A Monocular System for 3-D Object Recognition and Pose Determination

A. K. C. Wong, L. Rong and X. Liang
Systems Design Engineering
University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1
e-mail: akcwong@watnow.uwaterloo.ca
lrong@watnow.uwaterloo.ca
xliang@watnow.uwaterloo.ca

Abstract

A challenge in computer vision is to recognize 3-D objects and determine their 3-D pose from images using the known models of the objects. In this paper, a methodology, based on a single image acquired from a CCD camera, is presented for such a task. From intensity images, the salient features are first accurately and robustly detected by an algorithm. From the detected features, elliptical curves and straight lines are then grouped into feature groupings according to their feature types as well as their spatial, geometrical and topological relations. Through the correspondences between the feature groupings and the model features, the object's 3-D pose is determined from which their CAD models are projected back to the image for confirmation. Results from implementations are presented to demonstrate the efficacy of the methodology for recognizing polyhedral and cylindrical objects.

1 Introduction

In computer vision, 3-D object recognition and pose determination are closely related in both theory and implementation. In model-based cases, the essential task is to find through sensors the identity and the pose of 3-D objects from object models in the model base. In general, there are two common types of sensory inputs: 2-D intensity images obtained from CCD cameras, and range images obtained from range scanners [12]. Problems involving data of the first kind are generally more difficult, especially when only image data from a single CCD camera are available. This is known as the "monocular vision" problem and this type of problem is particularly important for vision based industrial automation [13].

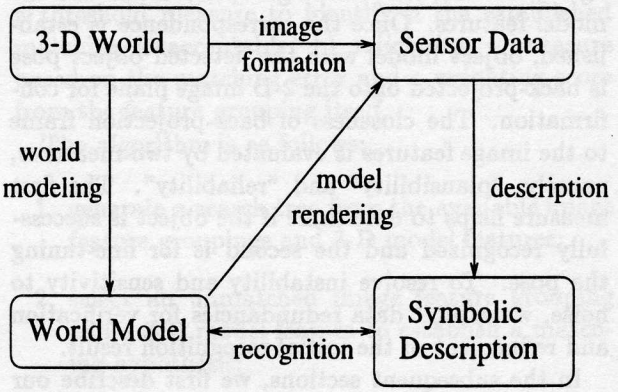


Figure 1: The general structure of an object recognition system (based on Besl and Jain 85)

The structure of a general object recognition system is shown in Figure 1 [1]. Among the five processes described here, the description (feature extraction and grouping) and the recognition process are the major issues to be presented in this paper.

Feature extraction is a process for finding distinct primitive characteristics or attributes in the image. In general, image features are classified into two major types: region-based and edge-based. Region-based features are derived from the estimation of statistical parameters. This is usually time consuming. Edge-based features, such as edges, lines and corners are more frequently used [10, 9, 6]. Classical approaches to detect edge-based features make use of lines and corners only. The close relations among these features are not utilized in the recognition process. Object recognition usually relies on feature matching assisted by pose determination.

In this paper, we present a vision system for object recognition and pose determination based on a single image acquired from a CCD camera. It first detects salient features from grey-level image and then groups them according to their feature types as well as their spatial, geometrical and topological relations. The types of feature groupings include: (1) four corner points and the triplets of lines forming corners; (2) curve segments that fit ellipses with estimated parameters. The use of matching hypotheses generated based on feature groupings is usually more robust and effective than the combinatorial matching of point features.

The object recognition and pose estimation problem is based upon Fischler and Bolles' algorithm [5]. It determines the 3-D object pose from a 2-D image by relating the feature groupings found and the model features. Once the correspondence is established, object model with the detected object pose is back-projected onto the 2-D image plane for confirmation. The closeness of back-projection frame to the image features is evaluated by two measures, namely, "plausibility" and "reliability". The first measure helps to determine if the object is successfully recognized and the second is for fine-tuning the pose. To resolve instability and sensitivity to noise, we exploit data redundancies for verification and refinement of the initial recognition result.

In the subsequent sections, we first describe our feature detection, grouping and pose determination algorithms. We then present results from experiments to demonstrate the efficacy of the methodology for recognizing polyhedral and cylindrical objects.

