Vision Analysis Using Computer Aided Geometric Models

B. Bhanu, T. Henderson and S. Thomas

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

Abstract

This paper presents the use of CAD/CAM models in the visual recognition of objects for robotic applications. As compared to previous work in machine vision, multiple hierarchical representations of an object obtained from geometric models are used for finding orientation and position information. Thus, design models are used to drive the vision analysis.

1. Introduction

In the past a number of techniques have been used to represent and model three-dimensional objects for computer vision applications. However, there has been an absence of a systematic approach for building such models for a large class of objects used in industrial environments. CAD/CAM technology offers new ways to model three-dimensional real world objects and to build systems for carrying out vision analysis. Figure 1 shows our approach to Computer Aided Geometric Design (CAGD) based vision analysis.

In this paper we first present a CAGD modeling system and then describe a number of methods which can be used to generate data for building a vision model. This is followed by a description of our ongoing work which allows multiple hierarchical representations of an object. From these, orientation and position of the objects can be recovered from vision data.

2. CAGD Modeling

The "left head" portion of the automobile part shown in Figure 2 was modeled using a Computer Aided Geometric Design (CAGD) system called Alpha_1 [7]. It models the geometry of solid objects by representing their boundaries as discrete B-splines. For a survey of CAGD see [1]. It allows the combination of primitive objects into more complex objects using set operations. It supports several

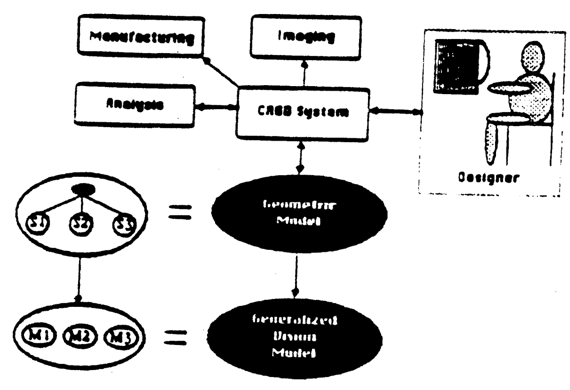


Figure 1. CAGD-Based Vision Analysis System

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, to support vision analysis and to produce numerically-controlled machining information for manufacturing [7, 3, 4, 8]. Here the use of Alpha_1 to perform various machine vision related functions is explored.

The first step in designing the "left head" portion of the part was to lay out the outline of the face of the part using fairly conventional two-dimensional design techniques. Points and construction lines were placed to define the centers and endpoints of the circular arcs, and straight line segments were inserted between the arcs. Figure 3 shows the defining curves together with some construction lines.

