AD-A037 723 STANFORD RESEARCH INST MENLO PARK CALIF F/G 9/2

HIERARCHICAL REPRESENTATION OF THREE-DIMENSIONAL OBJECTS.(U)

DEC 75 G J AGIN N00014-71-C-0294

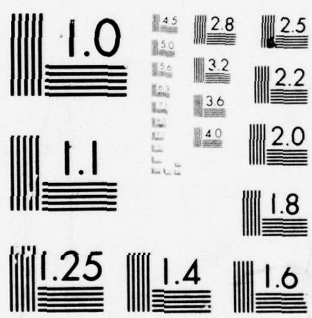
UNCLASSIFIED NL

| OF |

AD
A037723

END

DATE
FILMED
4-77


1.0
4.5
5.0
5.6
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

Annual Report

December 1975

31 p.

HIERARCHICAL REPRESENTATION OF THREE-DIMENSIONAL OBJECTS

By: GERALD J. AGIN

Prepared for:

OFFICE OF NAVAL RESEARCH
800 NORTH QUINCY STREET
ARLINGTON, VIRGINIA 22117

Contract Monitor: Marvin Denicoff, Program Director
Information Systems Branch

Contract N00014-71-C-0294

STANFORD RESEARCH INSTITUTE
Menlo Park, California 94025 · U.S.A.

STANFORD RESEARCH INSTITUTE
Menlo Park, California 94025 · U.S.A.

Annual Report *December 1975*

HIERARCHICAL REPRESENTATION OF THREE-DIMENSIONAL OBJECTS

By: GERALD J. AGIN

Prepared for:

OFFICE OF NAVAL RESEARCH
800 NORTH QUINCY STREET
ARLINGTON, VIRGINIA 22117

Contract Monitor: Marvin Denicoff, Program Director
Information Systems Branch

Contract N00014-71-C-0294

SRI Project 1187

Approved by:

BERTRAM RAPHAEL, *Director*
Artificial Intelligence Center

EARLE JONES, *Acting Executive Director*
Information Science and Engineering Division

ABSTRACT

This report summarizes research on methods for representing within a computer the shapes of common objects that a robot or intelligent computer would have to deal with. Such a representation should be capable of supporting man-machine communication based on words and on pictures. It should also provide a basis for direct interaction of a machine with its environment, using sensors such as television or a range finder.

As a vehicle for exploring these kinds of interaction we used a hierarchical, polyhedral representation to model electromechanical machinery. One feature of the method used was that the spatial relationships of one part to another could be characterized by "attachment points" located on each object. Symbolic descriptions were translated into geometric descriptions in terms of planes, edges, and points, from which visible outlines and occlusion relationships could be derived.

We were successful in demonstrating computer vision based on these models. Using a laser range finder we showed how to detect the presence or absence of pieces of an assembly, and were able to precisely establish the position and orientation of an air compressor on a tabletop. We were able to segment a conventional TV image into regions corresponding to the major subassemblies of the same compressor.

CONTENTS

ILLUSTRATIONS

I INTRODUCTION

This report summarizes research on methods for representing within a computer the shapes of common objects that a robot or intelligent computer would have to deal with. Such a method should be capable of supporting man-machine communication based on words and on pictures. It should also provide a basis for direct interaction of a machine with its environment, using sensors such as television or a range finder

We distinguish between two kinds of man-machine communication. We call communication with words semantic interaction. Since the analysis of natural English is a difficult task, we rely on the programming language LISP to convey semantic information without syntactic ambiguity. Important concepts are communicated with words, such as the names of objects, spatial and part/whole relationships among articles, and most importantly, notions of similarity. Very little effort has gone into quantifying similarity of three dimensional shapes; the most significant work has been reported by Winston [1].*

Communication using pictures we term graphic interaction. The machine may be called upon to draw a particular object for a human user to interpret. With suitable facilities, the user can indicate particular points or regions of interest, using a cursor or light pen Pictures are a natural medium for conveying shape information: "A picture is worth a thousand words". Many of the computer-aided design programs that have been demonstrated to date are heavily graphics oriented [2-4].

Detecting an environment through video or range sensors and then making sense of the data is the problem of computer vision. A variety of techniques have been demonstrated that analyze range data inferred from laser triangulation [5-7], "grid coding" [8], stereo correlation

----

* References are listed at the end of the report

[9], motion parallax [10,11], and reflectance assumptions [12]. These various programs build models to "explain" the range data they obtain, representing shape in different ways. but only in the work of Nevatia [7] does the computer manage to "recognize" the objects it sees, or to do more than simply transform one representation into another. Scene "understanding is the holy grail we seek, and recognition of isolated objects is the first step in its direction.

To explore the issues of semantic, graphical, and visual interaction with shape information, we made use of a polyhedral representation originally designed for the ARPA-sponsored Computer-Based Consultant project. This representation had been intended to model electromechanical machinery. One feature of the method was that the spatial relationships of one part to another could be characterized by "attachment points" located on each object. Symbolic descriptions could be translated into geometric descriptions in terms of planes, edges, and points, from which visible outlines and occlusion relationships may be derived.

We were successful in demonstrating computer vision based on these models. Using a laser range finder we showed how to detect the presence or absence of pieces of an assembly, and were able to precisely establish the position and orientation of an air compressor on a tabletop. We were able to segment a conventional TV image into regions corresponding to the major subassemblies of the same compressor. Section II of this report discusses the polyhedral modeling and its extensions.

The polyhedral representation proved severely limited in the capabilities we needed to extend its semantic and visual performance We are currently in the process of clarifying the requirements of a new representation to be implemented during 1976. Section III summarizes the positive and negative aspects of the particular system we used during 1975 as they relate to the new representation.