2 Algorithm Description

Figure 2 provides the functional modules of the system. In regards to the processing level of information, the system is subdivided into three major blocks: image pre-processing, feature grouping and object recognition. In this paper, we will focus on the last two blocks.

2.1 Visual Feature Grouping

An intelligent interpretation of 2-D image begins with edge features [14]. Based on the edge features, feature grouping clusters the individual 2-D features in an image to one or more groups each of which interprets a possible part of the object to be recognized. Feature grouping will effectively reduce the search space of object recognition and pose determination, and therefore, speed up the entire process.

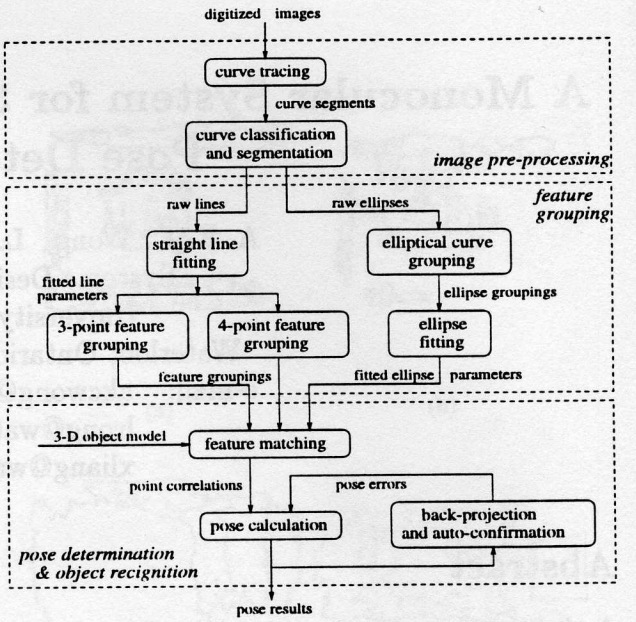


Figure 2: The software modules of monocular 3-D object recognition and pose determination system

It changes the low level 2-D image features into high level symbolic features.

The task of feature grouping can be formalized as the following: Given a set of 2-D image features F, the feature grouping procedure C outputs a set of possible feature groups $\{F_i\}$ such that each group is related to a part of an object imaged in the 2-D image, i.e.

$$C: \quad F \rightarrow \{F_1, \cdots, F_n\} \qquad (1)$$
$$s.t. \quad \bigcup_{i=1}^{n} F_i = F \quad and$$
$$F_i \Rightarrow \{obj_j\}$$

where the set of obj_j is in the model base. In this article, we restrict the element of F as one of the following two symbolic features: a) straight lines and b) elliptical and circular curves.

The procedure of feature grouping is essentially a heuristic process which searches for connected local features. The 2-D geometrical and topological constraints of feature combination reduce the number of possible groupings effectively.

2.1.1 Junctions and Corners

Through edge tracking, we obtain the traces of edges in a 2-D image. The traces consist of straight lines, curve segments and other features. As shown in Figure 3, straight lines can be separated from the edge traces and grouped together under grouping

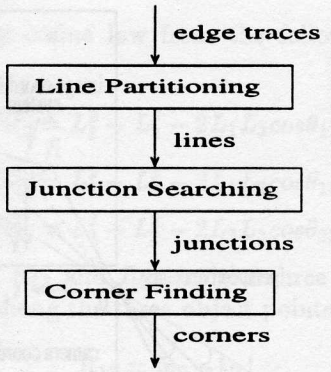


Figure 3: Straight line grouping procedure

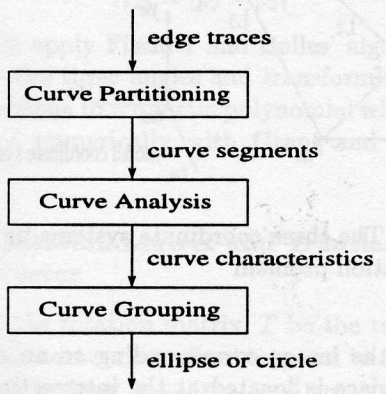


Figure 4: Curve grouping procedure

rules. Then junctions can be found in each straight line group. The junctions can also be grouped together. From each junction group, we can find corner groupings.

