The Synthesis of Visual Recognition Strategies[1]

Tom Henderson, Chuck Hansen, Rod Grupen and Bir Bhanu

UUCS-85-102

October 1, 1985

Abstract

A coherent automated manufacturing system needs to include CAD/CAM, computer vision, and object manipulation. Currently, most systems which support CAD/CAM do not provide for vision or manipulation and similarly, vision and manipulation systems incorporate no explicit relation to CAD/CAM models. CAD/CAM systems have emerged which allow the designer to conceive and model an object and automatically manufacture the object to the prescribed specifications. If recognition or manipulation is to be performed, existing vision systems rely on models generated in an *ad hoc* manner for the vision or recognition process. Although both Vision and CAD/CAM systems rely on models of the objects involved, different modeling schemes are used in each case. A more unified system will allow vision models to be generated from the CAD database. The model generation should be guided by the *class* of objects being constructed, the constraints of the vision algorithms used and the constraints imposed by the robotic workcell environment (fixtures, sensors, manipulators and effectors). We are implementing a framework in which objects are designed using an existing CAGD system and recognition strategies (logical sensor specifications) are automatically synthesized and used for visual recognition and manipulation.

---

[1]This work was supported in part by NSF Grants ECS-8307483, MCS-82-21750, DCR-8506393 and DMC-8502115. Chuck Hansen is an ARO Fellow.

1. Introduction

Computer vision has been an active research area for over 20 years. In the early days, emphasis was on low level processing such as intensity and signal processing to perform edge detection [1, 14]. Systems were constructed which only operated in very constrained environments or for very specific tasks [2, 13, 15]. It was quickly recognized that higher level concepts of *image understanding* were needed to successfully perform computer vision. More recently, models of objects and knowledge of the working environment have provided the basis for driving vision systems. This is known as model based vision. The pursuit of the fully automated assembly environment has fueled interest in model based computer vision and object manipulation.

The problem we are interested in solving is model based visual recognition and manipulation of objects in the automation environment. This involves building a 3-D model of the object, matching the sensed environment with the known world and locating objects. Not until the desired object is located and its orientation is known can a robot gripper or hand manipulate it.

Our goal is to develop a system which will work in the environment of the automated assembly process. This is not intended to provide a general model for the human visual process but rather a solution to the problem of visual recognition and manipulation in a well-known domain. The constraint we are imposing is one which limits the necessity of modeling the entire world. Rather, the known world to us is that of the automated environment in which this system is intended to operate.

Simply stated, our approach is to provide an integrated environment in which the CAGD model can be used to generate appropriate recognition and manipulation strategies. A major aspect of this work is the successful development of a prototype system combining design, vision analysis and manipulation. In this paper, we address the problem of the automatic synthesis of recognition code. This synthesis is derived in terms of the shape model and the available recognition schemes and is couched in terms of logical sensor systems.

2. Object Specific Recognition: Specialization

The system we are developing is based on the notion of *specialization*. This means that we will take advantage of any particular information that can be culled from the CAGD shape model. This knowledge will then be encapsulated in a special package which provides for the recognition of an object or part of an object. Thus, instead of using a general recognition technique on all parts to be recognized (i.e., a weak method), we will produce specially packaged code (i.e., logical sensors) for recognition. These can then be instantiated independently as needed, and can be controlled as logical sensors.

The approach consists of three phases:

1. **Design**. The object is designed using a CAD modeling system (the Alpha_1 CAGD system in our case).

2. **Derivation of Intrinsic Features**. The recognition strategies are based on matching intrinsic features of the object's shape with those of unknown shapes. A set of intrinsic features are derived from the CAGD system, and includes such features as: surface points, color, texture, surface normals, surface curvature, etc. See Henderson and Bhanu [10] for more details on our approach to the use of intrinsic features as the interface between CAD and computer vision systems.

3. **Synthesis of Object Recognition Strategies**. Given a set of intrinsic features for a specific object and knowledge of the representation chosen in the CAGD system (e.g., Constructive Solid Geometry, Generalized Cylinders, or Boundary Representation), plus knowledge of the available recognition techniques and feature detectors, the system will choose and hook together the appropriate recognition code. This can be done in a parameterized way at first, but eventually will require more expertise in using the knowledge that the system has available.

