TR-92-004

REDUCING SEARCH SPACE FOR 3-D OBJECT RECOGNITION USING ART-1 NEURAL NETWORK

Hong-Yuan Liao†
Institute of Information Science
Academia Sinica, Taipei, Taiwan

Cheng-Chung Liang and Wei-Chung Lin
Department of Electrical Engineering and Computer Science
Northwestern University
Evanston IL, 60208

email: liao@iis.sinica.edu.tw
FAX: 886-2-782-4814
TEL: 886-2-788-3799

† to whom correspondence should be sent

REDUCING SEARCH SPACE FOR 3-D OBJECT RECOGNITION USING ART-1 NEURAL NETWORK

Hong-Yuan Liao†
Institute of Information Science
Academia Sinica, Taipei, Taiwan

Cheng-Chung Liang and Wei-Chung Lin
Department of Electrical Engineering and Computer Science
Northwestern University
Evanston IL, 60208

email: liao@iis.sinica.edu.tw
FAX: 886-2-782-4814
TEL: 886-2-788-3799

† to whom correspondence should be sent

REDUCING SEARCH SPACE FOR 3-D OBJECT RECOGNITION USING ART-1 NEURAL NETWORK

Hong-Yuan Liao†
Institute of Information Science
Academia Sinica, Taipei, Taiwan

Cheng-Chung Liang and Wei-Chung Lin
Department of Electrical Engineering and Computer Science
Northwestern University
Evanston IL, 60208

Abstract

In this paper, we propose a method to reduce the search space for 3-D object recognition using an ART-1 neural network. We also present a procedure to automatically construct CV libraries for modeling polyhedral objects. This procedure is considered as a fundamental process in the multiple-view approach to 3-D object recognition. Although there is no redundancy in the resultant CV library, the size of the library is still large if the target object is complex in shape. To increase the efficiency of the recognition process, a coarse-to-fine search strategy is adopted. In the coarse search phase, an ART-1 neural network is used to locate a set of CVs in the library that is most similar to the projection of the object. In the fine search process, those located CVs are used as the starting points for exhaustive search. Experimental results corroborating the theory are reported.

---

† to whom correspondence should be sent

1. INTRODUCTION

Three-Dimensional (3-D) object recognition is the process of matching an object to a scene description to determine the object's identity and/or its position and orientation in space [1-3]. Any system capable of recognizing its input image (intensity or range image) must in some sense be model-based. The problem of object recognition can be separated into two closely related subproblems - that of model building and that of recognition. There are different approaches to both these subproblems, and the procedure used for recognition will have a strong impact on the kind of model that will be required and vice versa. An object can be modeled by 3-D configurations of its surface [4,5] or volume [6,7] primitives having specified attributes and relationships. Alternatively, objects can be characterized more abstractly, using symbolic expressions. They are referred to as geometric models and symbolic models, respectively [8]. The easiest way to build an object model is to construct it manually by using CAD techniques. However, automatic generation of a 3-D object model from multiple views is always a better choice if possible. The model can be stored as a single 3-D representation in an object-centered coordinate system. It can also be stored as multiple 2-D projections in a viewer-centered coordinate system [9,10]. The nature of the recognition process depends on the form of the object model. Geometric models lend themselves to the hypothesis-verification paradigm [8] while symbolic models can be matched to symbolic scene descriptions using formal or heuristic graph-matching techniques.

The multiple-view approach [9-13] models objects by the set of its 2-D projections as seen from a set of predetermined viewpoints on the view sphere. This approach requires more storage space than the single-3D-model scheme since a large number of different 2-D projections are needed to fully represent the target object. However, the multiple-view approach offers one major advantage over the single-3D-model approach due to the fact that the features extracted from images can be directly matched with those associated with each member of the multiple-view model set. This circumvents the difficult problem of determining the transformation between 2-D (i.e. viewer-centered coordinate system) and 3-D (i.e. object-centered coordinate system) spaces which is always encountered in the single-3D-model approach.

In [13], we have proposed a computer system which automatically constructs multiple-view model database for polyhedral objects. This database is organized as a graph in which a node represents a characteristic view (CV) [10-12] and an arc represents the transformation between two CVs. It is also referred to as a CV library (or aspect graph). In constructing a CV library, a set of viewpoints uniformly distributed on a view sphere enclosing a target object is selected and a set of features from the projection of the target object is extracted from each viewpoint. Based on this feature set, a region growing process is then applied on the view sphere to form a finite set of regions so that each region on the partitioned view sphere constitutes a CV. In other words, all viewpoints in a region visualize the same topology with slightly different geometries. This procedure significantly reduces the size of the model database by merging those neighboring and topologically equivalent views.

Although the redundancy of the model database has been reduced in the CV library generation process, the size of the library is still large if the target object is complex in shape. After the model database is constructed, the next problem is to determine where we should start the search process. This decision is crucial since a good selection of start point brings us close to the solution and thus vastly reduces the search time. In this paper, we propose to use an ART (Adaptive Resonant Theory)-1 neural network [14,15] to accomplish this task. The ART-1 neural network basically clusters binary vectors into categories. It possesses distinctive properties such as self-adjusting memory search and direct access. With self-adjusting memory search, it is capable of doing parallel memory search that adaptively updates its search order to maintain efficiency while its categories resulting from clustering becomes arbitrary complex in the course of learning [14]. With direct access, it can activate the category of a familiar input pattern without searching.

These functions enable a trained ART-1 system to locate candidate CVs in real time. In most situations, this step vastly decreases the search space. The candidate CV selection using ART-1 network is considered as the coarse search operation in a coarse-to-fine search process. The fine (or exhaustive) search process starts with these candidate CVs. It compares the topology and geometry of the unknown object (in the form of line drawings after image processing) and those projections in the neighborhood of these candidate CVs until a best match is found.

The remainder of this paper is organized as follows. In Section 2, the automated CV library generation process is described. In Section 3, the coarse-to-fine search process is presented. Then, some experimental results are reported in Section 4. Finally, the advantages of using ART-1 network in multiple-view approach to object recognition is discussed in Section 5.

2. AN AUTOMATED PROCEDURE FOR GENERATING CV LIBRARY

The model-building task in the multiple-view approach to 3-D object recognition requires that all the topologically different 2-D projections (i.e., CV library) be exhaustively generated and stored. To our knowledge, there is no automated procedure proposed to accomplish this task. This work is generally done by hand, requiring some degree of faith on the system designer's ability to exhaustively enumerate all the CVs. The reason why an automated procedure is still unavailable may be attributable to the lack of criteria for distinguishing topological changes between two neighboring views.

In this paper, we present a procedure for automatically generating CV libraries for polyhedral objects. One of the distinctive features of this procedure is the selection of a minimum feature set for determining whether two neighboring views are topologically equivalent. The details of this feature set will be presented in Section 2.2.2.

The process of automated CV library construction consists of three major tasks. First of all, a systematic method is needed to determine a set of viewpoints uniformly distributed on the view sphere. Thus, the view sphere is preliminarily partitioned. During the subsequent CV library construction process, in order to determine whether the topologies of two projections visualized from any two viewpoints are equivalent, a representation scheme is needed to describe the topology of a projection (in the form of line drawings). The second major task is to devise a feature vector for comparing two projections. Finally, a region growing process is needed to merge the connected regions with topologically equivalent CVs. Since the second task - feature set selection - is vital in understanding the operation of the ART-1 network, it will be elaborated in Section 2.2. The other two tasks will only be briefly described for completeness. For illustrative purpose, we use a polyhedral object shown in Fig.1 as a running example throughout this paper.