II USE OF POLYHEDRAL MODELS FOR MACHINE VISION

In our studies of model-guided computer vision, we were fortunate to have available an already developed tool for representing parts. A geometric modeling system was designed in 1974 for the ARPA-supported Computer Based Consultant (CBC) project. The details of the representation have been reported [13], and are summarized in Section A below. For that project, a primary requirement was the ability to model tools and electromechanical machinery. Such objects have a great deal of regularity and predictability. Dimensions are stable and are frequently known beforehand. While nonrigid members may be found in a typical workstation (fan belts, power cords, gaskets) the major portion of the workstation may be modeled by combinations of rectangular solids and circular cylinders. Braid [2] has shown many examples of machined parts that can be represented by combinations of simple primitive solids.

The modeling system proved useful in two sets of experiments on model-guided computer vision. The first set of experiments, described below in Section B, involved the use of a laser range finder for locating a known part in an unknown position and for verifying the presence or absence of a given part in an assembly. We believe the results achieved are significant, but they also point up some deficiencies of the polyhedral modeling.

The second set of experiments was concerned with the use of the polyhedral models to guide the segmentation of a scene obtained from video. Section C summarizes the procedure and presents some results.

Section D details a minor improvement made to the modeling system, to enhance its semantic abilities. This consisted of a more natural way of specifying relative positions and orientations, based on the homogeneous coordinate method, but with a facility for symbolic manipulations as well as numeric ones.

A. THE CBC SYSTEM

The system, as it existed at the beginning of 1975, consisted of a data structure in which parts could be conveniently described, together with some computer programs for manipulating the models. The modeling system had four principal components:

(1) A set of routines to manipulate semantic descriptions. They work with objects described as hierarchical compositions of subparts. The relative spatial positions of the subparts are included as part of a description. The routines evaluate parameters, explore the hierarchical structure, create copies where appropriate, and compute the absolute position of each subpart.

(2) A set of routines to transform semantic descriptions of primitives (dimensions and absolute position) into geometric, polyhedral descriptions (faces, edges, and vertices). The basic primitives are a rectangular solid, a right circular cylinder, and a wedge. A cylinder is approximated by a prisms of eight sides.

(3) A set of display routines that work with descriptions of polyhedral edges, drawing them in perspective from an arbitrary point of view.

(4) A set of routines that compute the silhouettes of objects represented as faces, edges, and vertices, to determine which part is in front of which when viewed in perspective, and to locate the center of the visible outline of a part.

The data structure in which parts are initially described to the system has several useful and unique characteristics. The description of parts and their relationships is in symbolic terms wherever possible. This is facilitated by the use of "attachment points". Each primitive part (that is, brick, wedge, or cylinder) has several places at which other parts may be joined. These points carry labels such as BASE, TOP, BACK, or RIGHTSIDE. We may place part A on top of part B for example, by matching the BASE of Part A with the TOP of Part B. When such a spatial relationship is specified it may be further modified, for example, by sliding Part A 6 inches to the right.

The shape of an air pump may be crudely represented by the following symbolic description:

```
        STRUCTURE
                [(CRANKCASE (BRICK 5.0 3.5 5.5))
                 (PISTON-CYLINDER (BRICK 3.1 3.1 5.0)
                                  (REF CRANKCASE TOP))]
```

Figure 1 illustrates this example. This description says that the pump is the union of two simpler parts, which are assigned the symbolic names CRANKCASE and PISTON-CYLINDER. The CRANKCASE is to be modeled as a rectangular solid, or "brick", of dimensions 5.0 x 3.5 x 5.5 inches. The PISTON-CYLINDER is a brick of dimensions 3.1 x 3.1 x 5.0 inches. The base of the PISTON-CYLINDER is to be placed on top of the CRANKCASE (that is, at the symbolic attachment point named TOP). The CRANKCASE has no explicit position descriptor, and its base will be the same as the base of the PUMP assembly.

The programs will process descriptions such as the above, creating copies of the model descriptions that have actual positions and orientations numerically specified. The copies are transformed into face-edge-vertex polyhedral descriptions, which may in turn be processed by the display subroutines to produce pictures like that of Figure 2. Such a display we call a wire model, because hidden-line elimination is not performed, and polyhedra are drawn with "wires" along each edge. The display can be presented in perspective from an arbitrary viewpoint, and an interactive interface allows rotation, translation, and scaling of the three-dimensional projection.