This work was supported in part NSF Grants ECS-8307483 and MCS-82-21750

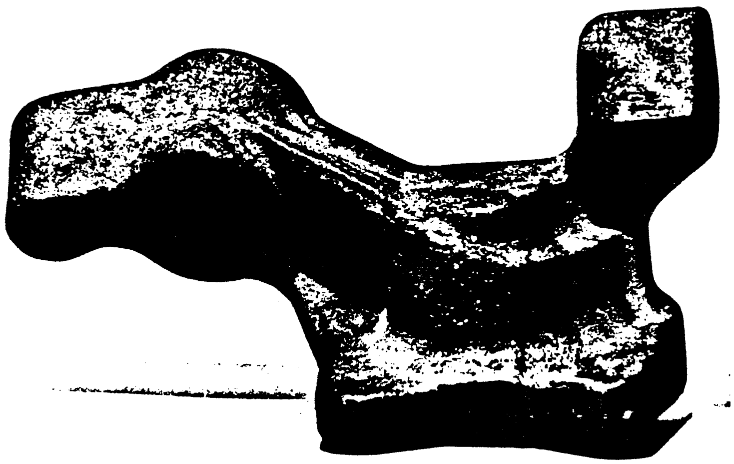


Figure 2. Automobile Part

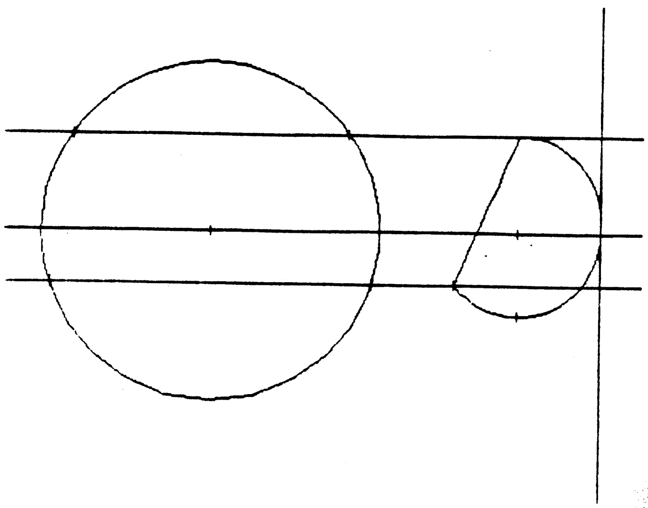


Figure 3. Stage 1 in the Design

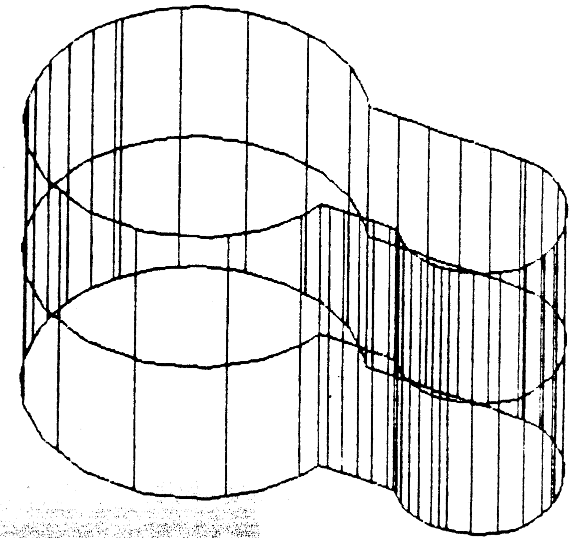


Figure 4. Stage 2 in the Design

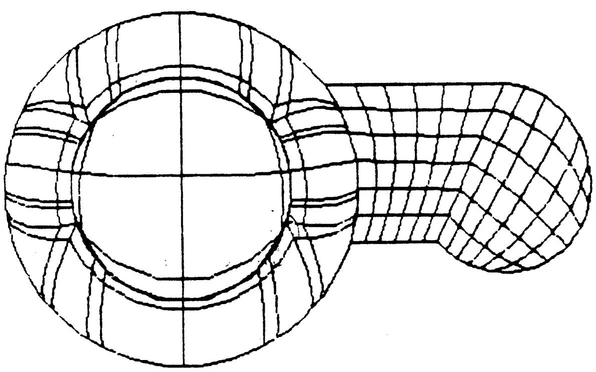


Figure 5. Stage 3 in the Design

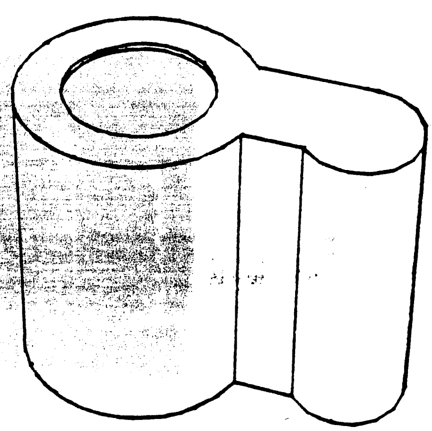


Figure 6. Stage 4 in the Design

The side surface was produced by extruding the face outline. That is, the face outline curve was copied at the correct distance and corresponding points on the two curves were joined. The result is shown in Figure 4.

The surface of the face was produced in several steps. It was divided into three sections corresponding to the large cylinder, the small cylinder and the connecting neck. The outline of each of these was broken into four segments and a boolean sum surface computed. The division into parts was done so that the resulting surfaces had identical parametrization along their matching edges, and could therefore be rejoined into a single surface. The circular depression was produced by embedding its circular outline in the face surface and extruding it to the proper depth. It remained represented by a single B-spline. The face is shown in Figure 5.

Finally, the bottom face was formed by reflecting the top surface through the center plane of the object. The separate surfaces were then assembled into a model of the head. A hidden line view of the head is shown in Figure 6.