2.1. Viewpoint Determination

The process of viewpoint determination is the first step toward automated multiple-view model generation. The major goals of this task are : (1) to find a systematic viewpoint generation method which calculates the coordinates of all the viewpoints on the view sphere and, (2) to design a data structure which organizes the generated viewpoints and builds up the relationships between each viewpoint and its neighbors in an efficient manner so as to facilitate the region growing procedure discussed in the subsequent section.

During the construction of the library of 2-D projections (or models), the object of interest is viewed from different directions in a fixed distance. The loci of all the possible viewpoints constitute a view sphere. To approximate the view sphere, we start with a regular polyhedron. In this paper, we use an icosahedron to approximate the view sphere. Each face of the icosahedron is divided recursively into 16 smaller triangular facets. The 320 facets† are finally projected onto the view sphere. The centroid of each facet is used

† The objective of the proposed method is to generate the CV library automatically. The number "320" which is reasonable and adequate in most situations is determined empirically.

as the viewpoint. Fig.2 shows the simulation result after 320 facets are projected onto the view sphere.

2.2. Feature Set Selection

We have described how to obtain viewpoints on the view sphere systematically in the previous section. If we apply a straightforward library construction method to model the object from 320 viewpoints, the space complexity of the model database will be unnecessarily large. This is because some neighboring viewpoints on the view sphere visualize the same topology and it is redundant to put them all in the database. To rectify this problem, a set of criteria for distinguishing topological changes between neighboring views is required. In this section, we first describe a representation scheme for a 2-D projection in the form of line drawings. Then, a set of features for classifying 2-D projections is introduced. This feature set is proved [13] to be the minimum set required to detect topological change between two neighboring views.

2.2.1. Representation of a 2-D Projection

As we have mentioned earlier, the multiple-view approach requires a set of 2-D projections to fully represent an object. The recognition phase of this approach is simply to find a best match between the description of object projection and one of the model descriptions in the database. In this section, the structural description proposed in [18] is extended to a two-level scheme. Since we deal with polyhedral objects, topological changes due to self-occlusion can not be described by a single level description. The proposed two-level scheme is general enough to describe all topological variations for the projections of a polyhedral object. Besides, this representation scheme is rotation invariant.

Definition: A component contour is a simple closed curve composed of a set of component segments. Each segment is either a line segment or a curve segment. □

Definition: A structural description DP of a projection P is a pair $DP=(C,R)$. $C=\{C_1,C_2,...,C_n\}$ is a set of connected closed contours. Each component contour C_i ($1 \le i \le n$) has two kinds of descriptions. The first one is a binary relation $F_i \subseteq A \times V$ where A is a set of possible attributes and V is a set of possible values. The second one is also a structural description G_i which will be explained later. $R=\{CR_1,CR_2,...,CR_K\}$ is a set of named N-ary relations over C. For each $k=1,2,...,K$, CR_k is a pair (NR_k,R_k) where NR_k is a name for relation R_k, and for some positive integer M_k , $R_k \subseteq C^{M_k}$. Thus, set C represents the component contours of the projection, and set R represents the interrelationships among the components. G_i is a pair (CG_i,RG_i), $CG_i=\{S_{i,1},S_{i,2},...,S_{i,n_i}\}$ is a set of component (curve or line) segments, one for each of the n_i concatenated primitive segments of the component contour. Each component segment $S_{i,l}$, $1 \le l \le n_i$, is a binary relation $S_{i,l} \subseteq QA \times QV$ where QA is a set of possible attributes and QV is a set of possible values. $RG_i=\{CRG_{i,1},CRG_{i,2},...,CRG_{i,J}\}$ is a set of named N-ary relations over CG_i. For each $j=1,2,...,J$, $CRG_{i,j}$ is a pair $(NRG_{i,j},RG_{i,j})$ where $NRG_{i,j}$ is a name for relation $RG_{i,j}$ and for some positive integer $M_{i,j}$,$RG_{i,j} \subseteq (CG_i)^{M_{i,j}}$. Thus, set CG_i represents the component segments of the component contour C_i, and set RG_i represents the interrelationships among the segments. □

Example: Given a 2-D projection as shown in Fig.3, a complete two-level structural description is described as follows:

First-level structural description P

$P=\{C,R\}$

$C=\{C_1,C_2,C_3\}$

$C_1=\{$(shape,concave_polygon),(edge,6)$\}$

$C_2=\{$(shape,concave_polygon),(edge,5)$\}$

$C_3=\{$(shape,convex_polygon),(edge,4)$\}$

R = {(partial_connected,partial_set),(full_connected,full_set)}
partial_set = $\{(C_2,C_3)\}$
full_set = $\{(C_1,C_2),(C_1,C_3)\}$

Second-level structural description of component contour C_1 is G_1 where
$G_1 = \{CG_1, RG_1\}$
$CG_1 = \{S_{1,1}, S_{1,2}, S_{1,3}, S_{1,4}, S_{1,5}, S_{1,6}\}$
$S_{1,1}$ = {(type,line_segment)}
$S_{1,2}$ = {(type,line_segment)}
...
$S_{1,6}$ = {(type,line_segment)}

RG_1 = {($S_{1,1}$_neighbor,$S_{1,1}$_neighbor_set),
($S_{1,2}$_neighbor,$S_{1,2}$_neighbor_set),
...
($S_{1,6}$_neighbor,$S_{1,6}$_neighbor_set)}
$S_{1,1}$_neighbor_set = $\{(S_{1,2},S_{1,6})\}$
$S_{1,2}$_neighbor_set = $\{(S_{1,1},S_{1,3})\}$
...
$S_{1,6}$_neighbor_set = $\{(S_{1,5},S_{1,1})\}$
The structural descriptions of C_2 and C_3 are similar to that of C_1 and are omitted.
□

In this section, we have defined a complete structural description for a 2-D projection. However, a straightforward comparison of topologies based on this description is tedious. To alleviate this problem, we are urged to find a "condensed" feature set which contains less descriptional contents while still detects topological changes between two views rendered from two adjacent viewpoints.

2.2.2. Selecting a Proper Feature Set

By definition, any 2-D projection can be represented as a two-level structural description. The first-level description is $P=(C,R)$, where C is the set of component contours and R defines the relations among all elements in set C. $C_m=(CG_m,RG_m)$ (where $1 \le m \le |C|$, $|\ |$ represents the size of an arbitrary set) denotes the second-level structural description of the m-th component contour. CG_m is the set of all C_m's component segments and RG_m defines the relations among all the elements in set CG_m.

For any 2-D projection, there are four potential sources that may cause the change of topology. They are: C and R from the first-level structural description and CG_m, RG_m(for $1 \le m \le |C|$) from the second-level structural description. In [13], we proved that $|C|$, $|CG_m|$ (for $1 \le m \le |C|$) and R are potential sources for topological change. RG_m can be changed only when $|CG_m|$ is changed and vice versa. In other words, if we can detect the change of $|CG_m|$, we are sure that RG_m is also changed.†

The beauty of this feature set is that it enables us to detect topological change between two neighboring viewpoints without the need of detailed 2-D descriptions. This speeds up the region growing process significantly. To facilitate the region growing process, the three features are organized in the following fashion:

(1) the number of detected component contours -- m,

(2) a sorted list $(n_1, n_2, ..., n_m)$; where $n_1 \ge n_2 \ge \cdots \ge n_m$, m is the same as in (1) and n_i $(1 \le i \le m)$ is the number of edges (component segments) of an arbitrary component contour belonging to the 2-D projection under consideration,

(3) two ordered lists $((L_{11}, L_{21}, ..., L_{K1}), (L_{12}, L_{22}, ..., L_{K2}))$, where $L_{11} \ge L_{21} \ge \cdots \ge L_{K1}$, $L_{12} \ge L_{22} \ge \cdots \ge L_{K2}$, L_{k1} $(1 \le k \le K)$ represents a sorted sequence of the number of fully connected edges between all adjacent component contours, and L_{k2} $(1 \le k \le K)$ is another sorted sequence representing the number of partially connected edges between all the adjacent component contours.