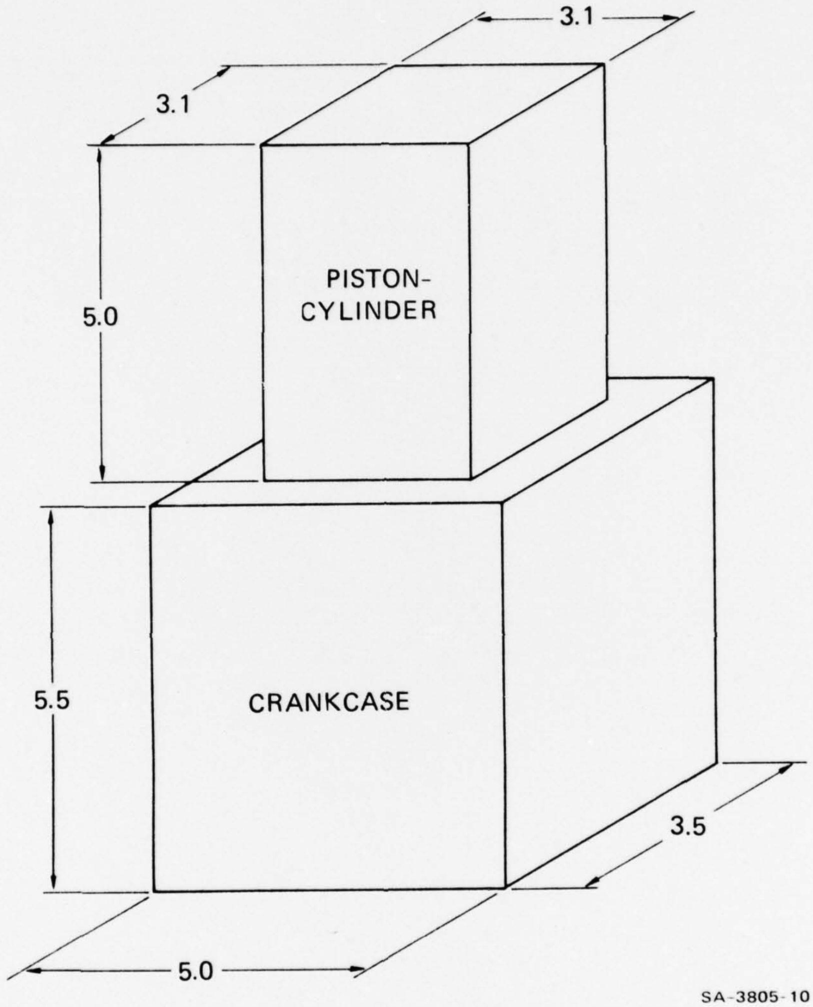
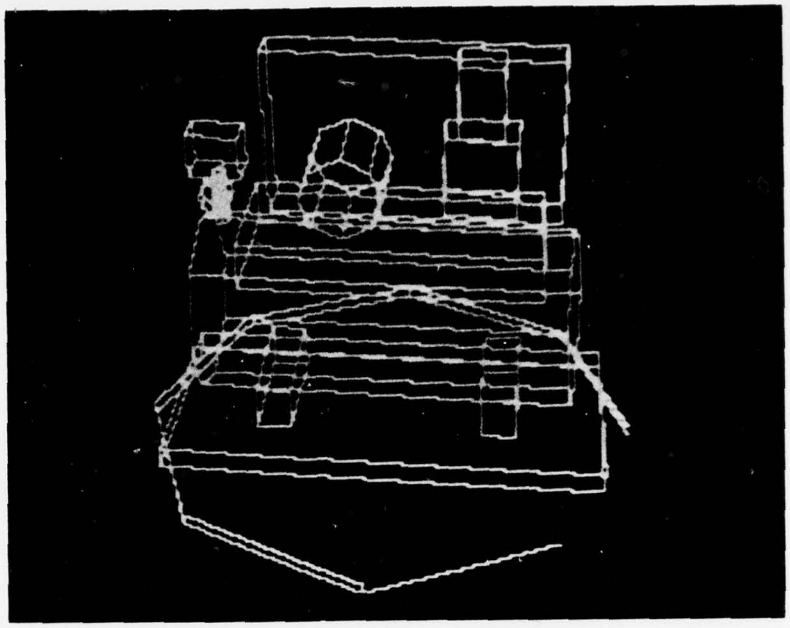


FIGURE 1 GEOMETRIC MODEL OF THE PUMP



SA-3805-47

FIGURE 2 WIRE MODEL OF THE COMPRESSOR

Although we do not perform a complete hidden-line elimination, we have a procedure that can determine for a given point on the display which surface of the model is closest to the viewer. Thus a user might position a cursor on the display screen so that the computer could answer the question: What part is this?

By substituting a TV camera and a tiny light bulb for the display cursor, a very crude "machine vision" can be accomplished. It is necessary that the models and the actual parts correspond very closely and that the transform of the camera be accurately modeled by the perspective transform of the display algorithm. It is easy to detect the light bulb in the TV image, but that is all the vision system "sees". If the compressor were missing, the system would not be able to detect the difference.

Another set of algorithms can calculate the visible silhouette of a given part, taking into consideration any parts that are closer to the camera than the named part and that may hide a portion of the silhouette. The computer may choose a point inside that visible outline at which to place a cursor. Or, if the display transform can be made to agree with the transform of the laser pointer (part of the laser range finder, which is described below), then, with good agreement between the models and the position of the compressor, the laser beam can be made to point to the named part.

B. USE OF THE LASER RANGE FINDER

The models described above were shown to satisfy some of our requirements in the semantic and graphic domains. The obvious next step was to close the loop between the models and the real world, testing the use of the models in computer vision. This section describes a series of experiments that used a laser range finder for obtaining "visual" information. The next section details the integration of the models with TV data.

Initial experiments with the range finder and the models were a simple test for the absence or presence of a specific part on the air compressor assembly. When this test had been *demonstrated to work* properly, we undertook the more difficult job of locating the compressor in the field of view when its position was only approximately known.

The range finder we used has been described [14]. The beam from a helium-neon laser is modulated at 9 MHz and deflected by a steerable mirror assembly so that it can be directed about the room. A photomultiplier tube detects the reflected light when the laser beam illuminates an object. Because of the finite velocity of light, a shift will occur in the phase of the 9-MHz modulation, proportional to the length of the path from the laser to the object and back to the photodetector. This phase shift can be measured, digitized, and fed to the computer as an indication of the range to the laser spot.

The steerable mirrors and the laser together constitute a laser pointer. With its two directions of scan, the pointer has projective geometry similar to that of a television camera. With the addition of the time-of-flight ranging, we have an instrument that can plot a depth map of an entire scene, if desired. But because of the inordinate amount of time required (on the order of an hour for a 128 x 128 scene), for the purposes of this project we have measured range values only at the points where values have been actually needed.

