3D Modelling and Indexing for CAD-based Object Recognition

Leda Villalobos and Francis L. Merat

Electrical Engineering Department
Center for Automation and Intelligent Systems Research
Case Western Reserve University, Cleveland, OH 44106-7221

Abstract

A 3-D object recognition system for parts designed with a form feature CAD is presented. Objects are modelled as organized compositions of instantiated form primitives. Because of this, the burden of part recognition is shifted from matching the scene to the part database to matching the scene to the primitive database, which is usually a significantly smaller database. In this paper, we briefly review our approach to primitive and object representation, the architecture of the system, hypotheses generation, and primitive indexing. Results obtained with both synthetic and actual range images are presented.

1. Introduction

Increased production efficiency and quality are the fundamental promises of Computer Integrated Manufacturing (CIM). These promises have prompted massive research initiatives into the different aspects of CIM, among them concurrent engineering [7,8], automatic program generation [4], and autonomous manufacturing [5]. The growing interest in CIM has also made an impact on the philosophy of object recognition for manufacturing, since the use of CAD descriptions for automatic object modelling leads to considerable time savings in the development of vision systems [1, 2].

In this paper, an ongoing research effort to develop a 3-D object recognition system for CAD form features, e.g. hole, slot, pocket, etc., will be discussed. The heart of the approach resides in modelling objects as collections of inter-related manufacturing primitives. This allows the main recognition task to be broken down into less stringent ones of qualitative primitive identification. A hybrid neural net/rule-based expert system is being developed to perform this feature recognition.

In section 2, we introduce the primitives currently supported by the recognition system, and the representation schemes used to model these primitives and objects. System architecture is briefly described in section 3, while hypotheses generation and verification are presented in sections 4 and 5. Results for synthetic and actual range images are shown in section 6, with conclusions in section 7.

2. 3D form Feature Modelling

Central to the design of a recognition system is the scheme adopted for object modelling. Model representation plays a fundamental role in database organization, choosing search and matching algorithms, processing the original image data, and in the specification of the overall flow of information and control. In this research, objects are constructed from the combination of a selected set of form manufacturing primitives. In the following sections, these primitives and their modelling scheme are introduced.

2.1. Manufacturing Primitives

The topological/geometrical primitive representation is examined in the context of a selected, reduced set of manufacturing primitives. Nine form manufacturing primitives were chosen: through hole, blind hole, boss, rib, through slot, step, step to shoulder, pocket, and edge cut. Their variety is sufficient for designing a substantial number of different parts, thus validating recognition methodology; nevertheless, they are simple enough to facilitate implementation and testing. Surface connectivity graphs for these primitives appear in Figure 1.

Primitives are modelled at two levels: surface topological arrangement, and surface geometry.

2.2. Topological Modelling

In this research, a viewer centered approach is taken to represent the form primitives. This representation is, nevertheless, topological in nature. Contrary to most vision systems which restrict aspect views —or aspect, for

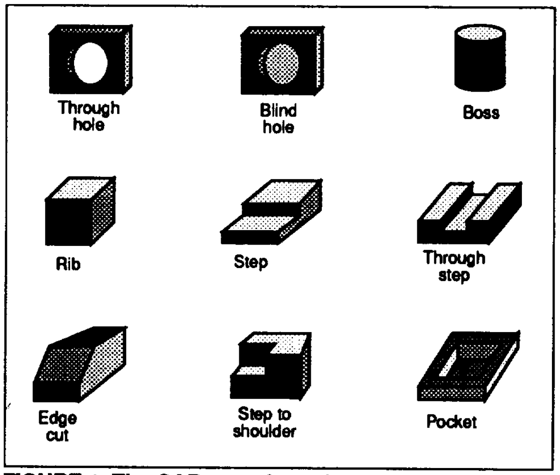


FIGURE 1. The CAD manufacturing primitives.

short—to be purely edge constructions, our approach introduces a topological representation graph for each primitive aspect. An aspect is topologically described by a combination of surfaces with specific unary and binary relational properties.

At the topological level, the only unary feature included is the shape of the surface patch. To generate the topological graph representation of an aspect, three binary relations are extracted for every pair of surfaces: their relative orientation, spatial proximity, and geometric equivalence.