Two descriptions output from the automated model construction program are shown in Figs.4(a)-(b). The extracted feature sets shown in Figs.4(a)-(b) are then converted into the forms shown in Figs.5(a)-(b), respectively. These forms are used in topological comparisons.

† We select $|CG_m|$ instead of RG_m since $|CG_m|$ is easier to detect.

2.3. Region Growing Process

The purpose of the region growing process is to partition the view sphere into a set of regions such that each region corresponds to a CV and the collection of all CVs forms the CV library. Here, we only give a brief description about this process. The details of this process can be found in [13]. The feature set extracted from each view is used as the criterion to partition the view sphere into maximally connected regions such that all the 2-D projections rendered from all viewpoints in a region have the same topology. Each partitioned region ultimately become a CV in the CV library. Each CV in the library contains a neighborhood list which registers all its neighboring CVs.

It is noted that the computational overhead is drastically reduced based on the proposed feature set to perform region growing. This is because a thorough description of a 2-D projection is not needed for detecting topological changes. Moreover, the space complexity is also reduced since we do not have to store anything at each viewpoint until all regions are fully grown. Fig.6 shows an instance of a partially parcelled view sphere based on the target object shown in Fig.1.

3. A COARSE-TO-FINE SEARCH PROCESS FOR OBJECT RECOGNITION

An automated CV library generation process has been presented in the previous section. However, the size of the library is still large if the target object is complex in shape. To speed up the recognition process, a coarse-to-fine search strategy is proposed. In the coarse search phase, an ART-1 network is used to locate a small set of CVs in the library most similar to the projection of the unknown object. Then, the fine search process starts with topological comparison between the scene description and the located CVs. The fine search process stops whenever a best match is found. The rest of this section is organized as follows. In Section 3.1, the ART-1 network is briefly reviewed. Then, the details of using ART-1 to perform coarse search is covered in Section 3.2. Finally, the fine search process is briefly described in Section 3.3.

3.1. The ART-1 Network

ART-1 is a neural network architecture that self-organizes stable categories in real time in response to arbitrary sequences of input patterns. Basically, it is a competitive learning model enhanced by a feedback process that leads to category stabilization. Like the rest of the competitive learning models, its basic purpose is to recognize or classify an input pattern to a category or cluster that is learned by the model previously. The architecture of ART-1 contains two layers of processing elements, F_1 and F_2, as shown in Fig.7. Layer F_1 receives the input pattern and a feedback signal from layer F_2, performs matching process, and activates the search process. Layer F_2 provides a competitive field for those learned categories, generates the most related category to the input pattern, and provides a feedback signal to F_1. Details of ART-1 architecture is described in [14]. Learning takes place when the input pattern and the feedback signal locks into a loop, i.e., they produce a resonant state. The behavior of an ART-1 network is described by a set of differential equations. While the system approaches a steady state, the signals and weights in the system have only two steady states. Hence, from the functional point-of-view, ART-1 performs clustering of binary vectors. As long as the input can be consistently transformed to a binary vector, many pattern recognition and classification problems can be efficiently and correctly solved by ART-1.

3.2. Using ART-1 for Coarse Search

The key step in this application is the transformation of a feature set to a binary feature vector that can be fed into an ART-1 network. Each entry in the feature set is an integer and different element in the feature set has different weighting factor. It is easy to code each element in the feature set using a string of binary digits. The difficult part is to take the weighting factor of each feature into consideration in forming the binary digits and then combine all the binary digits into a binary vector. Since the values of integers in the feature set are usually less than 10 in our objects of interest, an integer is represented by the same number of binary digits as its value. Several strings of binary digits that represent a whole feature set are arranged to form a 10 by 10 matrix. The top two most significant features are used to circular-shift the square matrix columnwise and rowwise, respectively.

According to the aforementioned feature representation scheme in Section 2.2.2, the elements in a feature set are ordered as m, $(n_1, n_2, \cdots, n_m)$, and $(L_{11}, L_{21}, \cdots, L_{k1})$, $(L_{12}, L_{22}, \cdots, L_{k2})$. Since m is the most important feature, 10 binary digits (referred to as the α group in the sequel) are used to represent it. n_i, $1 \le i \le m$, is represented by the number of binary digits (referred to as the β_i group in sequel) the same as $|n_i|$. Since $(L_{11}, L_{21}, \cdots, L_{k1})$ and $(L_{12}, L_{22}, \cdots, L_{k2})$ are relatively less important features, they are condensed to two respective numbers $S_1 = \sum_{i=1}^{K} |L_{i1}|$ and $S_2 = \sum_{i=1}^{K} |L_{i2}|$ and represented with binary strings (referred to as the γ_1 and γ_2 groups in the sequel). Digits from the α group are entered in the third column of a 10 by 10 matrix. Each β_i group is placed in a row starting from the first row and beginning with the third column. The γ_1 and γ_2 groups are entered into columns one and two of the matrix. After this basic matrix is formed, two most important features m and n_1 are used to circular-shift this matrix columnwise $|m|$ times and rowwise $|n_1|$ times, respectively. The operation to circular-shift a matrix columnwise once transforms the matrix $[C_1 C_2 \cdots C_p]$, where C_i's are column vectors, to another matrix $[C_p C_1 C_2 \cdots C_{p-1}]$. Similarly, the operation to circular-shift a matrix rowwise once transforms the matrix $[R_1 R_2 \cdots R_q]^T$, where R_i's are row vectors, to another matrix $[R_q R_1 R_2 \cdots R_{q-1}]^T$, where $[\cdot]^T$ represents the transpose of matrix $[\cdot]$. The resultant matrix can be transformed to a binary vector and used as the input to the ART-1 network. The original matrices and the shifted matrices of the two examples in Fig.5 are shown in Figs.8(a),(c) and (b),(d), respectively. This scheme takes the weighting factors into consideration in forming the binary vector and thus is able to distinguish different CVs. Of course, our scheme is just one of the feasible transformations that can detect the difference between features.

ART-1 is applied to perform the coarse search process using the transformed binary vectors. In the training phase, the feature set of each CV in the CV library is transformed to a binary vector and fed into ART-1. In the testing phase, a feature set is extracted from an unknown view in an image. This feature set is then transformed and fed into ART-1 in order to select a set of CVs that are most similar to the projection in the image. Those located CVs are then used as the starting points to perform fine search. The coarse search process using ART-1 reduces the search space to a significantly small size. This will be further elaborated in Section 4.

3.3. Fine Search Process

The fine search process starts with those CVs located in the coarse search process and compares the topology of the (object) projection with all the views in the neighborhood of the located CVs until a best match is found. In the fine search process, the operation the topological comparison is unavoidable. A number of algorithms such as *graph distance* measure [19,20] and *relational distance* measure [21,22] have been proposed for topological comparison. Basically, the topological comparison is performed between two descriptions derived separately from two 2-D projections. This part is beyond the scope of this paper and is currently under investigation.