The system we describe here integrates the CAGD design system with the robotic workcell. The system contains knowledge of recognition strategies, shape representations, available sensors, and manipulation strategies. It uses this knowledge to guide the vision system and robot in the process of recognizing, locating and manipulating objects in the workcell environment.

The key issues in automatic generation of recognition strategies are:

1. producing the appropriate intrinsic features for a given object,

2. choosing an appropriate vision model for the object, and

3. choosing recognition algorithms and sensors.

The most difficult of these these three is selecting the appropriate vision model. Once the type of vision model is chosen, the appropriate intrinsic features are extracted from the CAGD model and used to generate the object model for the specific object. While this may not be straightforward, algorithms can be developed which can perform this transformation. The problem of selecting a vision model is constrained by several factors. One is the availability of recognition algorithms. If we consider the available algorithms to be stored in a library, the selection can be constrained by this library. The simplest case of selecting the correct representation occurs when the recognition library of known strategies is limited to one representation. This can be considered simple, even though the transformation from the CAGD model base may be nontrivial, since the selection of the shape representation is dictated by the singleton library of recognition schemes. Similarly, knowledge of the sensors available in the robotic workcell will further constrain the recognition procedure. These too can be thought of as being a library of available sensors.

The process is further complicated by the existence of CAGD models composed of multiple representations. For each complete CAGD model, there might possibly be several forms of representations contributing to the final result. If we think of the CAGD model as forming a tree of representations whose leaves are homogeneous models, we can match each of the shapes represented by these homogeneous models with some shape matching algorithm available to us in the library. Figure 1 demonstrates this idea. Consider a CAGD model to be made up of multiple structures, $\mathbf{S}_i$, each of which might possibly be in a different representational form. For each of the $\mathbf{S}_i$'s, the system must select an appropriate algorithm and sensor type to perform the matching in the workcell. This constrains the type vision model, $\mathbf{M}_i$, to be used.

Once the representation strategy is determined, the transformation from the CAGD representation to the recognition representation must be performed. Knowledge of this transformation is encoded along with knowledge of existing recognition algorithms. Thus,

```
CAGD Model                          Vision Model

     o                                    o
   /   \                                / | \
S1       o         ====>             M1 M2 M3         ====>  Analysis
       /   \                                                 Routines
     S2     S3
```

Figure 1. Relation of CAGD models to vision models

the method for the transformation can be explicit in the system. For example, if the recognition strategy uses generalized sweep, the model built from the CAGD model base would be in the form of sections of the generalized cylinder. Should planar or quadric patches be selected, the representation for recognition would be a graph structure of relations between the patches. If feature vectors are the chosen method for recognition, the features can be extracted directly from the CAGD model or the CAGD system might first produce an image of one view of the object then the features can be extracted by the same algorithm which processes the sensed data.

A straightforward method for generation of recognition strategies is parameterization. The user is required to *fill in the blanks* for the sensors and algorithms for the particular object, or class of objects, modeled. Obviously, a more automated system is desired for this task. Drawing from our experience with Logical Sensor Specification in the MKS system, our proposed method is to combine several algorithms and sensors to form a specialized *object finder* [3, 4, 5, 7, 8, 9, 10, 11, 12]. The methodology provides a means for abstracting the specification of a sensor from its implementation along with providing transparency of hardware and software above the implementation level. Alternatively, we are also investigating how to embed knowledge of the algorithms and sensors in the system and to provide a rule base for the decision process. This requires a complex expert system (see [6] for a description of a preliminary system). In either case, the system will eventually be composed of multiple sensors and recognition methods.

There are different recognition methods which have been successfully applied but have never been unified in a single system. This system will include multiple recognition algorithms, each of which might require a different vision model representation scheme.