Since relative orientations are local, intrinsic properties, they are frequently used as features for 3D modelling. The relative orientation between two surfaces is given by the angle formed by their orientation vectors. For this purpose, the orientation of a planar surface is given by its normal; for a cylindrical surface, it is given by the axis direction. Because of the characteristics of the selected primitives, we classify the orientation relations as parallel, perpendicular, or oblique.

In terms of spatial proximity, we are interested in establishing whether the surfaces are physically adjacent. The condition of adjacency requires the surfaces to share at least a portion of their physical boundaries; the shared edge is either concave or convex. Geometric equivalence refers to those cases in which non-contiguous surfaces exhibit the same geometric characteristics. A summary of the topological features which can exist between any two surfaces for the selected primitives is given in TABLE 1.

Representative examples of topological relational graphs for two aspects are shown in Figure 2. It should be emphasized that the simplicity of the selected primitives makes compact topological graphs possible. For primitives involving complex surface parametrization, such as splines, the representation of topological properties would require more elaborate relational trees.

TABLE 1. Topological Features

FEATURE	Orientation	Adjacency	Equivalence
LEC	Parallel	Convex	No
LIC	Parallel	Concave	No
LID	Parallel	No	No
PEC	Perpendicular	Concave	No
PIC	Perpendicular	Concave	No
PID	Perpendicular	No	No
ANC	Oblique	Concave	No
PSE	Parallel	No	Yes

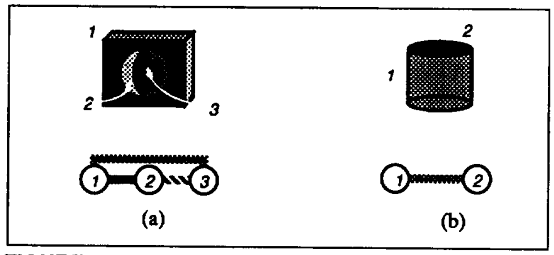


FIGURE 2. One of the aspects and topological graphs for (a) a blind hole, and (b) a boss. Thick, dark links in the graph indicate a LIC topological relation; thick light links indicate a LEC relation; and thick, dashed links indicate a LID relation.

2.3. Boundary Modelling

Boundaries are used to complete surface modelling by detailing geometry. In this research, we use boundary modelling as a tool for verifying/rejecting primitive hypotheses. The geometric descriptions include the type of surface and any available boundary information. Faces are qualitatively described by construction of relational edges or boundary groups, since it is considered that attempting the detection of loops might be an unnecessarily expensive task.

Surface geometric descriptions can be done by tracing the edges identified through the application of classical edge detectors or by detecting the transitions from one surface patch to another. Because of the difficulties in dealing with edge detectors, this later approach is preferred.

2.4. Part Modelling

In the part database, each part is described as an organized aggregate of instantiated manufacturing primitives. A primitive is instantiated by indicating the dimensions of its characteristic parameters, and its location in the part with respect to a global reference frame.

The characteristic parameters are instantiated following the convention shown in Figure 3. For example, a through hole is characterized by its radius and the orientation of its axis. The Z axis of the local frame runs along the axis of the hole, pointing out of the material, while the origin is located at the opening of the hole.

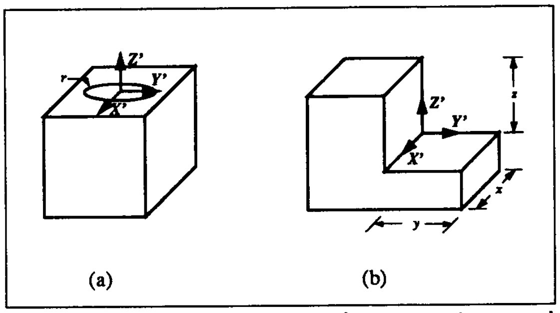
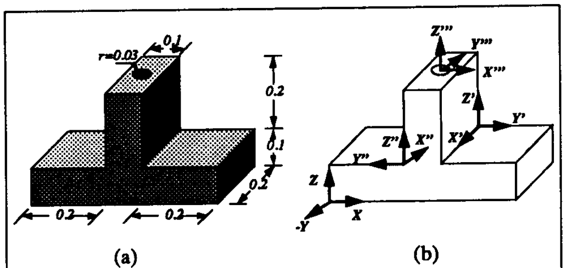


FIGURE 3. Primitive geometric parameters and definition of local reference frames. (a) A through hole is completely characterized by its radius and the axis orientation. (b) The geometry of the step is defined by the dimensions of the cut.