4. EXPERIMENTAL RESULTS

In our experiments, we used the three polyhedral objects shown in Fig.9 to test the proposed coarse search process. An ART-1 network is first trained by using a set of binary vectors from all the CVs in the library for several iterations until it reaches the state of stable categorization. This is when the system does not generate new categories and every input from the library always belongs to an category generated by the system in the previous iterations. When the test samples are drawn from one of the 320 viewpoints on the view sphere, the successful rate is 100 percent. Here it reflects the property of direct access when a familiar pattern is input and the category that it should belong to is "resonant" instantly without search in memory. When the samples are selected from the neighborhood of the 320 viewpoints, the successful rate drops to 90 percent. This is due to the training viewpoints don't cover the whole view sphere. There are some peculiar feature sets in some view angles that are not in the training set and result in error classification.

In Figs.10(a) and 11(a), two randomly generated synthetic images are fed into a trained ART-1 network. The located CVs which correspond to each test image are shown in Figs.10(b)-(f) and 11(b), respectively. In another set of experiments, we used the two range images shown in Figs.12(a) and 13(a). After applying the image segmentation algorithm proposed in [4], the results are shown in Figs.12(b) and 13(b), respectively. The most similar CVs located by the ART-1 network are shown in Figs.12(c)-(d) and 13(c)-(f), respectively. Figs.14(a) and (b) show the actual locations of these located CVs on the view sphere for the range images in Figs.12(a) and 13(a), respectively.

5. CONCLUDING REMARKS

In this paper, we have presented a method to reduce the search space for 3-D object recognition using an ART-1 neural network. In the modeling phase, a method to automatically generate the CV library of a polyhedral object is presented. The CV library generation process is an important step in multiple-view approach to 3-D object recognition. The task includes not only generating the minimum aspect graph but also describing each node (line drawings of a 2-D projection) and arc (the transformation between a pair of nodes) in a form suitable for subsequent matching process. The automated CV library generating system described in this paper is able to handle all kinds of polyhedral objects.

Although the redundancy has been reduced in the CV library generation process, the size of the library (or aspect graph) is still large if the target object is complex in shape.

To select a good starting point for searching the most similar view in the aspect graph is crucial to an efficient recognition algorithm. A method which is capable of performing this task to reduce the search space using an ART-1 neural network has been proposed. It is used in the coarse search phase of the object recognition process to locate a set of CVs most similar to the projection of the object. There are some advantages of using ART-1 in the multiple-view approach. Firstly, it is a universal clustering scheme which does not count on the shapes of the target objects. Secondly, it achieves stable categorization under arbitrary learning environment. This is a favorable property in the multiple-view approach to 3-D object recognition since the number of viewpoints needed to construct a CV library can be further increased whenever needed. Finally, ART-1 can directly access a learned category of a familiar input pattern no matter how large and complex the learned categories are. This kind of search speed is desirable in all applications. In the coarse-to-fine search process, the problem of finding a transformation or coding method to achieve maximum distance in discriminating categories or clusters with minimal number of digits is still under investigation. The possibility of using ART-1 in the fine search process is also under study.

REFERENCES

[1] P. J. Besl and R. C. Jain, "Three-Dimensional Object Recognition," *ACM Computing Surveys* , Vol.17, No.1, pp.75-145, Mar. 1985.

[2] M. Oshima and Y. Shirai, "Object Recognition Using Three-Dimensional Information," *IEEE Trans. on Pattern Analysis and Machine Intelligence* , Vol.PAMI-5, No.4, pp.353-361, July 1983.

[3] R. T. Chin and C. R. Dyer, "Model-Based Recognition in Robot Vision," *ACM Computing Surveys* , Vol.18, No.1, pp.67-108, Mar. 1986.

[4] P. J. Besl and R. C. Jain, "Segmentation Through Variable-Order Surface Fitting," *IEEE Trans. on Pattern Analysis and Machine Intelligence* , Vol.10, No.2, pp.167-192, March 1988.

[5] R. Hoffman and A. K. Jain, "Segmentation and Classification of Range Images," *IEEE Trans. on Pattern Analysis and Machine Intelligence* , Vol.9, No.5, pp.608-620, September 1987.

[6] W. N. Martin and J. K. Aggarwal, "Occluding Contours in Dynamic scenes," *Proc. Conf. Pattern Recognition Image Processing* , pp.189-192, Aug. 1981.

[7] W. N. Martin and J. K. Aggarwal, "Volumetric Description of Objects from Multiple views," *IEEE Trans on Pattern Analysis and Machine Intelligence* , Vol.PAMI-5, pp.150-158, Mar. 1983.

[8] H. G. Barrow and J. M. Tenenbaum, "Computational Vision," *Proceedings of IEEE* , Vol.69, No.5, pp.572-595, May 1981.

[9] M. R. Korn and C. R. Dyer, "3-D Multiview Object Representations for Model-Based Object Recognition," *Pattern Recognition* , Vol.20, No.1, pp.91-103, 1987.

[10] I. Chakravarty and H. Freeman, "Characteristic Views as a Basis for Three-Dimensional Object Recognition," *Proc. Soc. Photo-Opt. Instr. Engrs*, 336, pp.37-45, 1982.

[11] H. Freeman and I. Chakravarty, "The Use of Characteristic Views in the Recognition of Three-Dimensional Objects," *Pattern Recognition in Practice*, E. Gelsema and L. Kanal, Editors, North-Holland Publishing Company, Amsterdam, pp.277-288, 1980.

[12] I. Chakravarty, "The Use of Characteristic Views as a Basis for Recognition of Three-Dimensional Objects," *Ph.D. Dissertation*, Dept. of Computer and Systems Engineering, Rensselaer Polytechnic Institute, 1982.

[13] F.-Y. Liao, "Multiple-View Approach to 3-D Object Recognition," Ph.D. dissertation, Dept. of Electrical Engineering and Computer Science, Northwestern University, June, 1990.

[14] G. A. Carpenter and S. Grossberg, "A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine," *Computer Vision, Graphics, and Image Processing*, 37, pp.54-115, 1987.

[15] G. A. Carpenter and S. Grossberg, "The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network," *Computer*, pp.77-88, 1988.

[16] R. L. Hoffman, "Object Recognition from Range Images," *Ph.D. dissertation*, Michigan State University, Dept. of Computer Science, 1986.

[17] C. K. Cowan and P. D. Kovesi, "Automatic Sensor Placement from Vision Task Requirements," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.10, No.3, pp.407-416, May 1988.

[18] L. G. Shapiro and R. M. Haralick, "Structural description and inexact matching," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-3(5), pp.504-515, 1981.

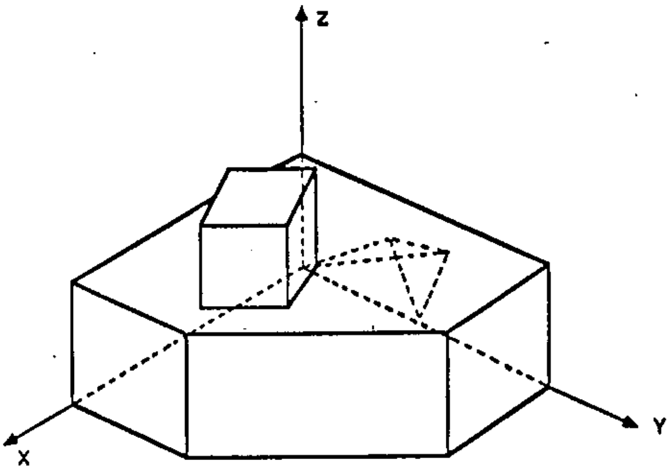


Fig.1 A polyhedral object.

