VISION ANALYSIS USING COMPUTER AIDED GEOMETRIC MODELS[1]

B. Bhanu, T.C. Henderson and S.W. Thomas

Department of Computer Science
The University of Utah
Salt Lake City, Utah 84112

January 27, 1985

Abstract

This paper presents the initial results of our work in using the Computer Aided Geometric Design (CAGD) representations and models as a basis for the visual recognition of 3-D objects for robotic applications. We describe some techniques and algorithms which allow the generation of computer representations and geometric models of complicated realizable 3-D objects in a *systematic manner*. These representations and models are obtained using (a) available CAGD techniques, and (b) data acquired from various range finding techniques. As compared to previous work in machine vision, multiple hierarchical representations of an object obtained from geometric models can be used for finding orientation and position information.

---

[1]This work was supported in part by the System Development Foundation and NSF Grants ECS-8307483 and MCS-82-21750

1. Introduction

In the past a number of different sensors and techniques have been used for the representation and modeling of 3-D objects and to determine the orientation and position of objects in 3-space [7, 19, 24]. However, there has been an absence of a *systematic approach* for building such models, for a large class of objects used in industrial environments, which can be used for matching with 2-D images or arbitrary 3-D views of objects. The models and matching strategies have been limited in scope because of the lack of generalization to other objects or even objects of similar types with minor variations in their descriptions. The emergence of Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) technology as a driving force in manufacturing engineering has provided new opportunities for the use of geometric models of real world 3-D objects and to build systems to carry out the tasks of visual recognition and identification [17]. In this paper we are concerned with the use of Computer Aided Geometric Design (CAGD) representations and models as a basis for the visual recognition of objects for robotic applications. Current CAGD systems offer an interactive design environment by providing facilities to create images of the designed parts, perform analysis functions on them (e.g., finite element analysis), and produce numerically-controlled machining information for manufacturing. Figure 1 shows our approach to CAGD based vision analysis.

```
Interactive Design  ——   CAGD System        ——   NC Manufacturing
                         Representation
   Analysis         ——   and Models         ——     Graphics
                              |
                         Computer Vision
                         Representation
                           and Models
                              |
                         Vision Analysis
```

Figure 1. CAGD-Based Vision Analysis System

In the following we describe the work that has been done using these representations in the area of computer vision and CAGD. Here we present our representation scheme which allows multiple hierarchical representations of an object. This is followed by a discussion on modeling and matching. Finally we describe a number of methods which can be used to generate data for building a vision model and present our initial results.

2. Representation in Vision

Geometric modeling is one of the key components of a domain-independent model-based 3-D industrial machine vision system. Here our interest is in the representation, modeling and recognition of rigid, opaque 3-D solid objects. Three general classes for the representation of 3-D solid objects are (a) surface or boundary, (b) volume, and (c) sweep [2, 31]. In the boundary representation schemes, a 3-D solid object is represented by segmenting its boundary into a finite number of bounded "faces" and describing the structural relationships between the faces. Another approach to surface representation is to express the surfaces as functions on the "Gaussian Sphere" [22, 35]. Volumetric representations include spatial occupancy, cell decomposition and constructive solid geometry [32, 33]. Sweep representations consist of translational sweep, rotational sweep, 3-D sweep and general sweep.

Since a direct model of a 3-D object in terms of its volume (e.g., as a 3-D array) may easily exhaust the memory capacity of a system, representation by oct-trees has been considered [23]. These may make space array operations more economical in terms of memory space. A simple approach to analyzing 3-D objects is to model them as polyhedra. This requires a description of the objects in terms of vertices, edges and faces. Baumgart [5] developed a 3-D geometric modeling system ("Geomed") for application to computer vision. He used a face-based representation for planar polyhedral objects, called the "winged-edge" representation. The Euler primitives are used for polyhedron construction and shape operators include union, intersection and difference. Geomed provides many capabilities; for example, arbitrary polyhedra may be constructed, altered or viewed in perspective with hidden lines eliminated. Bolles et al. [9] have used a CAD model that contains a standard volume-surface, edge-vertex description as well as pointers linking topologically connected features. Their preliminary model uses a pointer structure similar to Baumgart's "winged-edge" representation. Wesley et al. [38] have used polyhedral models for automated mechanical assembly in their geometrical modeling system GDP. Their Automated Parts Assembly System (AUTOPASS) [25] language has never been implemented. Before automated assembly can be successful, it is essential to have robust representations, models, and general purpose techniques for determining the orientation and position of 3-D objects for a large class of industrial parts.

For curved and more complex objects, other representations and models have been used, such as generalized cylinders (or cones). Generalized cylinders or cones are a quite popular representation in computer vision [1, 11, 28]. However, there are some problems with this representation. There are infinitely many possible generalized cones representing a single object. More constraints are needed to get a unique description. Although it is possible to represent arbitrary shapes with generalized cones by making them arbitrarily complex, their computation is difficult. They are also not well suited to descriptions of non-elongated objects and objects of arbitrarily deformed surfaces enclosing little volume. The generalized cone primitives used by Nevatia and Binford [28] and Brooks [11] are not sufficient to represent the automobile casting shown in Figure 2. Note that this casting does not contain any major horizontal or vertical surface and is quite complicated in shape.