To instantiate the location of a primitive, a global reference frame which remains available throughout the fabrication of the part, is first chosen. The pose of a primitive is given by the rotation matrix between the global and local reference frames, and the position vector which goes from the origin of the global reference frame to the origin of the local reference frame.

The model of a part has four slots. The first slot identifies the part with an ID number and a descriptive label. Next, the second slot gives the dimensions of the stock, while the third one indicates the types of primitives present in the part. Finally, the last slot lists the instantiated primitives organized as follows: the primitive type, the number of occurrences of that primitive type in the part, and the full description of the primitives. This description includes an identification number and two vectors, one with the geometric parameters and the other with the pose. Figure 4. shows the model of a vise jaw created with two steps and a through hole. This information can be readily obtained from the standard output of a form feature CAD system [6].

It is important to emphasize that, with this modelling style, the part database becomes a collection of compact, simple linguistic descriptions. Even more, the addition of new models to the database does not involve any modifications to the structure of the recognition system itself. This is so because, for recognition purposes, indexing specific items from the part database first requires the identification of the primitives present in the image.




```
model(part_id(1,vise_jaw),
    stock(0.5,0.2,0.3),
    primitives([step,th_hole]),
    description([step(2, [(1,param([0.2,0.2,0.2]),
                    pose([0,1,0],[-1,1,0],[0,0,1],[0.3,0.2,0.1])),
                 (2,param([0.2,0.2,0.2]),
                    pose([0,-1,0],[1,0,0],[0,0,1],[0.2,0,0.1]))]),
        th_hole(1, [(1,param([0.03]),
                    pose([1,0,0],[0,1,0],[0,0,1],[0.25,0.1,0.3]))])])).
```

(c)

FIGURE 4. A part and its composite primitive description. (a) Dimensioning for a vise jaw; (b) location of the global reference frame *XYZ*, and the local reference frames for every primitive; (c) symbolic description of the part as a conglomerate of instantiated primitives.

Primitive indexing is described in the following sections.

3. System Architecture

A block diagram for the architecture of the recognition system appears in Figure 5. The system has five major modules: an expert system, a bank of neural networks, static and dynamic databases, and an image segmentation module. Brief descriptions of these modules are given next.

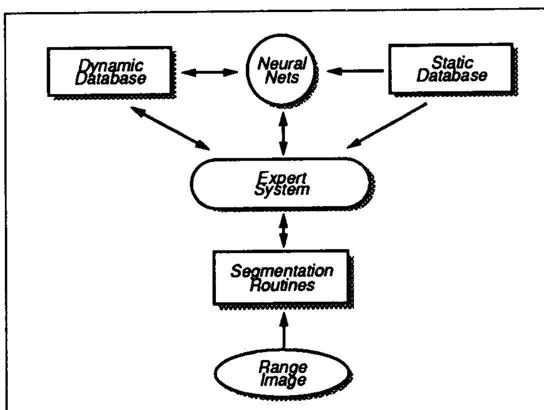


FIGURE 5. Object recognition system organization.

The expert system operates as the central controller of data flow and interpretation, carrying out deductive functions and generating hypotheses. Primitive aspect models (topological and geometric), and object models are stored in three static databases. All intermediate hypotheses and the results of their verification routines are kept in dynamic databases. Hopfield neural nets are used for graph matching in hypotheses pruning tasks. Surface and edge segmentation are neural based [9].

4. Topological Primitive Indexing

As it was mentioned before, the core of recognition by components is the detection and identification of instances of form primitives in the image data. In this section, we present the primitive indexing paradigm in detail.