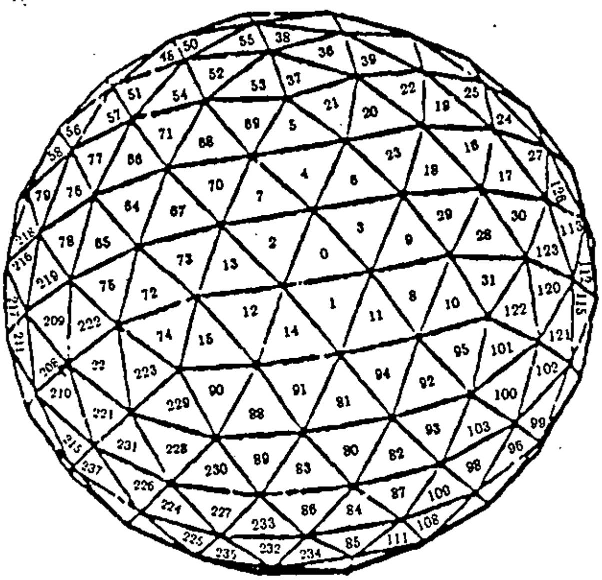


Fig.2 The view sphere after 320 facets of the icosahedron are projected onto it.

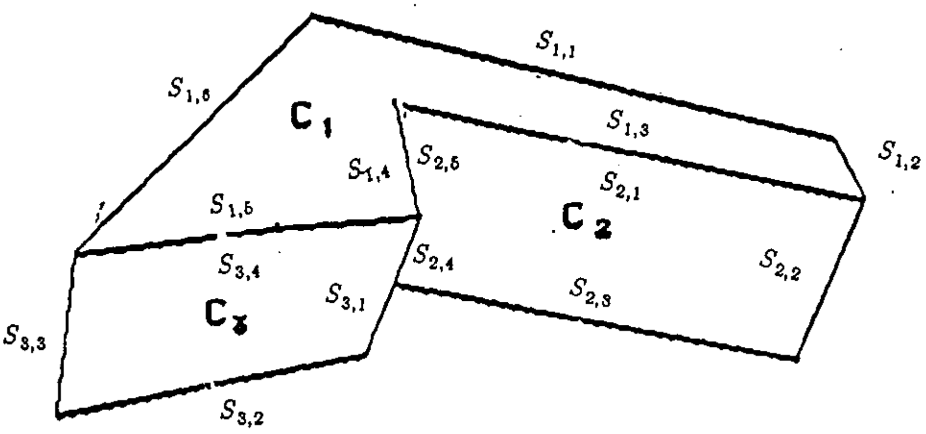
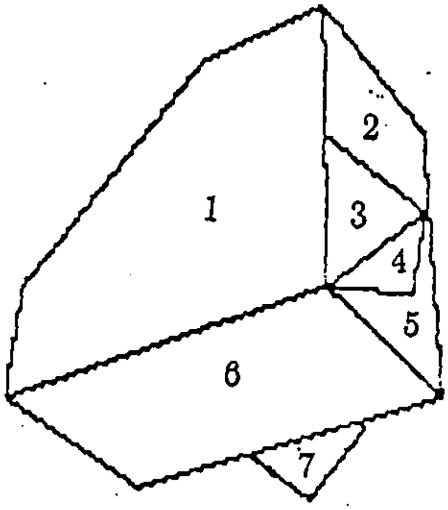
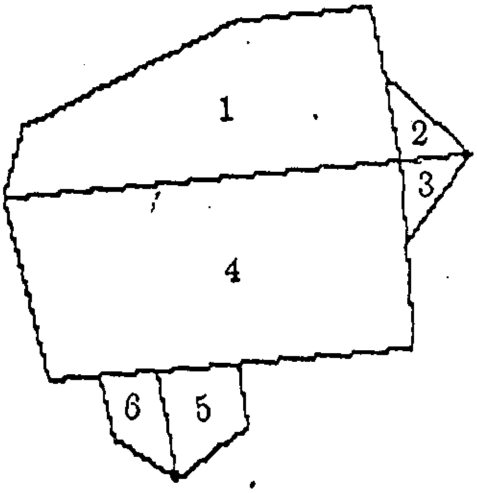


Fig.3 A 2-D projection of an L-shaped object.



(a)

process viewpoint no. 273
5 edges in region 1
4 edges in region 2
3 edges in region 3
3 edges in region 4
4 edges in region 5
4 edges in region 6
3 edges in region 7
region pair 2 & 1 has 1 partial con.
region pair 2 & 1 has 0 full con.
region pair 3 & 1 has 1 partial con.
region pair 3 & 1 has 0 full con.
region pair 6 & 1 has 0 partial con.
region pair 6 & 1 has 1 full con.
region pair 3 & 2 has 0 partial con.
region pair 3 & 2 has 1 full con.
region pair 4 & 3 has 0 partial con.
region pair 4 & 3 has 1 full con.
region pair 5 & 4 has 0 partial con.
region pair 5 & 4 has 2 full con.
region pair 6 & 5 has 0 partial con.
region pair 6 & 5 has 1 full con.
region pair 7 & 6 has 1 partial con.
region pair 7 & 6 has 0 full con.



(b)

process viewpoint no. 287
4 edges in region 1
3 edges in region 2
3 edges in region 3
4 edges in region 4
4 edges in region 5
4 edges in region 6
region pair 2 & 1 has 1 partial con.
region pair 2 & 1 has 0 full con.
region pair 4 & 1 has 0 partial con.
region pair 4 & 1 has 1 full con.
region pair 3 & 2 has 0 partial con.
region pair 3 & 2 has 1 full con.
region pair 4 & 3 has 1 partial con.
region pair 4 & 3 has 0 full con.
region pair 5 & 4 has 1 partial con.
region pair 5 & 4 has 0 full con.
region pair 6 & 4 has 1 partial con.
region pair 6 & 4 has 0 full con.
region pair 6 & 5 has 0 partial con.
region pair 6 & 5 has 1 full con.

Fig.4 Two examples to illustrate the representation scheme for 2-D projections.

viewpoint= 273, no. of regions= 7
--- sorted edge list ---
5
4
4
4
3
3
3
--- sorted full_conn. list ---
2
1
1
1
1
0
0
0
--- sorted partial_conn. list ---
1
1
1
0
0
0
0
0

(a)

viewpoint= 287, no. of regions= 6
--- sorted edge list ---
4
4
4
4
3
3
--- sorted full_conn. list ---
1
1
1
0
0
0
0
--- sorted partial_conn. list ---
1
1
1
1
0
0
0

(b)

Fig.5 Data structures of the two representations in Fig.4.