2.1.2 Ellipses and Circles

Figure 4 illustrates the curve grouping procedure. Just like the lines, curves can be partitioned from the edge traces. The curve segments are combined into groups and the equation of each group is then solved after the grouping process to determine the curve parameters. The equation for ellipse is expressed as:

$$\frac{[(x-a)\cos\theta+(y-b)\sin\theta]^2}{c^2}+\frac{[-(x-a)\sin\theta+(y-b)\cos\theta]^2}{d^2}=1 \quad (2)$$

where (a, b) is the center, c and d ($c > 0$ and $d > 0$) are the axes, θ is the rotation angle (direction) of the ellipse. When $c = d$, the ellipse becomes a circle.

In order to estimate the five parameters of an ellipse, at least five 2-D image points on each curve grouping are needed.

2.2 Hypothesis Directed Feature Matching

In this paper, recognizing an object from a 2-D image is essentially a feature matching process. It attempts to establish correspondence between feature groupings from 2-D images and the 3-D features from models in the object model base. In general, feature matching requires exhaustive search and is recognized as an NP-complete or NP-hard problem [4].

To speed up the matching, we introduce a hypothesis directed search based on a modified version of the constellation matching algorithm [11]. Hypothesis generated from observed feature groupings are used as heuristics. The hypothesis is evaluated with the following two criteria: a) *plausibility*: a threshold measure to identify if the established matching is acceptable; b) *reliability*: a measure based on the matching error and a weighting score from the feature grouping itself.

The algorithm is as follows:

1. generate a search tree from the available image feature groupings and 3-D model features;
2. select an unmatched image feature grouping and a 3-D model feature to establish a matching hypothesis;
3. estimate the characteristic view [2, 14] (a 2-D description of the 3-D object based on on a perspective projection) according to the hypothesis;
4. evaluate the plausibility and reliability of the matching hypothesis;
5. verify the hypothesis and record the successful matching;
6. go to step 2 to search for new matching pairs until all image feature groupings has been exploited or an plausible and reliable matching has been found;
7. output the result.

2.3 Pose Determination

Pose determination is a well studied problem. Algorithms employing analytic closed form solutions as well as numerical solutions can be found in [3, 5, 7, 8]. Based on these work, we have developed a pose determination package to resolve the problems of speed, accuracy and uncertainty.

2.3.1 The Pose Estimation Problem

The pose estimation problem is described as follows: Let $(p_1, p_2, ..., p_n)$ be the points of the observed object expressed in a coordinate system pre-assigned to the object, and $(P_1, P_2, ..., P_n)$ be the corresponding perspective projection points on the image plane. A pose determination algorithm would yield a rotation matrix R and a translation vector T which together map $(p_1, p_2, ..., p_n)$ onto $(P_1, P_2, ..., P_n)$. Apparently, a corresponding inverse rotation matrix and translation vector should map the image points back into model space. As a result, the determined model points can be expressed relative to the image plane or, as described later, to the camera coordinate system. We assume that both the principal point in the image plane (where the optical axis of the lens intersects with the image plane) and the focal length (distance from the center of the perspective to the principal point) of the imaging system are known. We also assume that the camera resides outside and above and object, and a convex hull would enclose the control points.