3. Generating Data for a Vision Model

Several methods may be used to generate data from which a vision model can be built. One obvious method is to simulate the action of a laser scanner. A common technique for producing realistic pictures of a solid model is called ray tracing. Light rays are traced from the eye to the object and then to the light source to determine the illumination of each point in the scene. For a more realistic image, reflected and refracted rays can be followed through several bounces. For the purposes of computing a pseudo-laser-scanner image, however, such refinements are unnecessary. Evenly spaced rays can be traced to the model, and the intersection points between the rays and the model passed to the vision processor. This is a very primitive method and takes almost no advantage of the properties of the model, but is a simple initial interface between a modeler which already has a ray tracing procedure and a vision processor which expects laser scanner data.

Another similar, simple procedure produces a somewhat more irregular set of points. The B-spline surfaces which comprise the boundary of the object can be subdivided into smaller pieces. Subdivision of a surface produces two or more smaller pieces, which, taken together, exactly make up the original surface. For more details on subdividing B-splines, see [5]. The bounding surfaces 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 given to the vision processor in lieu of scanner data. The major disadvantage of this method is the uneven spacing of the resulting points. It is, however, almost trivial to program.

Again, the previous method discards information which was available in the original model. During the subdivision process, it is possible to keep track of which pieces of the surface are adjacent to each other. Then, when generating points, an adjacency graph may also be generated, instead of requiring the vision processor to go through an error prone process to deduce a spatial adjacency graph. An example of this sort of output is shown in Figure 7.

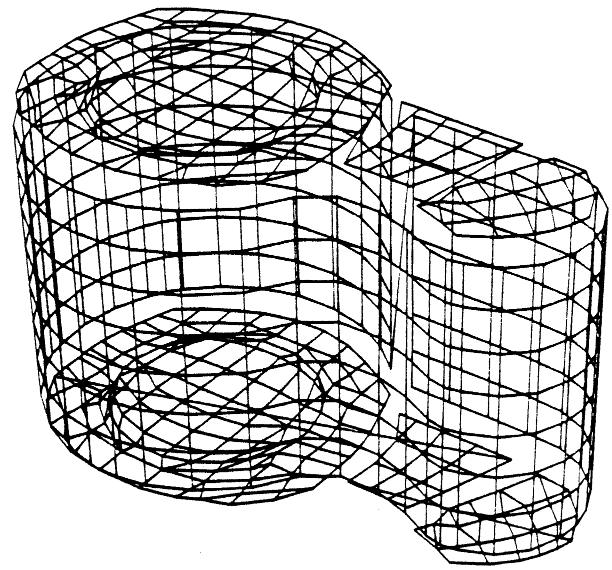
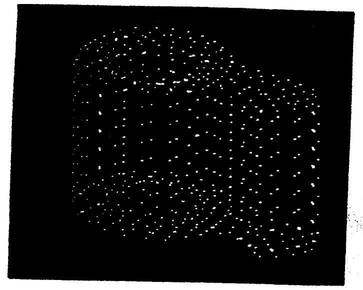
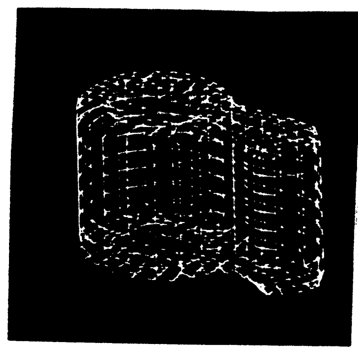


Figure 7. Adjacency Graph of Model

Alternatively, one can sample the model of the part to obtain a representative set of points from the surface of the object (see Figure 8a). From these, each point can be connected to its nearest neighbors by building the *spatial proximity graph* (see Figure 8b). From this graph is it easy to generate polygonal approximations to the data and then to perform matching based on those models. For a detailed discussion of the spatial proximity graph as a tool in computer vision, see [6].



a.



b.

Figure 8. (a) Sampled Points (b) Spatial Proximity Graph

All the preceding methods generate large quantities of data which the vision processor must process to find faces or other features. With a little knowledge of the type of features needed by the vision processor, it is possible to generate data for it in a much more intelligent fashion. For example, when feeding a vision processor that uses flat, polygonal faces, a simple check at each stage of subdivision can detect surface pieces which are already flat. These can be output at once as single polygons. The vision processor may still have to combine several of these polygons into a single face, but this is a much easier task than that of finding faces from a collection of points. Again, adjacency information retained from the subdivision process can aid the merge step. Given a suitable recognition function, any other type of face may be found and output during subdivision; thus giving the maximum amount of information to the vision processor.