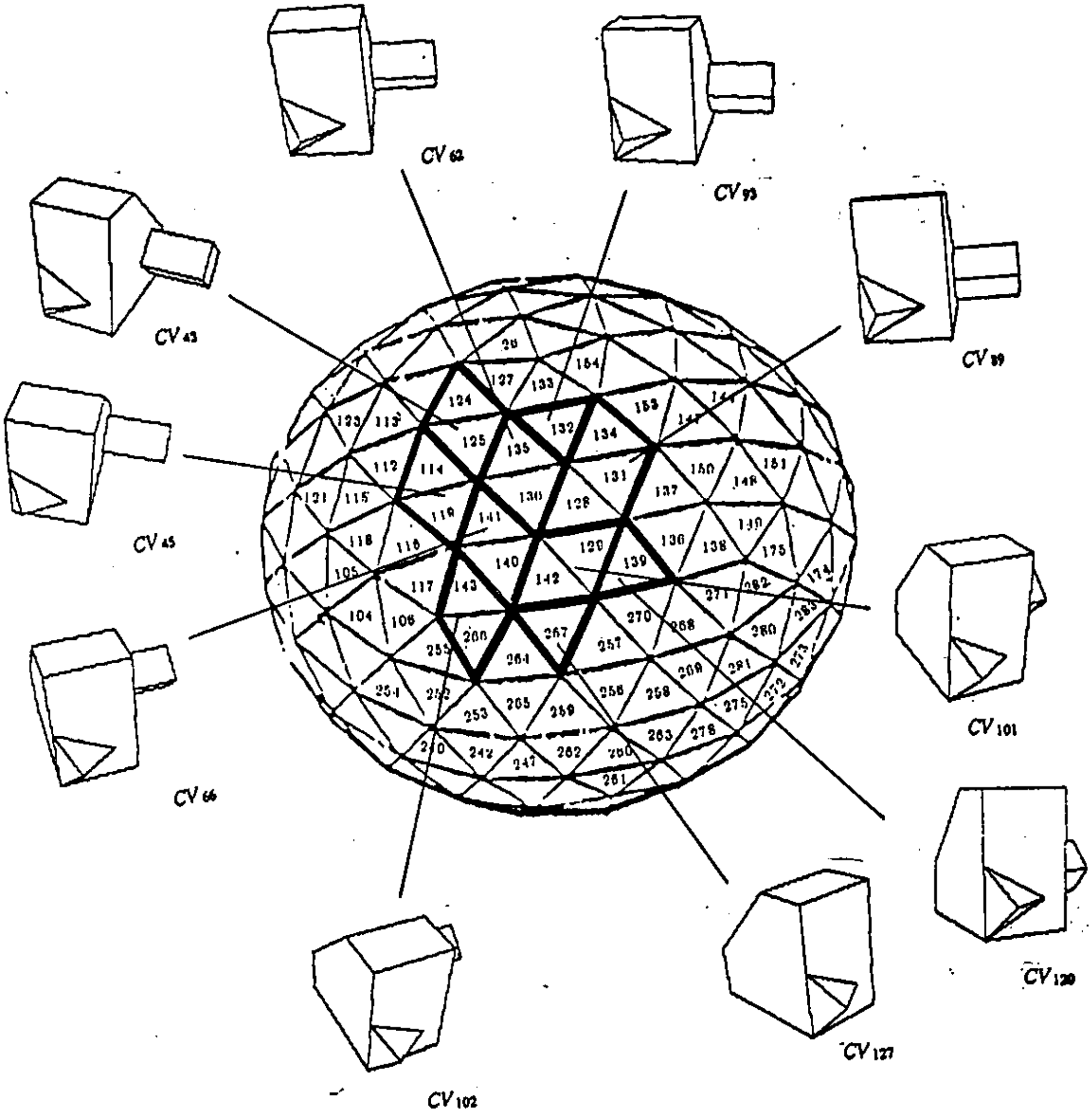


Fig.6 An instance of the partially parcelled view sphere with each CV bounded by bold lines.

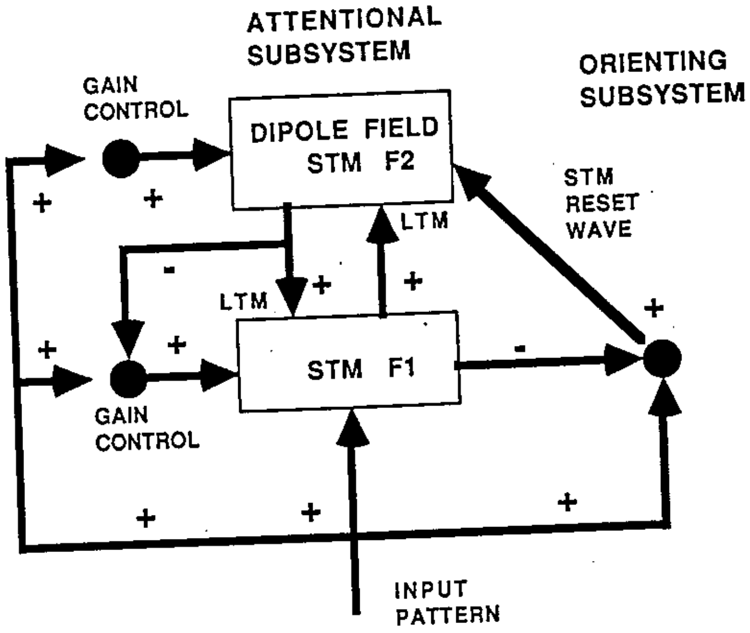


Fig.7 The architecture of ART-1 system (redrawn based on Fig.3 of [8]).

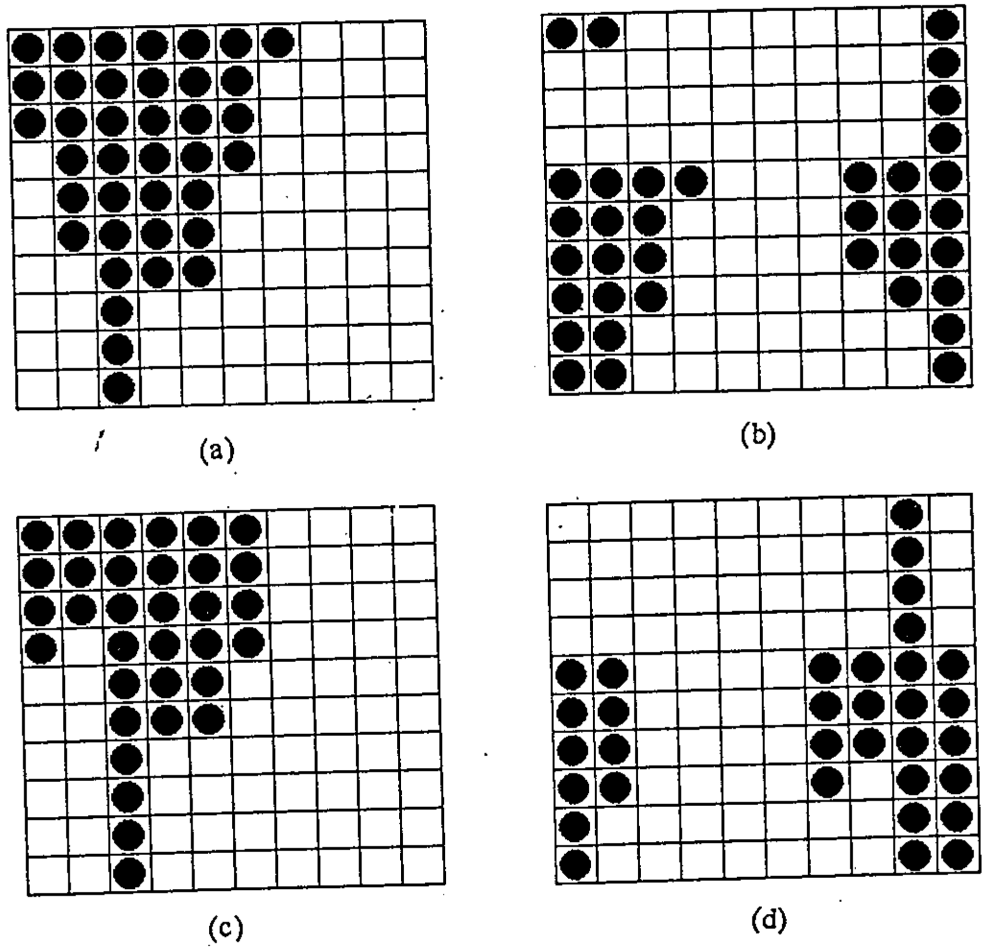


Fig.8 (a) and (b) are the original and the shifted matrices of the example in Fig.5(a). (c) and (d) are the original and the shifted matrices