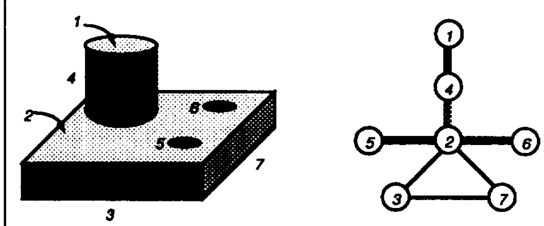
4.1. Topological Graphs

An undirected, attributed graph G is used to describe the topological relations between the surfaces of a segmented image. Each surface visible in the segmented image is assigned a node in the graph, and for any two adjacent surfaces, there is a link connecting their corresponding nodes. Nodes are labeled according to the type of the associated surface, while links are labeled according to the topological relations of the surfaces interactions. The dictionary of topological relations is the same as that introduced in Table 1. An example of a topological graph constructed for a segmented image appears in Figure 6.

4.2. Preliminary Topological Indexing

The goal of the preliminary hypotheses generation paradigm is to produce a coverture P of G, $P=\{P_j, j=1,2,..., K\}$, where P_j is a path of G for which there is supporting topological evidence that it could correspond to an instance of a primitive. Each path P_j has associated two sets of hypotheses, H_{cj} and H_{pj}. H_{cj} is the set of *complete aspect hypotheses*, while H_{pj} is the set of *partial aspect hypotheses* for path P_j. Complete aspect hypotheses are those primitive instances for which P_j could be a topological isomorphism; partial aspect hypotheses are those primitive instances for which P_j could be a topological sub-isomorphism.

The generation of hypotheses from topological information is done by first finding all paths in the topological image graph which are compatible with the form primitives. Valid paths are created with a breath first search on the connectivity of each node in the image graph. The search in each branch of the tree is pruned whenever the corresponding qualitative connectivity graph does not match any primitive pattern.



(a) (b)

```
graph([(node(1),plane,links([(4,lec)])),
      (node(2),plane,links([(3,pec),(4,lic),(5,lec),(6,lec),(7,pec)])),
      (node(3),plane,links([(2,pec),(7,pec)])),
      (node(4),cylcx,links([(1,lec),(2,lic)])),
      (node(5),cylcc,links([(2,lec)])),
      (node(6),cylcc,links([(2,lec)])),
      (node(7),plane,links([(2,pec),(3,pec)]))]).
```

(c)

FIGURE 6. Constructing the topological graph of a segmented image. (a) Segmented range image with the individual surfaces identified; (b) corresponding topological graph; (c) symbolic representation as a PROLOG clause.

To accelerate topological indexing, we perform a qualitative pattern matching between the image and the primitives aspect graphs. The pattern has a total of 11 features organized in two groups, one related with the surface type description, and the other with the topological relations among the surfaces. The surface type portion of the pattern has 3 entries, indicating the number of planar, convex, and concave surfaces in the path, respectively. The topological description pattern has 8 features, which represent the sum of the graph's links with the topological relations shown in TABLE 1.

4.3. Topological Hypotheses Organization

Since graph matching is a computationally intensive task, the actual evaluation of preliminary topological hypotheses needs to be done in an organized, hierarchical fashion. To this end, preliminary hypotheses are organized in levels according to their potential evaluation return rates, estimated with two heuristics:

- Intuitively, a path having a qualitative correspondence with a complete aspect carries on more information than, say, a path corresponding only to a sub-set of an aspect. Primitive indexing is more reliable when primitives appear without occlusion or are not otherwise ambiguous.
- If there are two paths, one longer than the other, the

longer path has the potential to reduce the entropy of recognition more than the shorter path. This heuristic is motivated on the fact that several primitives can be topologically interpreted as being composed of simpler primitives. For example, a step to a shoulder can be seen as the interaction of two steps.

Based on these heuristics, we have grouped the preliminary hypotheses in three levels. The highest level of the hierarchy includes those paths which are not subsets of other paths, and have a non-empty set of complete aspect hypotheses. The next level includes those paths which are not subsets of other paths, and have a non-empty set of partial aspect hypotheses. Notice that it is possible to find the same path in both the first and second hierarchical levels. The last level of the hierarchy includes all remaining preliminary hypotheses paths. We refer to the first level of the hierarchy as the *primary topological hypotheses*, to the second as the *secondary topological hypotheses*, and to the third as the *tertiary topological hypotheses*. The third level is seldom used and we will not make any further reference to it in this paper.