Figure 2. Automobile Part

Although sweep representations such as generalized cylinders, and volume representations, such as constructive solid geometry, imply surface description, they fail to describe the junction or surface peculiarities. In the recognition of 3-D objects from partial views, we detect surfaces first, and only after seeing several different views of the object do we have enough data to obtain volume properties. For objects constructed from thin sheet-like material, surfaces are natural candidates for representation. Further surfaces are seen first. As such they are important for computer vision. Hence the need for surface or boundary based representations.

York et al. [39] have used a structured collection of Coons surface patches for representing 3-D objects where boundary curves are approximated by cubic B-splines

Their design of Coons patches is cumbersome, since it requires that a simple surface patch be designed on paper before it may be entered into the data base. Brady [10] proposes a symbolic representation of visible surface based on "curvature patches." They are computed locally by determining the tangent vectors that indicate directions in which the surface changes. Example directions include the principal curvature directions and the directions in which the normal curvature is zero. Smooth changes in curvature patch descriptions are obtained to determine the larger scale structure of a surface. It is not clear that curvature patch surfaces are perceptually "fairer" than surfaces developed in CAD.

We have used a region growing technique [7, 19] to obtain a planar surface representation of 3-D objects. Figure 3 shows the results of planar surface approximation in one view of the object shown in Figure 2. Various faces are shown in different colors. There are 22 faces in the view and they are labeled in the order they are found using the planar approximation algorithm [7]. Here we have used only geometrical information, i.e., position of a point in 3-D space. We have not used any topological information. However, such information is of great use; it can be derived or made available by the CAGD representation of 3-D objects and can be effectively used. Other techniques which have been explored by us include the split and merge technique, surface normals and clustering techniques for the segmentation of range data [6]. Faugeras [18] has also used a region growing algorithm working on a 3-D graph to approximate object surfaces. For a critique and details of various representation and modeling techniques for 3-D objects in computer vision, see [7, 19]

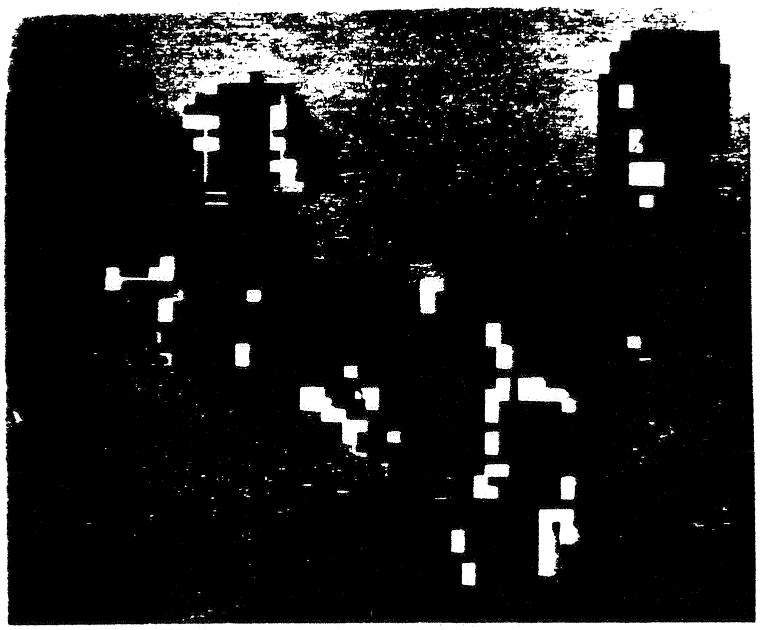


Figure 3. Planar Surface Approximation

3. Representation in CAGD

Constructive solid geometry (CSG) and boundary representations are the best understood and currently most important representation schemes in computer aided design. Present day 3-D wireframe models used in CAD and model-based vision have many deficiencies including ambiguity - it is easy to build a wireframe model that can be surfaced in several ways [32]. In CSG, the basic idea is that complicated solids can be represented as various ordered "additions" and "subtractions" of simpler solids by means of modified versions of Boolean set operators-union, difference and intersection [31]. For inherent boundary representations a number of different approaches are used. These include Coons patches, bicubic surface patches, Bezier methods and B-splines [4].

Current Geometrical Modeling Systems (GMS) use a limited class of primitives such as rectilinear blocks and conic surfaces (cylinders, cones and spheres). Although these are sufficient to cover a large number of conventional unsculptured parts, a GMS which includes sculptured solids is highly desirable. Also since the sculptured design is surface oriented, it is easier to incorporate it in a boundary based system. In general, boundary modelers tend to support stepwise construction of the models more easily than CSG modelers but require greater data storage. CSG modelers are inadequate for modeling sculptured parts: they have no capability at all for constructing and using sculptured surfaces as part of the boundary of the solid model. Some advantages of boundary representation are: there are many known surface models available from which to choose [4]; the mathematics of surface representation is well developed and complex shapes can often be represented with a single primitive [14, 36]; and it results in an intuitive model. A minor disadvantage is that it may be difficult to ensure the validity of a boundary representation of a set. On the other hand, CSG representations are not unique in general, since a solid may be constructed in many ways; the final result may not be easily visualized by looking at the primitives. However, the CSG representation is concise, validity is guaranteed and such a representation can be easily converted to a boundary representation. The comparison of CSG and boundary representation methods can be found in [32, 33].