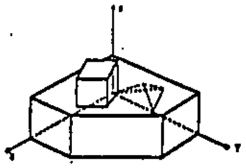
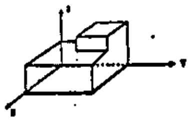
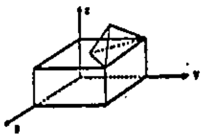
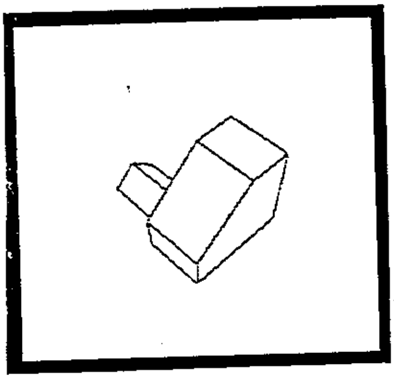
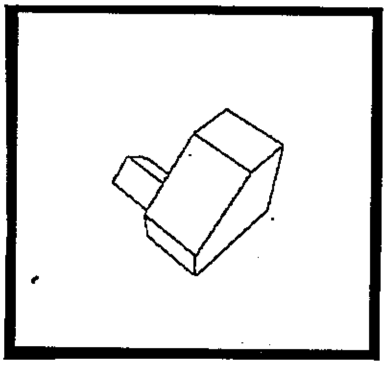
Target Objects	2-D projections	Regions found on the view sphere after region growing process
1		176
2		154
3		116

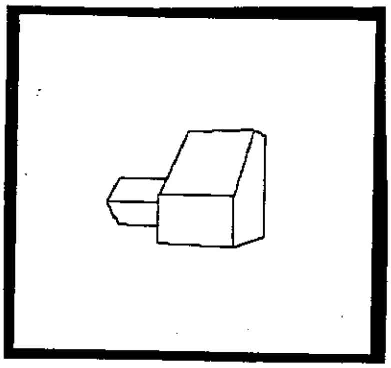
Fig.9 Different sizes of CV libraries generated from three different polyhedral objects.



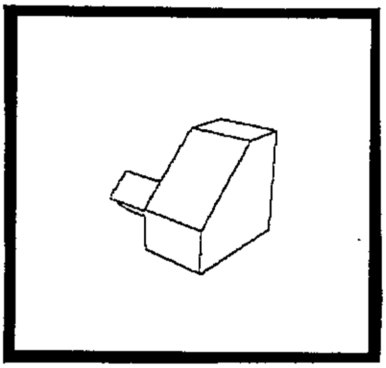
(a) A synthetic image.



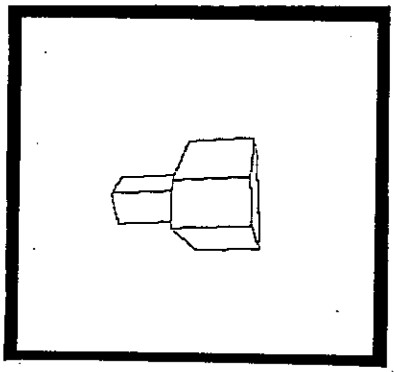
(b) Viewpoint 111 (CV 65)



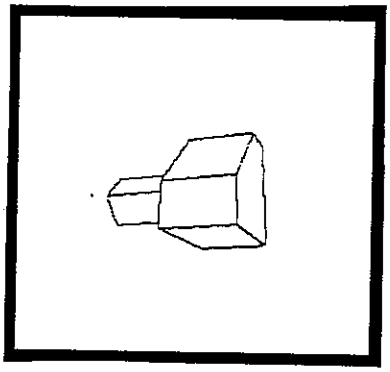
(c) Viewpoint 235 (CV 85)



(d) Viewpoint 318 (CV 86)

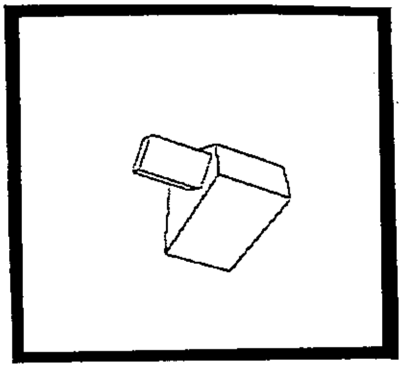


(e) Viewpoint 212 (CV 89)

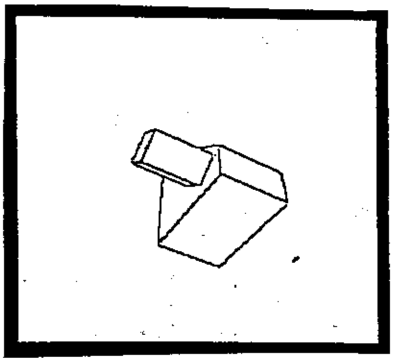


(f) Viewpoint 213 (CV 89)

Fig.10 (a) A synthetic image generated from an arbitrary viewpoint,
(b) to (f) are some views in the CV library that are considered by the ART-1 network to be most similar to (a).

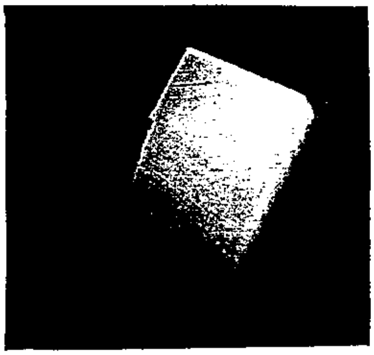


(a) A synthetic image.

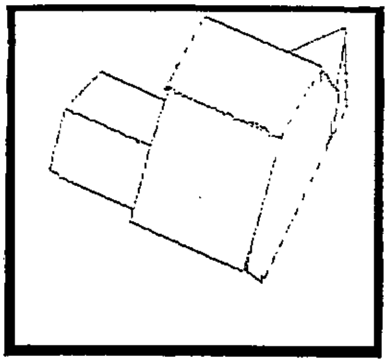


(b) Viewpoint 197 (CV 88)

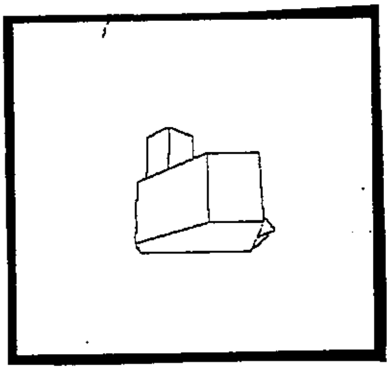
Fig.11 (a) A synthetic image generated from an arbitrary viewpoint, (b) A located CV that is most similar to (a).



