others. Brookes (BRO80,BRO81) used symbolic reasoning in the context of recognizing three-dimensional objects from single perspective views. Relational matching traditionally has required large exponential-time searches. The use of discrete relaxation for the matching process was formalized by Rosenfeld et.al. (ROS76). Later Haralick and Elliott (HAR79) examined speedups and tree pruning techniques that could be used for speeding up the tree searches that result in such matching processes.

Overview of Geometric and Relational Constraints in Matching

In matching three-dimensional objects to two-dimensional views, two types of constraints have to be satisfied. These are the geometric and the relational constraints. Geometric constraints control the appearance of each subpart of the object and consequently, of the object as a whole. The appearance of the object parts depends on the geometry of the camera such as its focal ratio and its distance and orientation with respect to the object. Relational constraints determine how the subparts of the view go together.

The problem of setting up a correspondence between the model and the image can be treated as a consistent labeling problem. In this formalism, each three-dimensional subpart (primitive) is postulated as matching a two-dimensional part in the image. The mapping has to satisfy certain consistency requirements. That is to say, no part should end up having mutually incompatible interpretations. If at any point, we assert that certain parts in the model gave rise to certain observed features in the view, we are in effect stating that a camera geometry exists and was used in generating the view. Since the entire object was mapped over to the view using the <u>same</u> camera parameters, we have to accept the fact <u>that</u> all the remaining parts of the model should also map to observed features in the image <u>under the same camera parameters</u>.

We make some assumptions about the camera parameters to simplify the resulting mathematical equations. First we assume that the camera is sufficiently far from the object as compared to the maximum distance in the model. This makes the perspective projection equations closer to those for the normal projection case. Along the lines of Brookes (BRO81), this can be viewed as a perspective-normal projection. Secondly, we assume that the object is being viewed right-side-up. This assumption fixes the roll angle of the camera, leaving us with only two free parameters to describe the camera location. These are the camera pan and tilt angles.

Under the simplified projection mathematics, we can also state what the appearance of our three-dimensional parts must be. Since sticks are linear features, their projections are either linear or disappear from the view. Spheres (our idealized blobs) can only appear as roughly circular artifacts. Plates are modelled as circles. Circles can appear linear (if seen on edge) or circular (if seen face on). For any in-between orientation, plates would look elliptical with the circularity being SIN(TILT), where TILT is the angle between the plane of the plate and the ray joining the center of the plate to the camera lens. This means that if a plate is assumed to map onto some two-dimensional polygon, the measured circularity of the polygon tells us directly what the tilt angle is for the plate.

Note that in the real world, plates are not exact circles. This has the effect of reducing the measured circularity, and consequently the tilt angle computed for the plate is lower than the actual tilt angle.

As a further consequence of the restriction of camera geometry, once two touching plates have been given interpretations, both the pan and the tilt angles of the camera are restricted. These values can be calculated because the geometry of the connection between the two plates is encoded in the angles specified in the Connects/Supports relation in the model. Once the camera position is restricted in this way, it can be used to generate the possible appearance of the remaining parts of the model, and the search mechanism is instructed to find them if possible. If at any stage, such a predicted feature is found to be absent, the process backtracks and tries alternative solutions. If however, the process can find an association for every part, it has found a camera position which causes the model to look like the object in the view.

So far, only the geometric constraints have been discussed. It is not enough for us to find a two-dimensional part in the view which looks like the projection of a part of the model. The part in the view should also participate in all the relations the three-dimensional model dictates. For example, if two sticks touch in the model, their projections in the view should also touch (provided both parts are visible). Every relation that fails to hold in the view is counted as an error, and if at any time the total error exceeds a predefined threshold, we reexamine our previous mappings and try alternate paths.

The Consistent Labeling Problem

In this section we will formalize the consistent labeling problem whose solution is determined by our matching procedure.

Let P be the set of primitives in the model. For each primitive $p \in P$, let T(p) be the type of primitive P, where T(p) is an element of the set {stick, plate, blob} .