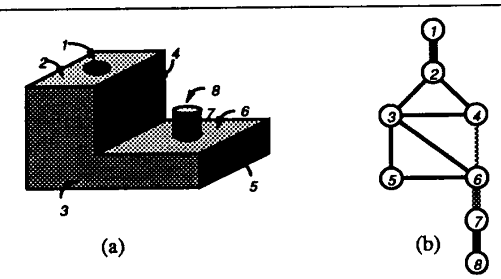
Once hypotheses are divided, they are pruned. Not all hypotheses are tested, though. Initially, only primary hypotheses examined. If the evaluation of these hypotheses is not sufficient to fully explain the scene, selected secondary hypotheses are also evaluated. Finally, selected tertiary hypotheses might have to be evaluated, although this situation would be extremely rare. Pruning is performed by evaluating the isomorphism or sub-isomorphism of the topological graph of the path with respect to the graph of the hypothesized primitive instance. This verification is carried out with optimizing neural networks. Neural net pruning is explained in Section 5.

4.4. Examples

Two representative examples are given next.

Example 1. Figure 7 shows a very simple part and its surface connectivity graph. The part is composed of a step with an attached boss and hole. In this case, three primary topological hypotheses are generated. Surfaces 1 and 2 are considered to be either part of a through hole (aspect 1), or a blind hole (aspect 2). Surfaces 7 and 8 are considered to be part of a boss (aspect 3). Finally, surfaces 2, 3, 4, 5, and 6 are considered to be part of a step (aspect 10). No secondary topological hypotheses are generated.

Example 2. Topological hypotheses for a more interesting part appear in Figure 8. The part has two primitives, a step to shoulder and a through slot. A number of primary topological hypotheses, some of them incorrect, are generated. As it will be seen in Section 5, the false hypotheses are pruned once the connectivity patterns of each node in the surface connectivity graph is analyzed. The correct hypotheses for the step to shoulder (aspect 2) and the through slot (aspect 8), are also generated. There are also several secondary hypotheses, which appear because portions of the image graph partially match the patterns that qualitatively define both primitives.

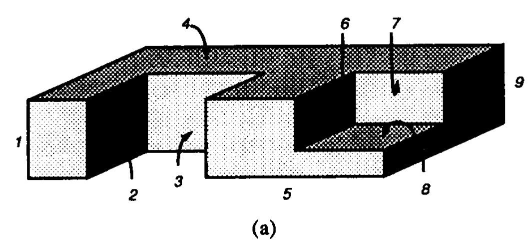


Primary_topological_hypotheses = [path([1,2], [th1, bh2]),
path([7,8],[bs3]),
path([2,3,4,5,6],[s11])]

Secondary_topological_hypotheses = []

(c)

FIGURE 7. Initial generation of topological hypotheses. (a) A part consisting of a step, with a through-hole and a boss; (b) its corresponding surface connectivity graph; and (c) primary and secondary topological hypotheses.



Primary_topological_hypotheses = [path([2,4,6,8], [s6]),
path([1,4,6,8], [s6]),
path([3,4,6,8], [s6]),
path([2,4,7,8], [s6]),
path([1,4,7,8], [s6]),
path([3,4,7,8], [s6]),
path([1,2,3,4,5], [ts8]),
path([4,5,6,7,8,9], [ss2])]

(b)

FIGURE 8. Initial generation of topological hypotheses. (a) A part consisting of a through step and a step to shoulder, and (b) its corresponding primary topological hypotheses

5. Neural Net Hypotheses Verification

The purpose of this matching is twofold. First, success/failure at matching the graphs further prunes the set of hypotheses; second, should the matching be successful, the solution of the surface correspondence problem would greatly simplify future hypotheses verification through surface fitting or edge tracing, since it becomes a model driven task.