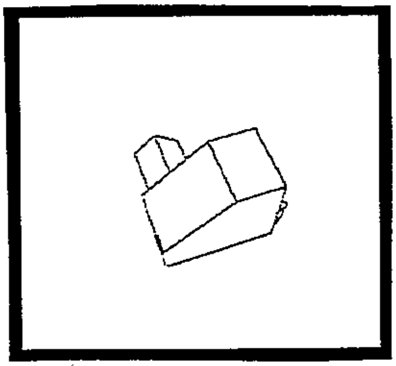
(a)



(b)

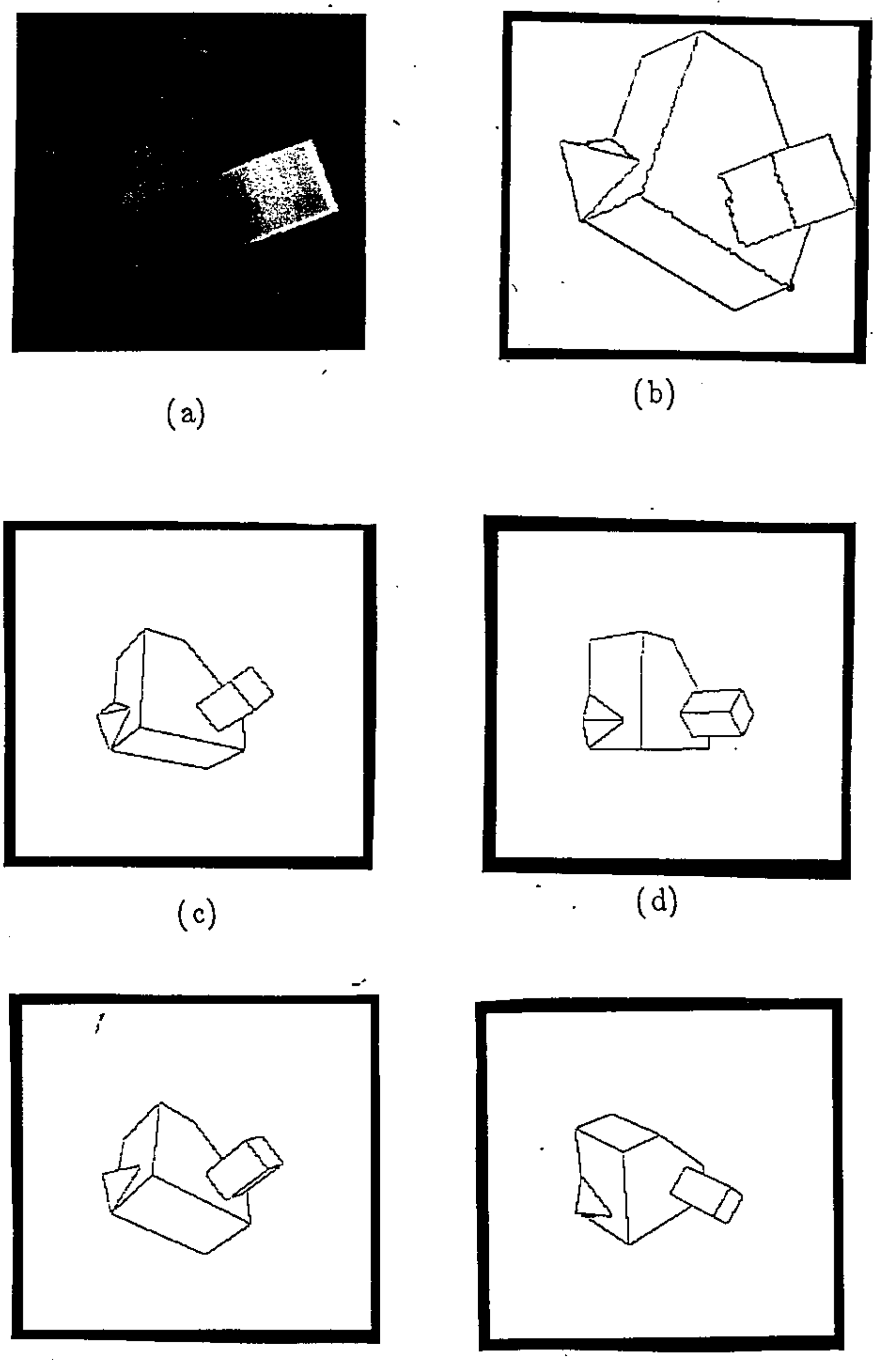


(c)



(d)

Fig.12 (a) A range image, (b) result after image segmentation, (c) and (d) are the most similar CVs located by the ART-1 network.

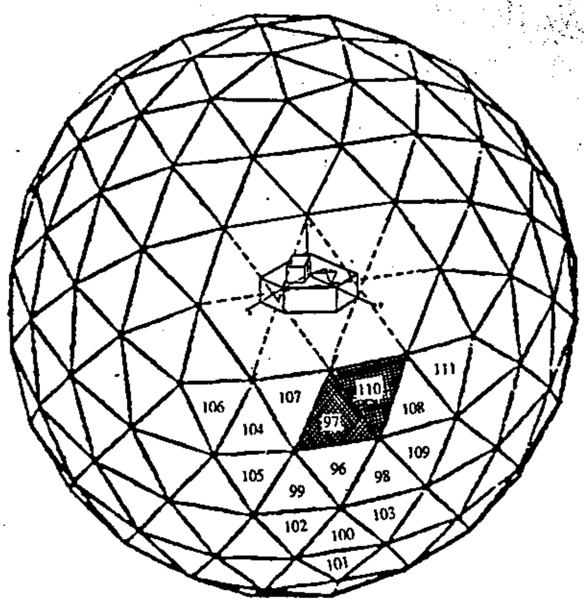


(a) (b)

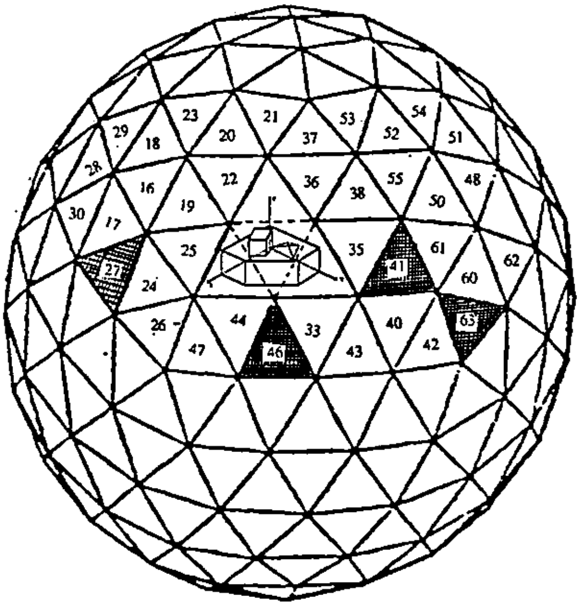
(c) (d)

(e) (f)

Fig.13 (a) A range image, (b) result after image segmentation, (c) to (f) are the most similar CVs located by the ART-1 network.



(a)



(b)

Fig.14 (a) The actual locations of the candidate CVs on the view sphere for the projection shown in Fig.12(a).
(b) The actual locations of the candidate CVs on the view sphere for the projection shown in Fig.13(a).