Recently there have been attempts to use a set of manipulative operations for boundary models for solid objects to construct a solid modeling system [26, 36]. These are designed for CAD/CAM environments, rather than for computer vision applications. In [26], a set of Euler operators is used on the topology of a boundary model, that is on the relative arrangement of its faces, edges and vertices. The operations allow the system to perform arbitrary modifications necessary for boundary representation models, the faces of which are planar polygons.

Until recently it was not possible to carry out Boolean operations on sculptured surfaces. Recent work by Thomas [36] attempts to combine the best attributes of CSG and surface-based representation systems by using subdivision techniques developed by Cohen et al. [16]. He uses a uniform boundary representation. The "primitives" are solids bounded by B-spline surfaces. As compared to the other work in solid modeling, his method does not require that the objects being combined have closed boundaries; they must only satisfy a weak completion criterion. Thus this method results in a powerful shape description system which allows the combination of primitives using set operations into arbitrarily-complex objects bounded by curved surfaces and the production of a model which represents such objects. Adjacency information about surface points and the intersection curve between two surfaces as a polyline can be obtained. Although he has used B-spline surfaces, his techniques are applicable to any surface representation scheme [14]. All this work has been incorporated in the Alpha_1 system [15]. (More details about Alpha_1 are presented below.) Thus, the advantages of both CSG and sculptured surface representation can be obtained in the shape representation of objects and the combination of objects via set operations. As a result of these significant advances in CAGD, we decided to use the Alpha_1 system for exploring the computer vision application.

Alpha_1 is an experimental CAGD based solid modeler system incorporating sculptured surfaces [15]. It allows in a single system both high quality computer graphics and freeform surface representation and design. It uses a rational polynomial spline representation of arbitrary degree to represent the basic shapes of the models. The rational spline includes all spline polynomial representations for which the **denominator** is trivial. Nontrivial denominators lead to all conic curves. Alpha_1 uses the Oslo algorithm [16] for computing discrete B-splines. Subdivision, effected by the Oslo algorithm, supports various capabilities including the computation associated with Boolean operations, such as the intersection of two arbitrary surfaces [36]. B-splines are an ideal design tool, they are simple yet powerful; many common shapes can be represented exactly using rational B-splines. For example, all of the common primitive shapes used in CSG systems fall into this category. Other advantages include good computational and representational properties of the spline approximation: the variation diminishing property, the convex hull property and the local interpolation property. There are techniques for matching a spline-represented boundary curve against raw data. Although the final result may be an approximation, it can be computed to any desired precision (which permits nonuniform sampling). At present, tolerancing information is not included in the object specification in Alpha_1 system. It is planned to be incorporated in the future. Once it is available, we can make our models in terms of classes of objects (rather than a single object) which are functionally equivalent and interchangeable in assembly operations. Figure 4 shows the relation between the CAGD model and the generalized vision model.

```
   CAGD Model                      Generalized Vision Model

       o
      / \                                  o
   s1    o            ---\               / | \            ---\   Analysis
        / \           ---/             m1 m2 m3           ---/   Routines
      s2   s3
```

Figure 4. Relation of CAGD Model to Vision Analysis

Given the CAGD model (perhaps by combining several modeling paradigms), a corresponding set of vision models (with some control structure) is generated. Once these models are available they provide the basis for standard 2-D and 3-D scene analysis. An early example of such an interactive system is the ACRONYM system [11, 12] designed for applications in computer vision and manipulation. The world is described to ACRONYM as volume elements and their spatial relationships and as classes of objects and their subclass relationships. It uses a hybrid CSG and general sweep scheme for the representation of rigid solids. The representations are CSG-like trees whose leaves are generalized cylinders. Like PADL (a geometric modeling system [13]) it allows variation in size, limited variation in structure and variation in structural relationships of the modeled objects. However, in ACRONYM, it may be difficult to design algorithms for computing properties of objects.

3.1. Modeling

One of the limitations of the earlier work in 3-D scene analysis has been the restriction to a single range image of the object in building a 3-D model [1, 28]. This results in a model of only part of the object, not the complete object. Nevatia and Binford [28] used tree structured generalized cone models of objects detected using a laser range finder. In our work [7], the 3-D model was automatically built by combining object points (obtained by using a laser ranging system) from a sequence of range data images corresponding to various views of the object, applying the necessary transformations and then approximating the surface by polygons. Another approach for automatic generation of a 3-D model which combines the information from several views is the work reported by Potmesil [30]. He uses a bicubic parametric patch as the basic surface element. A heuristic search algorithm is used to register partially overlapping surface segments into a common space and then the overlapping sections are merged. The match and merge process is iteratively repeated for all 3-D surface segments until a complete model of the object in a single 3-D space is created. It is to be noted that, as compared to these 3-D model building approaches in vision, CAD systems for industrial parts normally build models which are viewpoint independent and provide a 3-D description of the volume occupied by the part. Thus they can be used in vision and manipulation applications. Bolles et al. [9] have used a preliminary CAD model based on Baumgart's "winged-edge" representation.

3.1.1. Matching

In 3-D scene analysis, we have a model for 3-D objects and a method for matching unknown objects with the model. Matching of 3-D occluded objects involves the handling of multiple overlapping parts as it occurs in the bin-picking problem. Bolles et al. [9, 21] report their preliminary efforts to develop general techniques for recognizing and locating partially visible objects for solving the bin picking problem with 3-D uncertainties in their location. Their matching technique uses local features to achieve hypothesis generation and verification. Formation of a recognition strategy and ranking of focus features are done interactively. A basic problem is how to select the few features that are sufficient to identify the part unambiguously.