Let CS be the connects/supports relation.

$CS \subseteq P \times P \times how \times R^3$ where
$how \in$ {end-end,end-edge, ... surface-surface}
and R is the set of all real numbers.

Let TR be the triples relation.

$TR = P \times P \times P \times side \times R$ where
$side \in$ {same, opposite}.

Let S be the set of simple parts in the view. For each simple part $s \in S$, let C(s) be the circularity of s; where $C(s) \in R[0..1]$.

Let CS' be the two-dimensional connects relation.

$CS' \subseteq S \times S$.

Let TR' be the two-dimensional triples relation.

$TR' \subseteq S \times S \times S \times side$.

Let Tau and Phi be the sets of possible tilt and pan angles, respectively. Let null be a special label to be used when a primitive in the model maps to no simple part in the view.

An epsilon-consistent labeling is a mapping

f : P --> S U {null} X Tau X Phi

that satisfies the following three conditions:

1) Shape constraints

If f(p) = {s, Tau-p, Phi-px} then
a) if T(p) = stick;
then C(s) < C1
where C1 is the circularity threshold for sticks;

b) if T(p) = blob,
then C(s) > C2
where C2 is the circularity threshold for blobs; and

c) If T(p) = plate,
then C(s) = sin(Tau-p).

This states that any stick in the model can only map onto a polygon in the image that has a circularity value less than a pre-specified threshold, and blobs can only map to polygons with a high value of circularity. Note that the circularity measures are normalized to yield a value of 1.0 for circles and 0.0 for lines. Plates can map to features which have a circularity equal to the sin of the tilt angle predicted for the part.

2) View Constraints

If T(p1) = plate and T(p2) = plate;
and if f(p1) = {s1, Tau-1, Phi-12}
and if f(p2) = {s2, Tau-2, Phi-21}

and if $\{p2, p1, how21, A, B, D\} \in CS$;

then
Tau-2 satisfies E(Tau-1, Phi-12, A,B,D)
and Phi-21 satisfies E(Tau-1, Phi-12, A,B,D)

where E is the constraint propagation equation which relates the pan and the tilt angles on one plate to the pan and tilt angles of plates which it touches. For more details about the computational aspects of E, see Mulgaonkar (MUL81).

3) Relational Constraints

$$\text{Let } a = \sum_{i=1}^{\#P} \sum_{\substack{j=i \\ j \neq i}}^{\#P} Xij$$

where

$$Xij = \begin{cases} 1 & \text{if } \{pi,pj,\ldots\} \in CS, \\ & f(pi) = \{si,\ldots\}, \\ & f(pj) = \{sj,\ldots\}, \\ & \text{and } \{si,sj\} \notin CS' \\ 0 & \text{Otherwise.} \end{cases}$$

$$\text{Let } b = \sum_{\substack{k=1 \\ k \neq i}}^{\#P} \sum_{i=k}^{\#P} \sum_{\substack{j=i \\ j \neq i}}^{\#P} Ykij$$

where

$$Ykij \begin{cases} 1 & \text{if } \{pk,pi,pj,\ldots\} \in TR, \\ & f(pk) = \{sk,\ldots\}, \\ & f(pi) = \{si,\ldots\}, \\ & f(pj) = \{sj,\ldots\}, \\ & \{sk,si,sj,\ldots\} \notin TR' \\ & \text{OR} \\ & \text{if } \{pk,pi,pj,sijk,\ldots\} \in TR, \\ & f(pk) = \{sk,\ldots\}, \\ & f(pi) = \{si,\ldots\}, \\ & f(pj) = \{sj,\ldots\}, \end{cases}$$

$$\left| \begin{array}{ll} & C(si) < C1 \quad , \\ & \{sk,si,sk,sijk\} \notin TR' \\ 0 & \text{Otherwise.} \end{array} \right.$$

Then $[a + b] / [\#CS + \#TR] \leq$ Epsilon.