A calibration process (described in [13]) estimates the location and orientation of the steerable mirrors and the sensitivity and offset of the phase measuring equipment, so that the position of the laser spot can be obtained in Cartesian coordinates--x, y, and z with respect to the floor and walls of the room.

The first experiment to integrate the model and the laser was to detect missing parts in an assembly. The basic assumptions for this exercise were that the position of the assembly was accurately known and that the model of the assembly was basically correct, except that a particular part might or might not have been removed.

Briefly stated, the algorithm is to attempt to point the laser beam at the part in question, assuming that it is in place. From the model, an expected range reading can be calculated. The actual range to the laser spot is measured and is compared with the predicted range, to form the basis for a present/absent decision. Ideally, the decision should be based on two predicted values--under either of the the two mutually exclusive assumptions that the part is present or absent. In the actual program, the decision was made on whether the measured range was within a certain empirically derived threshold from the predicted range.

This procedure is nothing more than a quick test of a specific hypothesis, given sufficient information about the environment to accurately point the laser and predict the range to the part.

The method was demonstrated to work most of the time. Within the CBC supervisory system, a request to point at a part was translated to a call to the modeling and pointing system to detect whether the part was present. If we removed the pump from the compressor, then asked the program to point to the pump, the laser beam would point to where it thought the pump ought to be. If the range measured was not approximately equal to that predicted, the system would answer: "The pump is not present."

When the system did err in its judgment, the errors tended to be either of two kinds. In one case, the error in the measurement of range caused the measured range to be out of bounds. The threshold we chose for the discrimination was 6 inches. This choice represented a compromise between the expected dispersion of range measurements under varying conditions and the differences in actual range that result from removing a part. The quality of the range measurements has improved considerably since last March when these tests were performed, but at that time the range errors were occasionally outside the range given.

The other kind of error was due to inaccuracy of the model or the calibration. In general, the model and the real world tended to correspond within about an inch, but rarely better. Sometimes, when pointing at a small subpart, the laser would miss the subpart completely and report it missing when it was actually present. With a smarter, more complex algorithm to execute a search pattern, or to try to locate the part in question, such errors might not have occurred. But, in general, when the uncertainties were of the same size as the part to be sensed, our simple strategy was inadequate.

The second experiment linking the models and the range finder was to obtain the position and orientation of the compressor placed on a table somewhere in the field of view. By exploring with the range finder the system was able to locate the compressor and update its internal models. It would have been satisfying to find the method good enough to correct for miscalibration, but the basic inaccuracies of the range finder limited the precision to a level not as good as that obtainable by ruler-and-plumb-bob methods. Yet the success of the method in spite of the inaccuracies is all the more significant.

The experiment made use of several of the constraints of the environment to simplify the locating algorithm. The height of the tabletop is known, thus providing one constraint in position. The compressor is assumed to be in an upright position, constraining orientation to one degree of freedom. A truly general procedure would need to fix three degrees of freedom in position and three in rotation; here we need find only x and y in position (z or height being known) and angle of rotation about the vertical. Furthermore, the geometry and topology of the object being sought are accurately known. The strategy we used was based on the unique characteristics of the compressor.

A long-range goal for this project is the ability to locate any object in an arbitrary orientation, making use of whatever constraints are known in a given situation. To accomplish this will require much additional work on the models. Hand generating an algorithm to fit a specific situation is the first step toward the more general problem.

The first step in finding the compressor is to find its tank. It is known that the only thing to be found at the level of the middle of the compressor tank is the tank itself. If we search for points at that height, or 40 inches above the floor, then those points must belong to the tank. It is on this concept that we based our location strategy.

Searching with the laser, we must first find a point 40 inches above the floor. We do this by choosing a vertical line near the center of the field of view, and scanning down this line until we find a point whose height is appropriate. (The use of interpolation between points speeds this process.) About six probes with the laser are usually sufficient to find a point within one half inch of the desired height. If the X and Y of this point (i.e., its horizontal position) indicate that the point is near the middle of the room, then we may say with a good degree of confidence that the point is somewhere on the compressor tank. If, on the other hand, the point turns out to be on the wall, we may deduce that the vertical scan misses the compressor. Two additional scans may be tried, one on each side of the initial vertical scan. If one of these succeeds, we proceed as below; otherwise, we must admit failure at this point.

Having found one point on the compressor tank, the next task is to find other points nearby that are on the tank and 40 inches above the floor. The vertical search pattern is moved 2 inches to the right and repeated. (The nearby known point provides an initial estimate, making the search faster and less error prone.) Points are found at 2-inch intervals until the search fails, indicating that the right-hand end of the tank has been found. Starting again from the original point, additional points are located to the left until the left-hand end has been found.

Depending on the position of the compressor, the horizontal positions of the points found should indicate either one surface or two surfaces of the tank. Four cases are possible and are illustrated in Figure 3.

A least-squares straight line fitter will attempt to fit a single line through the points. The fit will succeed for the cases shown in Figure 3(a) and (c). In Figure 3(b) and (d), the line is segmented by drawing a line between the two end points, and choosing the point farthest from that line to be the division point. Straight lines will be fitted to each of these two segments.

It is this fitting process that is the most error-prone of the entire locating algorithm. At the time when these experiments were performed, the average error to be expected from the range-finder measurements was on the order of 1 to 2 inches. This generated considerable error in the fitting, making the decision whether or not to segment the line difficult. It also made choosing a point at which to segment difficult, since the average error was about the same as the distance between points.