Oshima and Shirai [29] have used range information for the recognition of blocks and simple machine parts by matching the features and relation based description of the scene with the stored model. The Hough transform technique of Ballard and Sabbah [3] to detect the presence of a 3-D object is based on the fact that all the planar regions be adjacent to each other in the object representation. However, in practice it may not always be feasible [7, 29]. There are other problems which arise with the use of Hough transform in 3-D [20]. To obtain good accuracy of a rotation transformation, one needs to quantize the accumulator finely, thus yielding large memory requirements and there is in general an infinity of transformations which map a plane on to a plane or a quadratic on to a quadratic, thus making the updating of the accumulator difficult. Faugeras [18] used an algorithm based on hypothesis prediction and verification to recognize and position 3-D objects. He used the rigidity constraint to reduce the combinatorial explosion problem. It has some serious theoretical problems for estimating the rotation matrix for quadric primitives. The matching work of York et al. [39] using curve and patch shape features is preliminary. Several other 3-D models and matching techniques used in computer vision are discussed in [7].

4. 3-D Vision Model Building and Recognition

As we have discussed there is at present a lack of convenient 3-D solid modelers which are capable of generating object descriptions suited to the problem of 3-D object recognition and localization. Here our objective is to provide and use such a description. In our system, parts are designed and modeled using the Alpha_1 CAGD system. It models the geometry of solid objects by representing their boundaries as discrete B-splines. It allows the combination of primitive objects into more complex objects using set operations. It supports several modeling paradigms, including direct manipulation of the B-spline surface, creation and combination of primitive shapes, and high-level shape operators such as bend, twist, and warp. The single underlying mathematical formulation of Alpha_1 simplifies implementation, but it is sufficiently powerful to represent a very broad class of shapes. It is able to create images of the designed objects, to perform certain analysis functions on them, and to produce numerically-controlled machining information for manufacturing [34].

By using the Alpha_1 system, several methods are available to generate data from which a vision model can be built. (a) *Ray tracing* is a common technique to produce realistic images of a solid model. Light rays are traced from the eye to the object and then to the light source to determine the illumination at each point in the scene. Evenly spaced rays can be traced to the model, and the intersection points between the rays and the model gives the range information [37]. This simple technique can be used to *simulate a laser scanner* and a set of 3-D points on the surface of the object can be obtained. (b) A procedure which is simple to program and produces an irregular set of points (nonuniform) is to subdivide the B-spline surface which comprises the boundary of the object [16]. The bounding surface can be subdivided until all the resulting pieces are smaller than some resolution. Then the center points of all the small surface pieces can be obtained as the surface representation of the object by 3-D points. (c) Noting that (a) and (b) discard information which is available in the original model, during the subdivision process in (b) while generating surface points, a spatial adjacency graph may also be generated. It could be very useful for surface approximation, since it contains topological information. (d) Methods as described in (a) to (c) produce the surface description in terms of the points on the surface of the object. These points have to be processed further to get the higher level surface representation. It is possible to obtain data in a much more intelligent fashion. As an example, during the subdivision stage, we can detect surface pieces which are planar (or quadric). Now the data available will be in terms of single polygons (or quadric or mixed types). Several of these polygons can be merged using the spatial adjacency graph to obtain a structured collection of large faces. Here the characteristic of the bounding surfaces (such as flat, cylindrical, etc.) retained in the model can also be used as an aid in the segmentation of actual range data and in the recognition tasks. We are currently investigating the above methods in detail in order to obtain intermediate level descriptions for the vision model.

The higher level multiple hierarchical representation model- An efficient representation depends upon the type of objects and the intended application. Classes of objects based on shape can be grouped as elongated, polyhedral, curved surfaces and complex objects which incorporate the components of other shape types. For each of these types a particular representation may be more suitable than others. Three important features of 3-D model representation are [27] (a) that its coordinate system is object-centered (b) that it includes volume primitives, that may exploit the space occupied by an object and not just its visible surfaces, and (c) primitives of various sizes are included, arranged in a modular hierarchical manner. In our CAGD based vision system, we claim to have all three features. Different parts of an object are designed such that they can have their own coordinate system. Once we have designed the complete surface, these parts are combined using set operations and we have the volume bounded by the surface. Properties such as the center-of-mass of the volume can be computed and all the points on the surface of the object can be referred to with respect to the body center of the object. Thus we can define canonically the coordinate scheme for an arbitrary shape and have the advantages of both a single coordinate frame that embraces the entire object and the distributed coordinate system for each articulated component or individual shape characteristic. By using set operations on volumetric primitives, we have a hierarchical representation of the object shape.