Figure 2 represents the general schema we have in mind for the generalization of recognition methods. The concept is to choose the proper element from **I**, **II**, **III**, and **IV** for each of the $\mathbf{S}_i$ in Figure 1.

```
I.      intensity                |
        range                    |   sensed data
        density                  |
        heat                     |
                                         ||
                                         ||
                                        \  /
                                         \/

II.     line segments            |   low level processing
        regions                  |
        texture                  |
                                         ||
                                         ||
                                        \  /
                                         \/
III.    features vector          |
        graph of features        |   derived description
            & relations          |
        generalized cylinders    |
                                         ||
                                         ||
                                        \  /
                                         \/
IV.     distance                 |
        parse                    |   matching paradigms
        graph matching           |
        tree search              |
```

Figure 2. Schema for Generating Recognition Strategies

3. Future Work

We are currently implementing the system. During the course of this work, we have encountered several interesting questions:

1. What properties of the shape of an object can be used to choose a recognition scheme?
2. Are some recognition schemes more appropriate for certain classes of shapes?

Typically, recognition schemes found in the literature are implicitly tied to specific classes of shapes (despite claims to the contrary). Other problems include the completeness of a recognition scheme given a restricted number of views of the unknown object. Given that we are working in a model-based vision domain, it may be possible to find a set of features which completely characterize the object from any view.

Another important area which we are only beginning to study is the use of the system for robot manipulation. This offers a potential boon in task-oriented robot programming in that it may be possible to decentralize and distribute the manipulation task based on manipulation strategies produced by the system.

References

[1] Ballard, D.H. and C.M. Brown.
Computer Vision.
Prentice Hall, New York, 1982.

[2] Barrow, Harry and Jay Tennenbaum.
Recovering Intrinsic Scene Characteristics from Images.
Technical Report 157, SRI International, April, 1978.

[3] Hansen, C., T.C. Henderson, Esther Shilcrat and Wu So Fai.
Logical Sensor Specification.
In *Proceedings of SPIE Conference on Intelligent Robots*, pages 578-583. SPIE, November, 1983.

[4] Henderson, T.C., E. Shilcrat and C. Hansen.
A Fault Tolerant Sensor Scheme.
Computer Science UUCS 83-003, University of Utah, November, 1983.

[5] Henderson, T.C., E. Shilcrat and C.D. Hansen.
A Fault Tolerant Sensor Scheme.
In *Proceedings of the International Conference on Pattern Recognition*, pages 663-665. August, 1984.

[6] Henderson, T.C., Bir Bhanu, C.D. Hansen and E. Shilcrat.
ASP: A Sensor Performance and Evaluation System.
In *Proceedings of Pecora IX*, pages 201-207. October, 1984.

[7] Henderson, Thomas C., Bir Bhanu and Chuck Hansen.
Distributed Control in the Multisensor Kernel System.
In *Proceedings SPIE Conference on Intelligent Robots*, pages 253-255. Cambridge, Massachusettes, November, 1984.

[8] Henderson, T.C. and C.D. Hansen.
A Kernel for Multi-sensor Robotic Systems.
In *Proceedings of the CAD/CAM, Robotics and Automation Institute and conference*, pages to appear. February, 1985.

[9] Henderson, T.C., C.D. Hansen, and Bir Bhanu.
The Specification of Distributed Sensing and Control.
Journal of Robotic Systems :to appear, 1985.

[10] Henderson, T.C., Bir Bhanu and Chuck Hansen.
"Intrinsic Characteristics as the Interface between CAD and Machine Vision Systems.
Pattern Recognition Letters :to appear, 1985.

[11] Henderson, T.C., Chuck Hansen and Wu So Fai.
Organizing Spatial Data for Robotic Systems.
Computers in Industry :to appear, 1985.

[12] Henderson, T.C., Chuck Hansen and Bir Bhanu .
"A Framework for Distributed Sensing and Control.
In *Proceedings of IJCAI 1985*, pages to appear. Los Angeles, CA, August, 1985.

[13] Horn, B.K.P.
Obtaining Shape from Shading Information.
In P. Winston (editor), *The Psychology of Computer Vision*, pages 115-155. McGraw-Hill, New York, 1970.

[14] Rosenfeld, A. and A. Kak.
Digital Picture Processing.
Academic Press, New York, NY, 1976.

[15] Witkin, A. P.
Recovering Surface Shape and Orientation from Texture.
Artificial Intelligence 17:17-45, 1981.