The number and relative lengths of the line segments fitted are sufficient to distinguish among the four cases of Figure 3. The parameters of the lines and the geometry of the tank give the horizontal

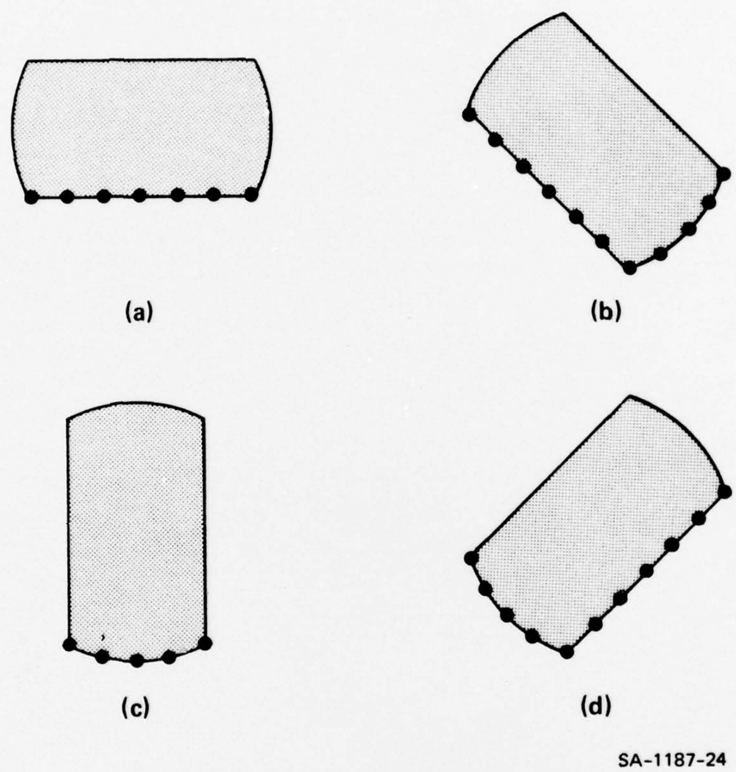


FIGURE 3 POINTS FOUND ON THE SURFACE OF THE TANK

location of the center of the tank and give the azimuth of rotation about the vertical axis.

Because the tank is symmetrical, there remains a 180 degree ambiguity in the orientation of the rest of the compressor. This is resolved by attempting to find the belt housing frame, a large piece of sheet metal on the compressor's superstructure. There are two equally plausible assumptions about the rotation of the compressor. Taking each assumption separately, the system will attempt to measure the range to the belt housing frame. Choosing the assumption that gives the better correspondence between predicted and measured range values is sufficient to complete the analysis.

Although the experiment was limited, we feel that it demonstrates some important principles. The first is that, with enough information to locate a part approximately, a range finder can refine that position estimate, locating the part to a precision limited only by the the accuracy of the range finder and the models. We demonstrated the use of only a single technique: tracing a contour at a fixed height. Other techniques that might be used, depending on the situations, are tracing profiles in other planes, locating depth discontinuities, finding edges and corners, and precisely locating one or more planes in space.

Second, and more important, we demonstrated the use of information about a specific situation in choosing the proper technique to solve a problem. This choice was inherent in the writing of the procedure to locate the compressor. What is yet to be demonstrated is the direct use of the models by an intelligent computer program for such a choice. The solution to that problem lies at the core of artificial intelligence research. It is toward such a solution that we are working.

C. USE OF MODELS IN SCENE PARTITIONING

A second set of experiments in the CBC workstation domain involved using the models to interpret TV gray-scale information, resulting in a partitioning of the image into regions corresponding to the parts of the compressor.

In a separate report [15], J. M. Tenenbaum and H. G. Barrow described a method of segmentation of TV images that makes use of knowledge about possible interpretations of the scene to constrain merging of regions. The image is first broken into a large number of primitive regions of uniform brightness and color. Regions may be assigned one or more "possible" interpretations. A set of constraints codifies how the various local interpretations must remain consistent. Adjacent regions of the segmented image are merged, beginning with those most similar in brightness and color, provided that the merging would not violate the constraints.

In the experiment we describe here, the initial interpretations and the constraints were supplied by the geometric model of the compressor. As in the previous experiment with the laser range finder, some initial assumptions were made about the nature of the scene. The image to be processed was assumed to be a frontal view of the air compressor, so the areas of the image representing specific parts of the compressor could be approximately predicted. The object of the exercise was to segment the scene into regions corresponding to each of the parts of the compressor, locating the precise boundaries between them.

For this experiment, a color TV image of the air compressor was digitized to 6 bits/color at 60 x 60 resolution (Figure 4). This digitized image was then partitioned into elementary regions composed of adjacent pixels with identical brightness, as shown in Figure 5. Because of the uniform coloring of the compressor, typical of mechanical equipment, a nonsemantic region-merging program proved to be highly unsatisfactory. Figure 6, for example, shows the partition that results from successively merging together pairs of adjacent regions with lowest color contrast, until 200 regions remain. It is evident that several significant errors, such as merging of the tank and base into a single region, have already occurred. Although pointless, the merging process obviously could be continued until the entire scene had been included in one big region.