Thus CAGD systems may support several 3-D object modeling paradigms. Likewise, computer vision systems can be based on one of several 3-D representation schemes ([19]). Most CAGD systems and computer vision systems rely on only one representation, not several (e.g., ACRONYM is based on generalized cylinders). If an object is represented under a single modeling paradigm, we have a *homogeneous representation*. We believe, for the reasons mentioned above, that it is possible and necessary to support several modeling types in both the design phase and the analysis phase. For example the complicated casting shown in Figure 2 may comprise a combination of subcomponents, each modeled with a different paradigm. The paradigms are surface, volume and generalized cylinder or (cone). The subcomponents of the object using these paradigms are represented hierarchically. Such a representation is called a *heterogeneous* representation and it can be derived from the CAGD representation. Thus the model of the object consists of multiple hierarchical representations. From the considerations of matching we allow a certain amount of redundancy in the model in the form that, in addition to whatever natural representation is defined, we also have surface representation in terms of planar faces for all objects; this is used for indexing purposes. Of course, as pointed out above from the CAGD model we derive the information about the overall size of the objects, local features (such as the number of holes, their location and corners, etc.) and properties of the bounding surfaces. As compared to the previous work [9, 18] matching technique uses both the characteristics of the faces and the edge information in order to determine the position and orientation of the objects.

4.1. An Example

We will now consider the entire design and analysis process for part of the object shown in Figure 2. As can be seen, the piece has a highly irregular surface which poses significant representational problems. Figures 5 to 8 show the "left head" of the automobile part (Figure 2), during several stages in the design. The stage 1 (Figure 5) in the design is to lay out the outline of the face of the part. Points and construction lines are placed to define the centers and endpoints of the circular arcs, and straight lines segments are inserted between the arcs. In stage 2 (Figure 6) the side surface is produced by extruding the face outline. In stage 3 (Figure 7) the surface of the face is produced in several steps. It is divided into three sections corresponding to the large cylinder, the small cylinder and the connecting neck. The outline of each of these is broken into four segments and a Boolean sum surface computed. The division into parts is done so that the resulting surfaces have identical parametrization along their matching edges, and can therefore be rejoined into a single surface. The circular depression is produced by embedding its circular outline in the face surface and extruding it to the proper depth. It remains represented by a single B-spline. In stage 4 the bottom face is formed by reflecting the top surface through the center plane of the object. The separate surfaces are then assembled into a model of the "left head." Figure 8 shows a hidden line view of the left head. An example of adjacency graph obtained from CAGD modeling scheme as discussed above is shown in Figure 9.