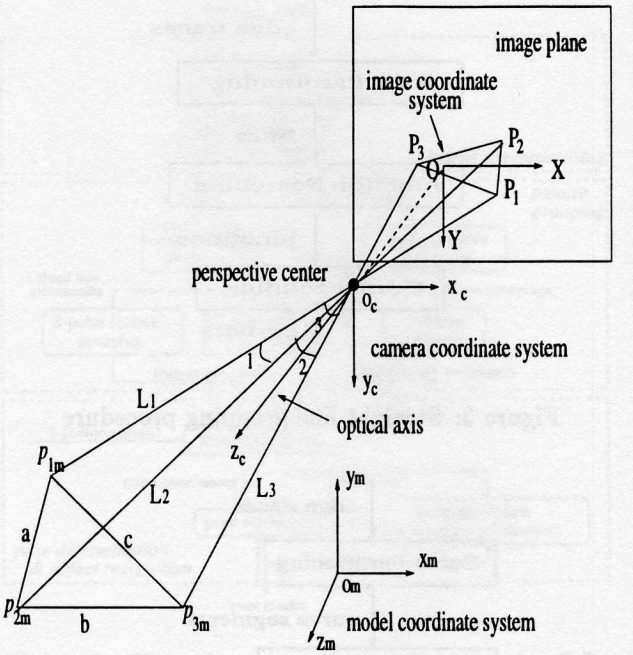


Figure 5: The three coordinate systems in the pose determination problem

2.3.2 Geometry, Projections and Transformations

In pose estimation, three coordinate systems, namely, image, camera and model system are adopted (Figure 5). In the following, we use the lower case p's and upper case P's to refer to the points in the 3-D space and on the 2-D plane respectively. The subscripts c and m signify that the point is expressed in the camera or model coordinate systems respectively.

Image Coordinate System

The image coordinate system is defined on the 2-D image plane. Following the convention in most pose estimation algorithms, we put the origin at the geometric center of the image (as opposed to the top-left corner in most of other image processing systems) with a horizontal x axis and a vertical y axis. To keep the image in a positive orientation we assume that the image plane is perpendicular to the viewing axis and located at a distance f (the lens focal length) from the camera center.

Camera Coordinate System

The camera coordinate system is a viewer centered reference system such that the observer is located at the origin which is the center of the lens. The viewing axis is colinear to the z axis. For simplicity, the x and y axes are chosen to be colinear to the x and y axes of the image coordinate system respectively.

In the perspective model as shown in Figure 5, the point on the image corresponding to an arbitrary point in space is located at the intersection of the image plane and the line joining the point to the center of the lens. Thus, if a point has coordinates $(x, y, z)_c$ in the camera system, its image point P will be $(X, Y, f)_c = (\frac{xf}{z}, \frac{yf}{z}, f)_c$ where f is the focal length.

Model Coordinate System

The model coordinate system is used to describe the CAD model of known objects. It is usually selected such that the origin is located at the center or a selected corner of one of the object's flat surfaces while the z axis is perpendicular to the plane.

2.3.3 The Perspective Tetrahedron

Suppose that we have three image points P_1, P_2, P_3 and their 3-D counterparts p_1, p_2, p_3. According to the perspective projection, the positions of p_1, p_2 and p_3 are determined from the image rays through the corresponding image feature points P_1, P_2 and P_3.

The constraints imposed by the relative 3-D position of the object points in both model and camera coordinate system could be expressed by the set of three distances ("legs") from p_1, p_2 and p_3 to the camera len's center. The three legs, also called the perspective tetrahedron, are denoted by L_1, L_2 and L_3 (Figure 5).

A solution set of L_1, L_2 and L_2 can be obtained

by applying cosine law from the following set of functions:

$$R_{12}^2 = L_1^2 + L_2^2 - 2L_1L_2cos\theta_{12} \quad (3)$$

$$R_{13}^2 = L_1^2 + L_3^2 - 2L_1L_3cos\theta_{13} \quad (4)$$

$$R_{23}^2 = L_2^2 + L_3^2 - 2L_2L_3cos\theta_{23} \quad (5)$$

where R_{12}, R_{13} and R_{23} are the three inter-point distances among the three object points:

$$R_{12} = |p_2 - p_1|$$

$$R_{13} = |p_3 - p_1|$$

$$R_{23} = |p_2 - p_3|$$

Here we apply Fischler and Bolles' algorithm [5] to obtain the three angles and transform the above three functions to a quartic polynomial whose roots are solved numerically with Grant and Hitchin's method.