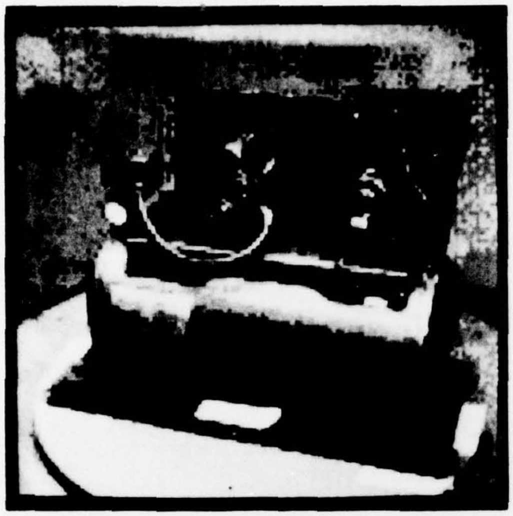


FIGURE 4 DIGITIZED IMAGE OF COMPRESSOR (5 BITS AT 120 x 120 RESOLUTION)

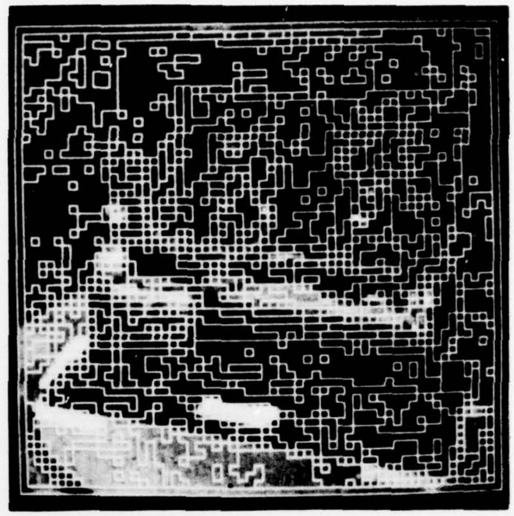


FIGURE 5 INITIAL PARTITION (AT 60 x 60 RESOLUTION); CONTAINS 931 REGIONS

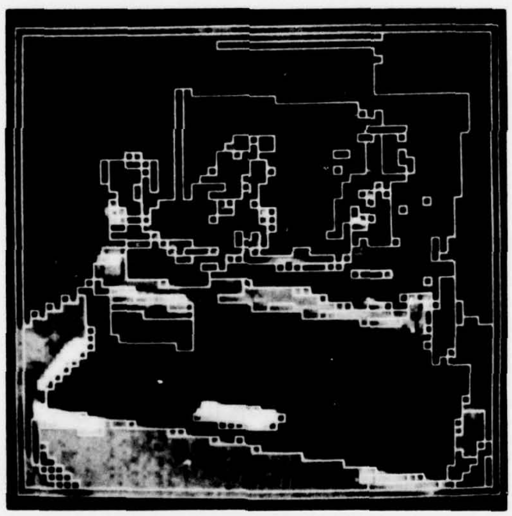


FIGURE 6 UNGUIDED PARTITION WITH ERRORS (200 REGIONS)

SA-4683-12

It was assumed that the relative location and orientation of the camera and compressor were known approximately. The uncertainty in relative position introduces a corresponding uncertainty in the prediction of which compressor component will be visible at a given point in the image The uncertainty in prediction can be represented by a set of overlapping regions, each of which expresses the composite for all compressor positions within the assumed range of uncertainty. Figure 7 shows the composite regions for the seven compressor parts distinguished in this experiment, plus the background. These regions were transcribed manually from a series of displays showing the compressor at various positions over the allowed range. The transcription process, however, would be straightforward to automate.

The overlapping regions shown in Figure 7 were used to assign initial interpretation sets to each pixel. An initial partition was then formed in which all adjacent pixels with identical brightness and interpretations were grouped into regions. Regions were then merged, subject to the existence of at least one common interpretation and to the existence of at least one region for each component part. Merging continued until no more merges were possible under the constraints.

The process terminated with a partition in which all adjacent regions had disjoint interpretations, as shown in Figure 8. Although the result is by no means perfect, it represents a considerable improvement over the attempt at unguided segmentation (Figure 6). Given the low resolution and the lack of color variation, the results are rather good.

The success of this experiment illustrates principles similar to those mentioned in the previous section. Given some knowledge about a scene (an approximate location for the compressor), it is usually possible to refine that information. In this case, the constraints on knowledge are more severe than in the demonstration of laser orientation (Section B), but given that those constraints are satisfied, the geometric models can be used directly and automatically to locate the boundaries.

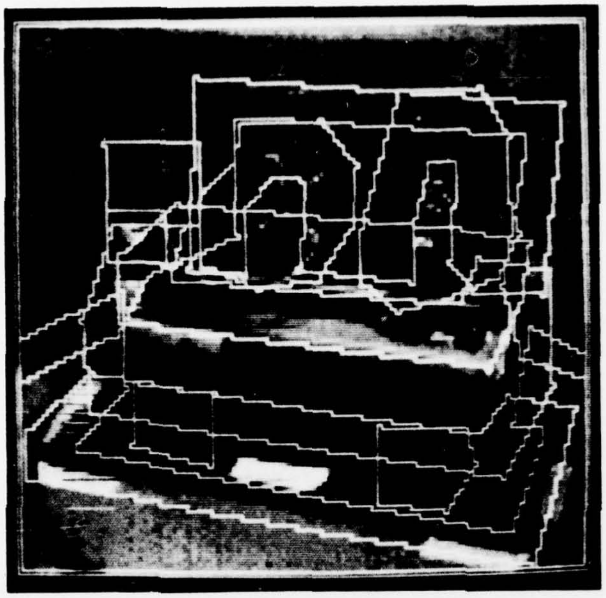
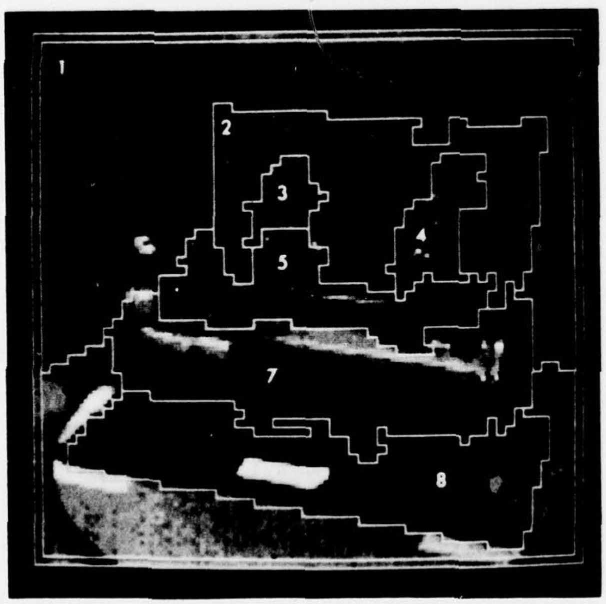