In many cases, the bounding surfaces are designed to have a certain characteristic. For example, the sides of the "left head" of the automobile part here are either flat or cylindrical. This information is retained in the model, and could be passed straight to the vision processor.

4. Model Matching

From the CAGD models of a given object, computer vision models are derived. These models are used by the recognition algorithms to perform the analysis tasks. For an overview of representation techniques in computer vision, see [6]. For example, given that the part of the object is described as a quadratic surface, a generalized cylinder representation to drive the automatic analysis of the object will be more useful. On the other hand, if the part of the object is described by a collection of planar faces in the CAGD model, then this set is used to derive another set of planar faces as the vision model (for example, see [2]). All this results in 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.

Once an appropriate computer vision model is selected for the complete object, it will inform the analysis system about which sensor or sensors to use to complete the analysis. For example, generalized cylinder models may be more easily analyzed using some kind of structured light sensor (e.g., a plane of light displayed across the object). Other objects may be more readily recognized based on features (such as the number of corners, holes, etc.) recovered from a digital image.

References

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

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

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

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

[5] Cohen, E.; Lyche, T.; and Riesenfeld, R. F
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.
Also Tech. Report No. UUCS-79-117, October 1979.

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

[7] Riesenfeld, R. F.; Cohen, E.; Fish, R. D.; Thomas, S. W.; Cobb, E. S.; Barsky, B. A.; Schweitzer, D. L.; and Lane, J. M.
Using the Oslo Algorithm as a Basis for CAD/CAM Geometric Modelling.
In Aangeenbrug, R. T. (editor), *Proceedings of the Second Annual Conference of the NCGA.*
National Computer Graphics Association, Inc., Baltimore, 14-18 June, 1981.

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

C4.4: **Computer Graphics II**

Chairman: **R.M. Mueller**, Control Data Corporation

Vice Chairman: **Frank J. Cerra**, John Wiley

Keynote Speaker: **B. Bhanu, T. Henderson and S. Thomas**, The University of Utah

Title: "Vision Analysis Using Computer Aided Geometric Models."

Authors:

P.K. Chatterjee, D.E. Scheck and A. Narain, Ohio University, "An Intergraph Approach to 3-D Modeling and Mine Design."

***Shieh Shuren and He Yuanjun**, Shanghai University of Science and Technology, "A New Computer Method for Three-point Perspective."

***Koichi ITO, Hidesada KANDA, Takao MUKAI, and Ryohzo HOSOI**, Osaka University, "Interactive Color Image Processing System Used for Heat Transfer Analysis of Continuous-Type Electric Furnace."

Ma De-Chang, Li Xiao-He and Lu Lu, Beijung Institute of Aeronautics and Astronautics, "A New Solid Modeling System with Sculptured Surface Functions."

Jiang Yu, Chen Xiaoming and Wu Junheng, Beijung Institute of Aeronautics and Astronautics, "A Surface Modeling Package-A New Approach to Handle Triangular and Five Sided Patches."

David C. Goodrich and Craig Molander, Autometric, Inc., "Superposition of Computer Graphics into the APPS-IV Analytical Stereoplotter."

Shu-ren Shieh and Yuan-jun He, Tong Ji University, "A New Computer Method for Three Point Perspective."

CAD/CAM, Robotics and Automation

International Conference

February 13-15, 1985

Conference Director

K.M. Ragsdell, P.E.
Huber O. Croft Professor of Engineering
Chairman
Mechanical and Aerospace Engineering
University of Missouri-Columbia
Columbia, Missouri 65211

Papers Review Chairmen

Eric Sandgren
Mechanical and Aerospace Engineering
University of Missouri-Columbia
Columbia, Missouri 65211

Gary A. Gabriele
Mechanical Engineering
Rensselaer Polytechnic Institute
Troy, [illegible] 12[illegible]80

Proceedings

CAD/CAM, Robotics and Automation International Conference

February 13-15, 1985

Tucson, Arizona