This constraint set gives the error counting procedure for relational errors. It states that the total error is the sum of the number of relations of the model which fail to carry over to the image, normalized by the total number of relations in the model.

Results.

A set of experiments were run using this system to verify the geometric and relational constraint propagation techniques. A database containing eleven object descriptions was generated and used for the testing. The objects in the database were man made in nature and consisted of chairs, tables, desks and other articles of office furniture. Test views were generated from exact three-dimensional descriptions of these objects, generated by computer from known camera positions. Some actual images of doll furniture taken from unknown camera positions were also used in the testing phase. These images were processed as discussed above. All views were compared with all the objects in the database. The results showed that the camera position was estimated to within ten degrees on the tilt angle and twenty degrees on the pan angle for the best mapping case.

In seventeen out of twenty-two cases, the model which mapped to the view with the lowest total error, turned out to be exactly the one from which the view had been generated. The remaining cases were those in which the boundary did not contain enough information to characterize all the parts of the object.

Figure 1 shows one of the models of a table in our database. Figure 2 shows a decomposed view of a table image used in the experiments. The actual view was generated from a camera position given by a tilt angle of 40 degrees from the top of the table. The final mapping obtained by the matching algorithm is given in Figure 3. The error for this mapping was 0.4. One leg of the chair is hidden in the view, and since each leg participates in one connects/supports relation (with the top) and three triples relations (one each with the other three legs), the missing leg gives rise to an error of 4 / 10 = 0.40. The reported tilt angle was 31 degrees for the final mapping shown. The reason it is less than the actual tilt angle is because the table top is not circular but is actually square, and this depresses the measured tilt angle as explained earlier.

Conclusions

This paper shows that scene analysis schemes for recognizing three-dimensional objects from single perspective views using geometric and relational information are viable. Our results show that accurate identification is possible in most cases. Research is currently underway to relax some of the restrictions on possible geometry and to use more information about the models such as foreshortening of sticks and obscuration. Use of interior lines instead of sillhouettes should also improve the results.

REFERENCES

[BIN71] Binford, T., "Visual Perception by Computer", IEEE Systems Science and Cybernetics Conference, Miami, Florida, December 1971.

[BRO80] Brookes, R.A., "Model Based Three-Dimensional Interpretation of Two-Dimensional Images", Technical Report, Stanford University, 1980.

[BRO81] Brookes, R.A., "Symbolic Reasoning Among Three-Dimensional Models and Two-Dimensional Images", Artificial Intelligence, Special Volume on Computer Vision, AI 17, 1981.

[CHA79] Chakravarty, I., "A Generalized line and Junction Labeling Scheme with Applications to Scene Analysis", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1, No. 2, Apr 79.

[DAV79] Davis, L.S., "Shape Matching Using Relaxation Techniques", IEEE Transactions on Pattern Analysis and Machine Intelligence", PAMI-2, No. 3, Mar 81.

[FEN75] Feng, H.Y., Pavlidis, T., "Decomposition of Polygons into Simpler Regions: Feature Generation for Pattern Recognition,", IEEE Transactions on Computers, C-24, 1975.

[FRE61] Freeman, H., "On the Encoding of Arbitrary Geometric Configurations", IEEE Transactions, Electronic Computers, EC-10, 1961.

[HAR79] Haralick, R.M., Elliott, G., "Increasing Tree Search Efficiency for Constraint Satisfaction Problems", Proceedings of th Sixth International Joint Conference on Artificial Intelligence, 1979.

[LAF81] Laffey, T.J, Haralick, R.M., Mulgaonkar, P.G., Shapiro, L.G., "A One Pass Border Tracking Algorithm", Technical Report CS81013-R, Department of Computer Science, Virginia Polytechnic Institute and State University, 1981.

[LAN76] Lanfue, G., "Recognition of Three-Dimensional Objects from Orthographic Views", Proceedings of the Third Annual Conference on Computer Graphics and Image Techniques and Information Processing, Pooch, U.W. (Ed.), 1976.