FIGURE 7 COMPOSITE REGIONS DELINEATING POSSIBLE AREAS OF IMAGE FOR EACH INTERPRETATION



Region	Interpretations
1	Background
2	Belt Housing
3	Motor
4	Pump
5	Tank Platform
6	Table
7	Tank Cylinder
8	Base

SA-4683-14

FIGURE 8 *FINAL PARTITION AND LABELS AFTER MODEL-GUIDED MERGING*

The possibilities for other types of techniques based on video images and geometric are many. For example, it would not be difficult to imagine the use of edge followers and line fitters to further refine the results obtained by partitioning. Again, however, the real problems in the use of such methods is the determination of when they are applicable. At present, this determination must still be made by human judgment.

D. SYMBOLIC MANIPULATION OF POSITIONS AND ORIENTATIONS

A number of lesser modifications and improvements were made while the emphasis of the modeling was still on the compressor. The principal accomplishment in this area was an improved way of dealing with descriptors of position and orientation.

Positions and orientations of parts and assemblies are represented within our computer programs by homogeneous transform matrices. (See [16].) Within the LISP implementation of our modeling system, as within most systems that do geometric modeling, routines exist to generate the primitive translation matrix (with an arbitrary x, y, and z), and a primitive rotation matrix about any of the three principal axes. To obtain a compound motion, primitive matrices are numerically multiplied, and the result matrix is stored to represent the compound motion.

If one or more of the parameters of a compound motion are unknown, however, numeric matrix multiplication cannot be performed. For example, the height of the pressure gauge can be obtained by multiplying together the matrices representing the relative locations of the gauge, pressure switch assembly, tank, compressor, table, and room. But if the location of the table relative to the room is not known, we would still like to be able to obtain the information that the pressure gauge is 17.4 inches above the tabletop.

To provide this sort of capability, we designed a multiplier of symbolic homogeneous transforms. The multiplier operates on constructs we call "position and orientation descriptors", each of which consists of a list of primitive elements. Each primitive element is either a translation of the form (TRANSLATE <x> <y> <z>) or a rotation of the form (ROTATE <axis> <theta>). <x>, <y>, <z>, and <theta> may be either numbers or symbolic quantities that presumably (but not necessarily) evaluate to numbers. <axis> should be of the form +X, -Y, and so forth.

The symbolic multiplier knows about such rules as the following:

(TRANSLATE A B C)(TRANSLATE D E F) = (TRANSLATE A+D B+E C+F)

(ROTATE <any axis> A) (ROTATE <same axis> B)
= (ROTATE <same axis> A+B)

(ROTATE -<any axis> A) = (ROTATE +<same axis> -A)

(ROTATE +X 90) (TRANSLATE A B C)
= (TRANSLATE A -C B) (ROTATE +X 90)

There are also rules for adding and subtracting numeric and symbolic quantities and for eliminating zero sums and null transformations. The symbolic multiplier systematically applies the rules to any position and orientation descriptor to simplify it wherever possible.

It is usually known that the position of the table is 6 inches to the right of and 30 inches behind the origin of coordinates (a reference mark on the floor). Obtaining the position of the pressure gauge requires multiplying the following position and orientation descriptors:

(TRANSLATE 6 30 0)	Locates table with respect to room
(TRANSLATE 0 0 27.6)	Locates board forming tabletop
(TRANSLATE 0 0 .7)	Locates top surface of board
(TRANSLATE 0 0 .95)	Locates turntable
(TRANSLATE 0 0 .7)	Locates top surface of turntable and compressor base
(TRANSLATE 0 0 1.5)	Locates top of base and bottom of compressor itself
(TRANSLATE 0 0 1.6)	Locates bottom of tank
(TRANSLATE 0 0 6.2)	Locates center of tank
(ROTATE +Y 90)	Provides a coordinate frame in which to describe a cylinder
(TRANSLATE 0 -6.2 0)	Locates side of cylinder
(ROTATE -Y 90)	Makes side of cylinder top of tank
(TRANSLATE -11.5 0 0)	Locates a place to attach pressure switch assembly
(TRANSLATE 0 -.5 2.4)	Locates the gauge within the assembly
(ROTATE +X 90)	Rotates the gauge forward

(TRANSLATE 0 0 .5)   Locates the center of the gauge

Symbolically multiplying the relative position descriptors yields (TRANSLATE -5.5 29.0 47.85) (ROTATE +X 90) for the position and orientation of the pressure gauge. If we suppose the position of the table in the room to be unknown, we can represent that by letting (TRANSLATE TABLEX TABLEY 0) represent the position of the table instead of (TRANSLATE 6 30 0) as above. Now the symbolic multiplication will give the result

```
(TRANSLATE (PLUS -11.5 TABLEX)
           (PLUS -1.0 TABLEY)
           47.85 )
(ROTATE +X 90)
```

for the position of the pressure gauge.