Multiple examples of Hopfield neural nets for object identification through graph matching have been reported in the literature [3]. The common denominator in these applications is that a graph created by extracting significant features from the image is matched against the graph of a part model. Nodes in the graphs represent features and their local properties; links represent relational properties or constraints, such as distances between features.

5.1. Surface correspondence constraints

The desired solution to our topological isomorphism problem requires the satisfaction of three types of constraints:

- *Uniqueness correspondence*; each surface in the image eventually matches only one surface in the model, and vice versa.
- *Shape correspondence*; the shape of an image node must be identical to the shape of the associated model node.
- *Topological correspondence*; the connectivity pattern of an image node must be identical to the connectivity pattern of the associated model node.

We have encoded these constraints in the energy function of a Hopfield optimization neural net [10]. The final stable state of the network indicates whether the matched primitive and image graphs are isomorphic. In case they are isomorphic, the neurons activation pattern shows the surface correspondence existing between both graphs.

5.2. Neural Net Indexing Examples

Two representative examples are given. In the first example, the correspondence problem is solved for two isomorphic topological graphs. The second example consists of examining the possibility of finding a correspondence mapping between an image graph and a model graph which are not isomorphic.

Example 3. In this example, the problem consists in matching the surface connectivity graph of a region of interest to the model of a step's aspect, as shown in Figure 9. Due to symmetry, there are two solutions. In the first solution, the correspondence pairs (I_i,M_j) between the i-th image surface and the j-th model surface are $\{(1,2),(2,1),(3,4),(4,5),(5,3)\}$. In the second solution, the correspondence pairs are $\{(1,3),(2,5),(3,4),(4,1),(5,2)\}$. Any one of these two solutions can be the final stable state of the neural net.

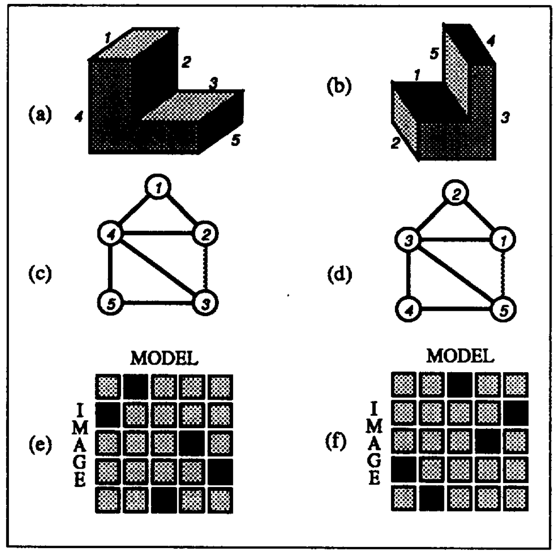


FIGURE 9. Solving the surface correspondence problem with a neural net. (a) A model aspect of a step (b) its corresponding surface connectivity graph; (c) surface assignment for an image region of interest; (d) its corresponding surface connectivity graph; (e) a final stable state for a particular correspondence solution; (f) stable state for another feasible correspondence solution.

Example 4. Here, the neural net tries to find a correspondence function for a step's aspect and a portion of the image of Example 2. As it is seen in Figure 10, from the final state of the network it is evident that no isomorphism could be identified.

6. Recognition Results

Representative results obtained with both synthetic and actual range images are presented next.

6.1. Synthetic Range Images

In Figure 11 we revisit Example 2. Recall that the set of primary topological hypotheses included a number of false indexings. Note how well the neural nets prune the initial set of primary topological hypotheses, and topologically

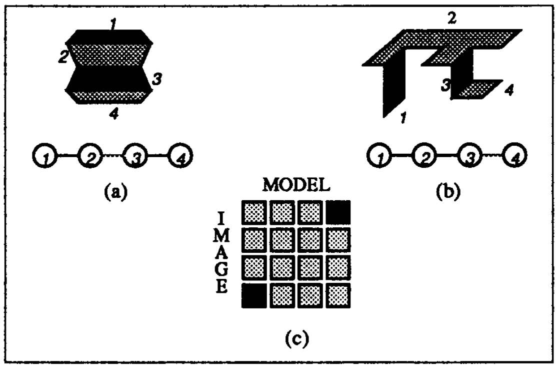


FIGURE 10. Detecting a failed graph matching. (a) A model aspect of a step; (b) a set of surfaces hypothesized as matching the model; and (c) final neural net state. Notice that no isomorphism is found.

verify the presence of a step to shoulder (aspect 2) and a through slot (aspect 8).