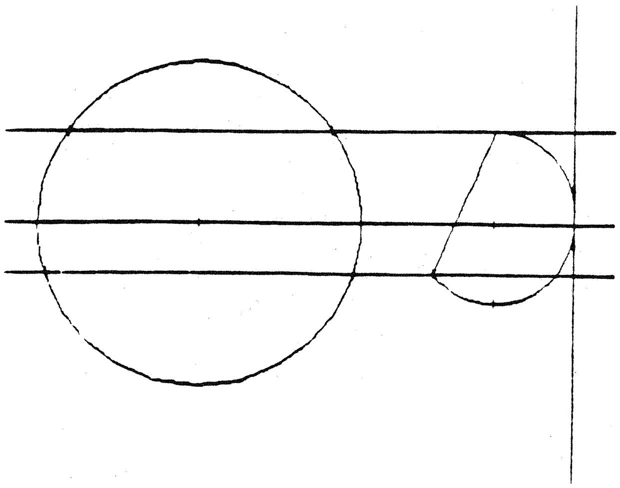


Figure 5. Stage 1 in the Design

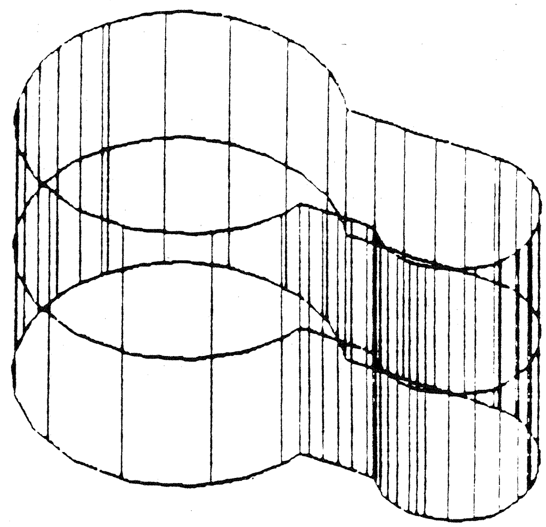


Figure 6. Stage 2 in the Design

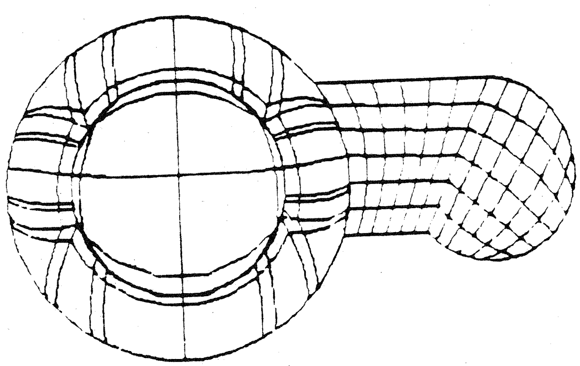


Figure 7. Stage 3 in the Design

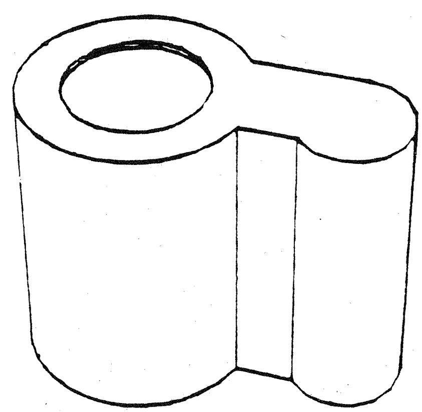


Figure 8. Stage 4 in the Design

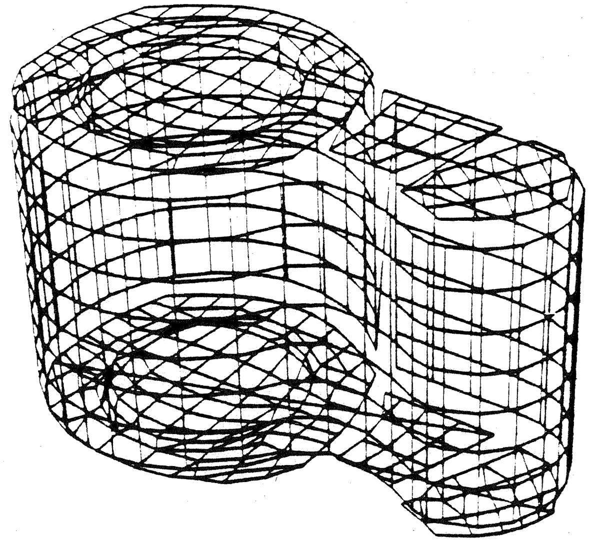
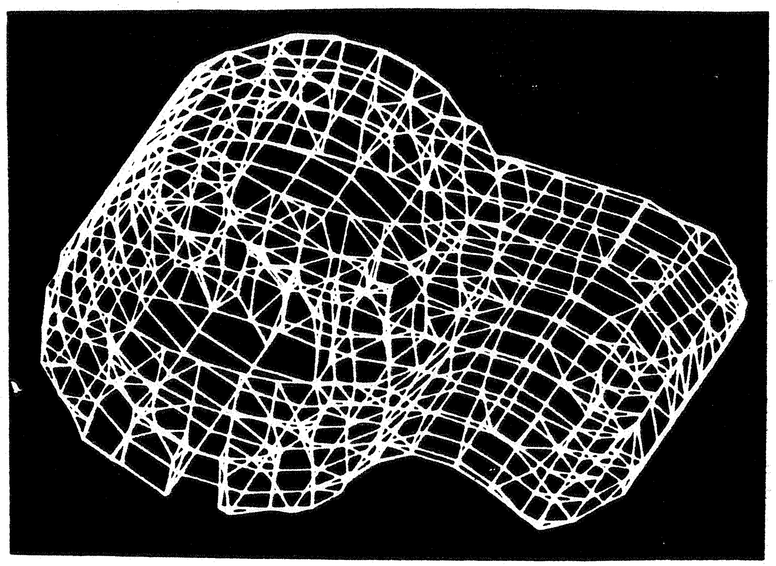
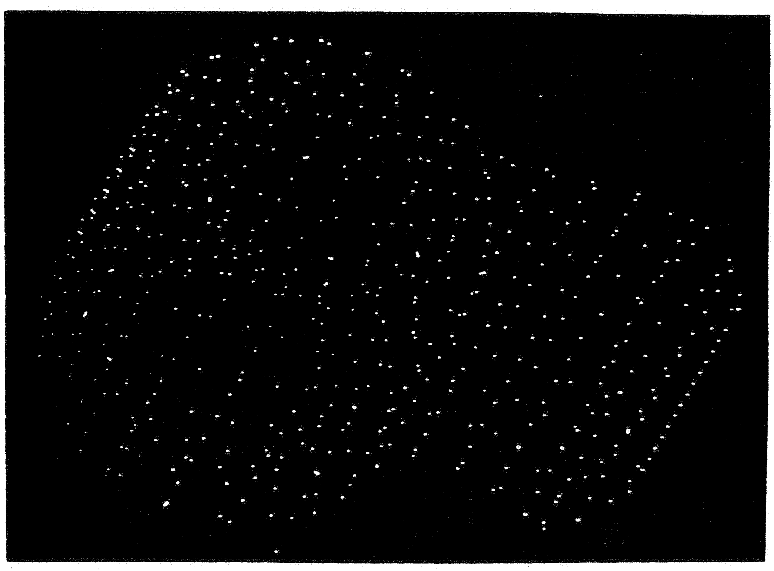


Figure 9. Adjacency Graph of Model

Once the part has been designed (or a subcomponent in this case), some method must be selected to generate the required data for building computer vision models. As discussed earlier, several methods can be used. Here we have chosen to use ray tracing to simulate the action of a laser range finder. Evenly spaced rays are traced to the model and the intersection points between the rays and the model are recorded. Such a set is shown in Figure 10(a). Once the set of points has been obtained, each point can be connected to its nearest neighbors by building the *spatial proximity graph* (see Figure 10b). From this graph it is easy to generate polygonal approximations to the data and then to perform matching based on those models [7, 19]. Given such a polyhedral model, scene analysis is performed by deriving descriptions of objects in the scene and matching those descriptions to the model. Final representation is a semantic network whose nodes can have multiple hierarchical representations and arcs describe geometric constraints and relationships. Recognition algorithms are based on such representations and make use of arc descriptions.



(a) Sampled Points



(b) Spatial Proximity Graph

Figure 10.