III WHAT WE HAVE LEARNED

A number of interesting and useful things have come out of the exercise with polyhedral models. Even though many limitations and drawbacks were discovered, some very useful and novel features were demonstrated. We are now in the process of designing a brand new representation system. We hope to incorporate the useful features of the old in the new, while correcting some of the deficiencies of the old.

One of the major goals of this project is to produce a system whereby objects can be described to the system using by natural, familiar, and intuitive concepts. For this, attachment points are a particularly useful feature. They provide a way to specify the relative position of two parts such that the surfaces of the parts are adjacent. Relative displacements can be specified relative to the attachment points, too, so flexibility and generality are not lost.

Visualizing rotations is a difficult task. In general, rotations about the vertical are easier to describe than those about horizontal axes. The difficulty increases when rotations about more than one axis and rotations of angles other than 90 degrees are involved. The problems generally relate to confusion and ambiguities with coordinate systems. At various times it may be useful to describe motions or directions in coordinate systems attached to individual parts, to assemblies, or to a gravitational frame of reference. A useful geometric modeling system should provide the capability of using any of these, as well as some explicit and reasonable defaults.

Modifying the description of an object already in the system was awkward. The main obstacle was that the symbolic and the polyhedral data structures were written in different computer languages and ran in different address spaces of the TENEX operating system. This made for duplication of information and difficulty in maintaining correspondence between the models. Whenever a change was made to the symbolic model, it was necessary to erase the entire polyhedral data structure, then regenerate it according to the new symbolic model.

In general, describing objects and knowledge about objects to a computer is a difficult task. Anything that can be done to make the task easier will probably be worthwhile in the long run. However, because the primary goal of this project is research results rather than a working system for interactive parts specification, the choice was made not to implement yet another interactive design and display system.

With regard to the use of the models for range-finder-based vision, we discovered that the very existence of the models is a powerful aid to making sense of a scene. For a top-down type of strategy, where a specific objective is sought, the models suggest specific tests to make with a limited number of range measurements. The tests are based on "distinguishing features", that is, on finding a test that will distinguish one hypothesis from an alternative one.

We have not been able so far to automatically generate any strategies based on distinguishing features. The polyhedral models do not lend themselves very well to the sort of analysis needed for that. The process of deriving edges and vertices from the symbolic and semantic models is well defined; reversing the process to derive useful information from the polyhedral representation is next to impossible.

The polyhedral models do not treat curved objects well. To do a better job with this we should add the capability of representing parts by a principal axis and a cross section described on that axis. These primitives we call snakes. They have been described by Agin and Binford [6].

When it came time to add snakes to the representation, we found that the polyhedral representation would have to be modified drastically to handle the snakes. The resulting system would probably be inefficient and clumsy. Because of the other deficiencies we found in the system, we decided that the best course would to start from scratch to design a better representation, based on both polyhedra and snakes, with closer ties between the symbolic and the geometrical data bases.

REFERENCES

1. P. H. Winston, "Learning Structural Descriptions from Examples," Project MAC Report MAC-TR-76, Massachusetts Institute of Technology, Cambridge, Massachusetts (September 1970).

2. I. C. Braid, Designing with Volumes, (Cantab Press, Cambridge, England 1973).

3. B. G. Baumgart, "GEOMED--A Geometric Editor," Stanford Artificial Intelligence Laboratory Memo AIM-232, Stanford University, Stanford, California (May 1974).

4. "An Introduction to PADL," Project Automation Project, University of Rochester, Rochester, New York (December 1974).

5. Y. Shirai and M. Suwa, "Recognition of Polyhedrons with a Range Finder," Proceedings Second International Joint Conference on Artificial Intelligence, Edinburgh, Scotland (1971).

6. G. J. Agin and T. O. Binford, "Computer Description of Curved Objects," IEEE Transactions on Computers, Vol. 25, No. 4 (April 1976).

7. R. K. Nevatia and T. O. Binford, "Structured Descriptions of Complex Objects," Proceedings Third International Joint Conference on Artificial Intelligence, Stanford, California (1973).

8. P. M. Will and K. S. Pennington, "Grid Coding, A Preprocessing Technique for Robot and Machine Vision," Proceedings Second International Joint Conference on Artificial Intelligence, Edinburgh, Scotland (1971).

9. M. D. Levine, D. A. O'Handley, and G. M. Yagi, "Computer Determination of Depth Maps," Computer Graphics and Image Processing, Vol. 2, pp. 131-150 (October 1973).

10. B. G. Baumgart, "Image Contouring and Comparing," Stanford Artificial Intelligence Laboratory Memo AIM-199, Stanford University, Stanford, California (October 1973).

11. H. Baker, "Building Models of Three-Dimensional Objects," Masters' Thesis, Department of Machine Intelligence, University of Edinburgh, Edinburgh, Scotland (1975).

12. B. K. P. Horn, "Shape from Shading: A Method for Obtaining the Shape of a Smooth Opaque Object from One View," Project MAC Report MAC-TR-79, Massachusetts Institute of Technology, Cambridge, Massachusetts (1970).

13. N. J. Nilsson et al., "Artificial Intelligence--Research and Applications," Progress Report, Contract DAHC04-72-C-0008, Stanford Research Institute, Menlo Park, California (May 1975).

14. N. J. Nilsson et al., "Artificial Intelligence--Research and Applications," Progress Report, Contract DAHC04-72-C-0008, Stanford Research Institute, Menlo Park, California (May 1974).

15. J. M. Tenenbaum and H. G. Barrow, "Experiments in Interpretation-Guided Segmentation", Artificial Intelligence Center Technical Note 123, Stanford Research Institute, Menlo Park, California (March 1976).

16. R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis, (John Wiley and Sons, New York, New York, 1973).