[MAR75] Marr, D., Nishihara, H.K., "Spatial Disposition of Edges in a Generalized Cylinder Representation of Objects that do not Encompass the Viewer", MIT AI Lab Memo #341, Dec 1975.

[MUL81] Mulgaonkar P.G., "Recognition of Three-Dimensional Objects from Single Perspective Views", Master's Thesis, Department of Computer Science, Virginia Polytechnic Institute and State University, Dec 1981.

[NEV77] Nevatia, R., Binford T.O., "Description and Recognition of Curved Objects", Artificial Intelligence -8, 1977.

[ROS76] Rosenfeld, A., Hummel, R.A., Zuker, S.W., "Scene Labeling by Relaxation Operators", IEEE Transactions on Systems, Man and Cybernetics, CMS-6, No. 6, June 1976.

[RUT70] Rutovitz, D., "Centromere Finding: Some Shape Descriptors for Small Chromosome Outlines", Machine Intelligence, 5, Meltzer, B. and Michie, D., (Ed.), Edinburgh University Press, 1970.

[SHA79] Shapiro, L.G., Haralick, R.M., "Decomposition of Two-Dimensional Shapes by Graph-Theoretic Clustering", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1, No. 1, 1979.

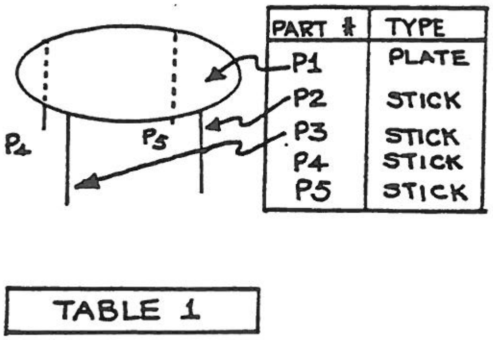
[SHA80a] Shapiro, L.G., Mulgaonkar, P.G., Moriarty, J.D., Haralick, R.M., "A Generalized Blob Model for Three-Dimensional Object Description", Second IEEE Workshop on Picture Description and Management, August 1980.

[SHA80b] Shapiro, L.G., "A Structural Model of Shape", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-2, No. 2, Mar 1980.

[SHA81] Shapiro, L.G., Moriarty, J.D., Haralick, R.M., Mulgaonkar, P.G., "Matching Three-Dimensional Object Models", Proceedings of the IEEE Conference on Pattern Recognition and Image Processing, 1981.

[VOE77] Voelcker, H.B., Requicha, A.A.G., "Geometric Modelling of Mechanical Parts and Processes", COMPUTER, Vol 10, No. 12, Dec 1977.

[WAL75] Waltz, D., "Understanding Line Drawings of Scenes with Shadows", in Psychology of Computer Vision, Winston P. (Ed.), McGraw-Hill, 1975.



PART #	TYPE
P1	PLATE
P2	STICK
P3	STICK
P4	STICK
P5	STICK

Figure 1: Sample Model of a Table

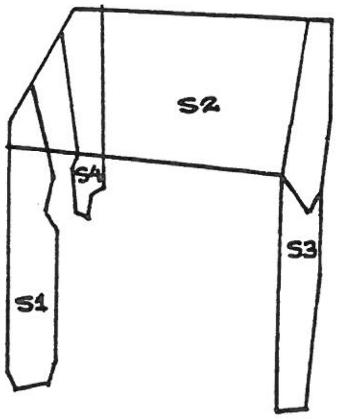


Figure 2: Decomposed View of an image.

FINAL MAPPING RESULTS

PRIMITIVE #	SIMPLE PART #
P1	S2
P2	S3
P3	S1
P4	S4
P5	—
TILT ANGLE	31 (degrees)

ERROR FOR THE MAPPING 0.4

Figure 3: Results of the mapping between the model in Figure 1 and the image in Figure 2.