The results described here are produced from actual range data, and show how synthetic range data could be used to produce a model. See Bhanu [7, 8] for reports on both 2-D and 3-D shape matching including occlusion. The "face matching" of an arbitrary 3-D view with the 3-D model is done using a hierarchical relaxation technique which explicitly maximizes a criterion function based on the ambiguity and inconsistency of classification. The results of partial shape recognition are used to determine the orientation of the complicated automobile casting (Figure 2) in 3-D space, when viewed from any vantage point. Figures 11 and 12 show some results of the relaxation shape matching analysis. Each of these views has 24 planar faces and they are labeled in the order they are found by the planar approximation algorithm. The 3-D model had 85 faces. The transformation matrices of direction cosines obtained by using the matched faces of the unknown view and the 3-D model are given below.

0.88383	0.09058	0.46854
-0.20947	1.00000	-0.19653
-0.45183	0.01863	0.86441

Transformation Matrix for Fig. 11

0.89699	-0.02619	-0.58716
-0.14116	1.00000	-0.01034
0.49117	-0.02597	0.79737

Transformation Matrix for Fig. 12

The actual rotation (with respect to 3-D model of the object) in Figures 11 and 12 was 30° and -30° respectively along the y-axis

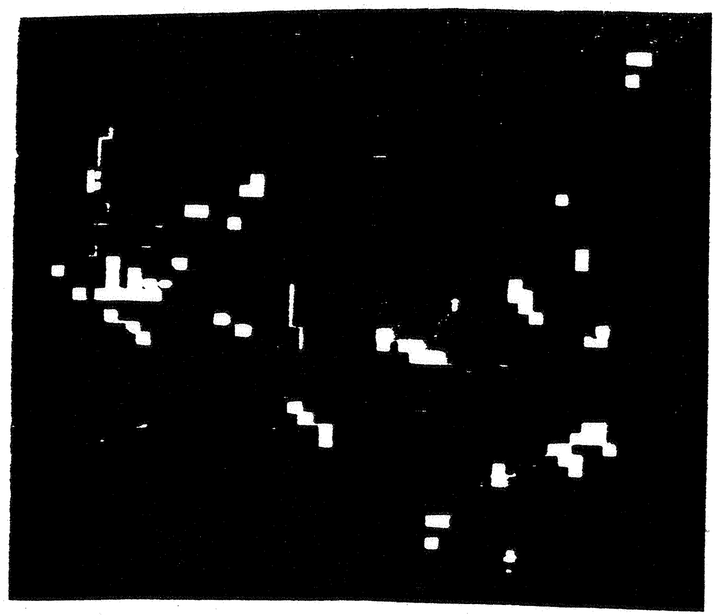


Figure 11. Faces Found in the 30° View

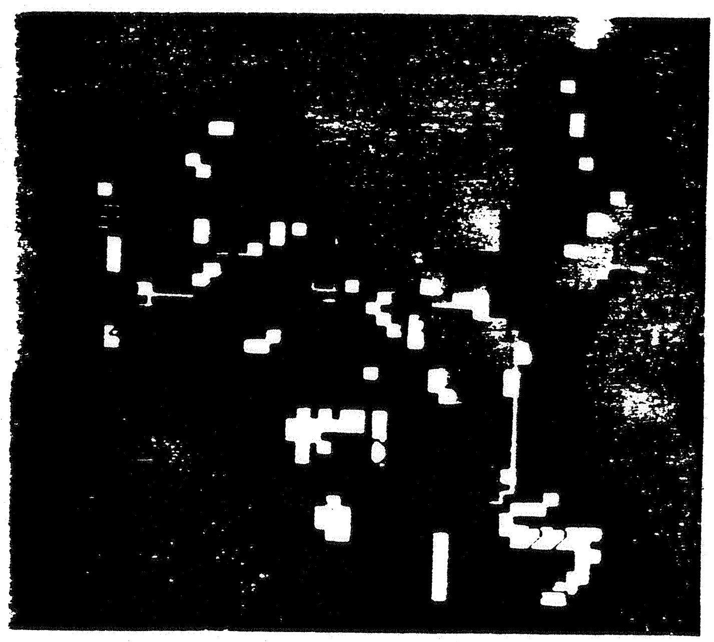


Figure 12 Faces Found in the 330° View

References

[1] G. Agin.
Representation and Description of Curved Objects.
Technical Report AIM-173, Computer Science Department, Stanford University, October, 1972.

[2] N. Badler and R. Bajcsy.
Three-Dimensional Representations for Computer Graphics and Computer Vision.
ACM Computer Graphics 12:153-160, August, 1978.

[3] D.H. Ballard and D. Sabbah.
On Shapes.
In *Proc. 7th IJCAI*, pages 607-612. Vancouver, August, 1981.

[4] R.E. Barnhill and R.F. Riesenfeld (eds.).
Computer Aided Geometric Design.
Academic Press, New York, 1974.

[5] B.G. Baumgart.
Geometric Modeling for Computer Vision.
Technical Report AIM-249, STAN-CS-74-463, Computer Science Department, Stanford University, October, 1974.

[6] B. Bhanu.
Shape Matching and Image Segmentation Using Stochastic Labeling.
Technical Report 1030, USC-IPI, August, 1981.

[7] B. Bhanu.
Representation and Shape Matching of 3-D Objects.
IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6(3):340-351, May, 1984.

[8] B. Bhanu and O.D. Faugeras.
Shape Matching of Two-Dimensional Objects.
IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6(2):137-155, March, 1984.