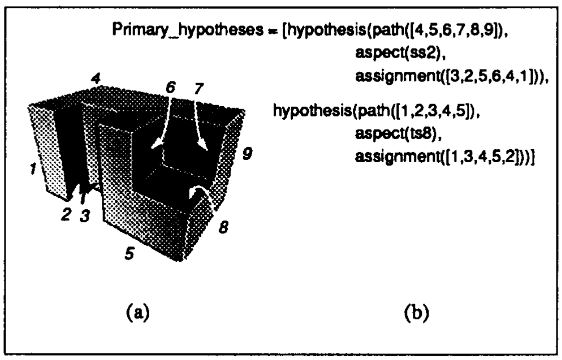


FIGURE 11. Topological hypotheses for synthetic range image of Example 2. (a) Pseudo-intensity image, and (b) primary hypotheses. Both the through slot and the step to shoulder primitives are properly recognized.

In Figure 12, we have an image with two objects, one composed of a step with a hole, and the other of two bosses. Here again the system succeeds at properly identifying the individual CAD primitives. There are two hypotheses to explain surfaces 3 and 5. The first hypothesis considers the surfaces to be part of a through hole, and the second to be part of a blind hole. The geometric verification routines would examine both hypotheses and eventually declare the blind hole an invalid one.

6.2. Actual Range Images

Results for actual range images are given in Figures 13 and 14. These images were originally scanned at the Pattern Recognition and Image Processing Laboratory, Michigan State University, and are now available in an archive set up at Washington State University by Professor P. Flynn. They were produced with a Technical Arts 100X scanner.

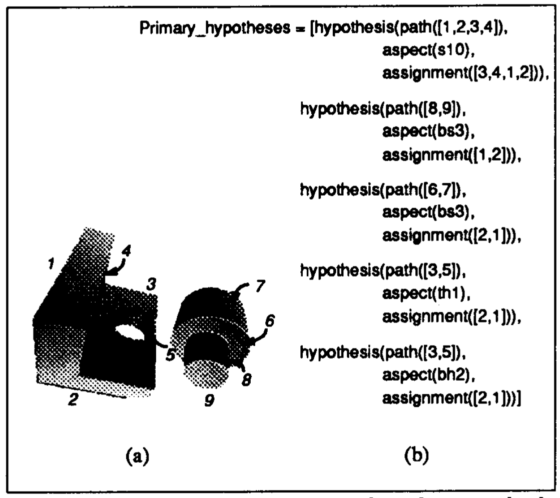


FIGURE 12. Hypotheses generation for synthetic range image. (a) Pseudo-intensity image, and (b) corresponding topological hypotheses.

In Figure 13, we have an image of a block with a column, which can be described as a boss attached to a rib. Two primary hypotheses are topologically verified by the neural nets, one for a rib (aspect 1), and the other for a boss (aspect 3).

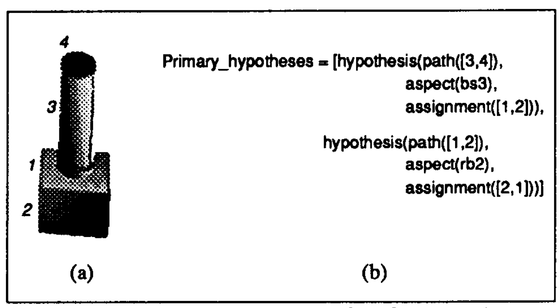


FIGURE 13. Hypotheses generation for actual range image of a column. (a) Pseudo-intensity image, and (b) corresponding topological hypotheses.

Figure 14 shows results for an image of a tape roll. In terms of our CAD primitives, the tape roll can be described as being composed of a boss with a through hole. Here again, the boss is correctly identified, while the hole is explained as being either a blind hole (aspect 2) or a through hole (aspect 1). Note how the topological description simplifies dealing with incomplete surfaces, making recognition robust.

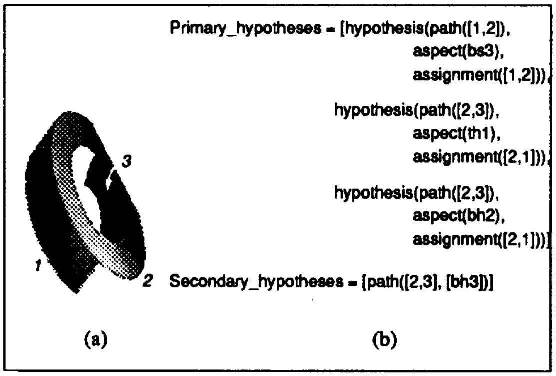


FIGURE 14. Hypotheses generation for actual range image of a tape roll. (a) Pseudo-intensity image, and (b) corresponding topological hypotheses.

7. Conclusions

We have described a 3-D object recognition system for parts designed with a form feature CAD system. In this system, objects are modelled as organized compositions of instantiated form primitives. Primitives are modeled at the topological and geometrical levels. At the topological level, primitives are described as collections of surfaces with specific topological relations. Primitive indexing is done in two steps: a fast, qualitative pattern matching, and a more attentive, relational graph matching.

This scheme offers the following advantages:

- In terms of computational recognition time, the representation shifts the burden of sub-graph matching from the object model database to the CAD primitive model database. Since the size of the primitive database remains constant for a particular CAD system and is considerably smaller than the object database, the matching costs are substantially reduced.
- Addition of new items to the object database has little impact on recognition complexity both in terms of matching time and in modification of recognition heuristics.
- Since the core of an object representation is its manufacturing feature description, interfacing with CAD/CAM software is simplified thus permitting easy compatibility within the CIM production system.

8. References

[1] Arman, F., and J.K. Aggarwal, "CAD-Based Vision: Object Recognition in Cluttered Range Images Using Recognition Strategies," *CVGIP. Image Understanding*, Vol. 58, No. 1, pp. 33-42, 1993.

[2] Flynn, P., and A.K. Jain, "CAD-Based Computer Vision: From CAD Models to Relational Graphs," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 13, No. 2, pp. 114-132, 1991.

[3] Lin, W-C., F-Y. Liao, C-K. Tsao, and T. Lingutla, "A Hierarchical Multiple-View Approach to Three-Dimensional Object Recognition," *IEEE Trans. on Neural Net.*, Vol. 2, No. 1, pp. 84-92, 1991.

[4] Masotti, G., and T. Bombardi, "Automatic production of NC code for machining from features in generic parts," *Computing and Control Engineering Journal*, Vol. 3, No. 6, pp. 287-295, 1992.

[5] Marefat, M., M. Sandeep, and R.L. Kashyap, "Object-Oriented Intelligent Computer Integrated Design, Process Planning, and Inspection," *Computer*, Vol. 26, No. 3, pp. 54-65, 1993.

[6] Merat, F., and G. Radack, "Automatic Inspection Planning within a Feature-Based CAD System," J. Robotics and Computer Integrated Manufacturing, Vol. 9, No. 1, pp. 61-69, 1992.

[7] Prasad, B., R.S. Morenc, and R. Rangan, "Information Management for Concurrent Engineering: Research Issues," *Concurrent Engineering: Research and Applications*, Vol. 1, No. 1, pp. 3-20, 1993.

[8] Sarkar, B., and C-H. Menq, "Smooth-surface approximation and reverse engineering," *Computer Aided Design*, Vol. 23, No. 9, pp. 623-633, 1991.

[9] Villalobos, L., and F. Merat, "Neural Net Range Image Segmentation for Object Recognition," *Proc. Applications of Artificial Neural Networks IV*, SPIE 1965, Orlando, 1993.

[10] Villalobos, L., and F. Merat, "Recognition of Mechanical Form Features in Range Images," *CAISR Tech. Report 93-132, Case Western Reserve University*, 1993.