[9] R.C. Bolles, P. Horaud and M. J. Hannah.
3DPO: A Three-Dimensional Part Orientation System.
In *Proc. 8th IJCAI*, pages 1116-1120. Karlsruhe, August, 1983.

[10] M. Brady.
Representing Shape.
In *Proc. International Conference on Robotics*, pages 256-265. March, 1984.

[11] R.A. Brooks.
Symbolic Reasoning Among 3-D Models and 2-D Images.
Artificial Intelligence 17 285-348, 1981.

[12] R. Brooks, R. Greiner and T.O. Binford.
The ACRONYM Model Based Vision System.
In *Proc. 6th IJCAI*, pages 105-113. Tokyo, 1979.

[13] C.M. Brown.
PADL-2: A Technical Summary.
IEEE Computer Graphics & Applications :69-84, March, 1982.

[14] E.S. Cobb.
Design of Sculptured Surfaces Using the B-spline Representation.
PhD thesis, University of Utah, June, 1984.

[15] E. Cohen.
Some Mathematical Tools for a Modeler's Workbench.
IEEE Computer Graphics & Applications :63-66, October, 1983.

[16] E. Cohen, T. Lyche and R.F. Riesenfeld.
Discrete B-splines and Subdivision Techniques in Computer-Aided Geometric Design and Computer Graphics.
Computer Graphics and Image Processing 14(2):87-111, October, 1980.

[17] J. Encarnacao and E.G. Schlechtendahl.
Computer Aided Design.
Springer-Verlag, 1983.

[18] O.D. Faugeras.
New Steps Toward a Flexible 3-D Vision System for Robotics.
In *Proc. 7th International Conference on Pattern Recognition*, pages 796-805. Montreal, July-August, 1984.

[19] T.C. Henderson.
Efficient 3-D Object Representations for Industrial Vision Systems.
IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-5(6):609-618, November, 1983.

[20] T.C. Henderson and Wu So Fai.
The 3-D Hough Shape Transform.
Pattern Recognition Letters 2:235-238, June, 1984.

[21] P. Horaud, R.C. Bolles.
3DPO's Strategy for Matching Three-Dimensional Objects in Range Data.
In *International Conference on Robotics*, pages 78-85. March, 1984.

[22] B.K.P. Horn, R.J. Woodham and W.M. Silver.
Determining Shape and Reflectance Using Multiple Images.
Technical Report A.I. Memo 490, Mass. Inst. of Technology, August, 1978.

[23] C.L. Jackins and S.L. Tanimoto.
Oct-Trees and Their Use in Representing 3D Objects.
Computer Graphics and Image Processing 14(3):249-270, 1980.

[24] R.A. Jarvis.
A Perspective on Range Finding Techniques for Computer Vision.
IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-5(2):122-139, March, 1983.

[25] L.I. Lieberman and M.A. Wesley.
AUTOPASS: An Automatic Programming System for Computer Controlled Mechanical Assembly.
IBM Journal of Research and Development :321-333, July, 1977.

[26] M. Mantyla and R. Sulonen.
GWB: A Solid Modeler with Euler Operators.
IEEE Computer Graphics and Applications 2:17-31, September, 1982.

[27] D. Marr and H.K. Nishihara.
Representation and Recognition of the Spatial Organization of Three Dimensional Shapes .
Technical Report A.I. Memo 377, Mass. Inst. of Technology, August, 1976.

[28] R. Nevatia and T.O Binford.
Description and Recognition of Curved Objects.
Artificial Intelligence 8:77-98, 1977.

[29] M. Oshima and Y. Shirai.
Object Recognition Using 3-D Information.
In *Proc. 7th IJCAI*, pages 601-606. Vancouver, August, 1981.

[30] M. Potmesil.
Generating Models of Solid Objects by Matching 3-D Surface Segments.
In *Proc. 8th IJCAI*, pages 1089-1093. Karlsruhe, August, 1983.

[31] A.A.G. Requicha.
Representations for Rigid Solids: Theory, Methods, and Systems.
ACM Computing Surveys 12(4):437-464, December, 1980.

[32] A.A.G. Requicha and H.B. Voelcker.
Solid Modeling: A Historical Summary and Contemporary Assessment.
IEEE Computer Graphics & Applications :9-24, March, 1982.

[33] A.A.G. Requicha and H.B. Voelcker.
Solid Modeling: Current Status and Research Directions.
IEEE Computer Graphics & Applications :pp. 25-37, October, 1983.

[34] R.F. Riesenfeld.
Summary of the Concepts of the Alpha_1 CAGD System.
Detroit Engineer :8-11, April, 1982.

[35] D.A. Smith.
Using Enhanced Spherical Images for Object Representation.
Technical Report A.I. Memo 530, Mass. Inst. of Technology, May, 1979.

[36] S.W. Thomas.
Modelling Volumes Bounded by B-Spline Surfaces.
PhD thesis, University of Utah, June, 1984.

[37] L.V. Warren.
Geometric Hashing for Processing Complex Scenes.
Master's thesis, University of Utah, March, 1985.

[38] M.A. Wesley et al.
A Geometric Modeling System for Automated Mechnical Assembly.
IBM Journal of Research and Development 24(1):64-74, January, 1980.

[39] B.W. York, A.R. Hanson and E.M. Riseman.
3D Object Representation and Matching with B-Splines and Surface Patches.
In *Proc. 8th IJCAI*, pages 648-651. Karlsruhe, August, 1981.