2.3.4 Rotation Matrix and Translation Vector

Let R be the rotation matrix, T be the translation vector from camera coordinate system to the model coordinate system; p_{1m}, p_{2m} and p_{3m} be the coordinates of the three object points in the model coordinate system; and p_{1c}, p_{2c} and p_{3c} be the coordinates of the three object points in the camera coordinate system. As suggested in Section 2.3.2, p_{1c}, p_{2c} and p_{3c} can be determined from their locations in the image coordinate system. We then have:

$$p_{im} = Rp_{ic} + T, \qquad i = 1, 2, 3 \quad (6)$$

With the known locations of p_{1m}, p_{2m} and p_{3m} (in the model coordinate system) and L_1, L_2 and L_3 from the solution, the translation vector T, which is actually the coordinate of the camera len's center relative to the model's origin, can be easily solved by a set of three quadratic equations. Then we can transform Equation 6 into:

$$p_{im} - T = Rp_{ic} \qquad i = 1, 2, 3 \quad (7)$$

Now we combine the three point vectors into a matrix and obtain:

$$\bar{p}_m = (p_{1m}, p_{2m}, p_{3m})$$

$$\bar{p}_c = (p_{1c}, p_{2c}, p_{3c})$$

Equation 7 then becomes:

$$\bar{p}_m - T = R\bar{p}_c \quad (8)$$

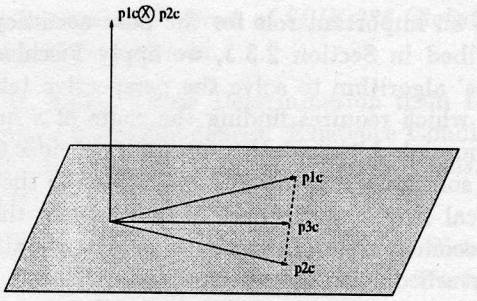


Figure 6: The solution of three colinear points

We then can solve R by:

$$R = \bar{p}_m\bar{p}_c^{-1} \quad (9)$$

Note that we have to find the inverse of a matrix $\bar{p}_c$. Thus R cannot be figured out while $\bar{p}_c$ is singular. This does happen when the three vectors are on a common plane. In this case, we could still calculate R by two of vectors and the norm vector of the plane. As seen in Figure 6, we choose two vectors containing the farthest separated image points and their cross product to construct $\bar{p}_c$. $\bar{p}_c$ will not be singular since the two vectors and their cross product will not be coplanar unless one of the vectors is zero.

2.4 Pose Verification and Refinement

The above pose estimation algorithm yields several solutions. The reliability and plausibility measures are then used as criteria for pose verification and refinement. Given one of the pose solutions R and T (called a "candidate"), it is straightforward to obtain the inverse transformation R' and T' which are then used to project the entire object model from the model coordinate system to the 2-D image. In the image coordinate system, the back-projection error is then calculated based on the discrepancy between the 2-D image features and the corresponding projected 3-D model features. The back-projection error, the number of matched feature groupings and their pre-calculated weighting scores are then combined into a plausibility criterion. The candidate with the largest plausibility is selected as the best solution. If the measure of the best solution is still smaller than a pre-set threshold, we conclude that the object recognition fails.

There are a number of factors, such as image noises and the distortion of the camera lens, which may cause inaccuracies in pose estimation. It is surprising to find that the calculation of rounding error

plays an important role for the pose accuracy. As described in Section 2.3.3, we apply Fischler and Bolles' algorithm to solve the perspective tetrahedron which requires finding the roots of a quartic polynomial. Although the approach provides closed form solutions, it is inevitably sensitive to the ratio of focal length and object distance when this ratio becomes large. In fact, the quartic equation is a perturbed form of equation $(x-1)^4 = 0$. It is well known that this equation may be very sensitive to the perturbation and stable while accurate roots are hard to obtain. The Grant and Hitchin's method always return a set of roots but in some cases the equation is ill-conditioned and there is no guarantee on their accuracy.

To enhance the reliability of the pose estimation, we introduce a pose refinement process to "fine-tune" the solutions using the redundant information. In the algorithm described in Section 2.3.3, R and T are calculated by three pairs of matching points. When extra matching feature pairs and ellipse/circle pairs are available (true in most cases), they will be used only for evaluating the matching errors and the plausibility criterion. In pose refinement, among all the matching pairs, a combination of four pairs with the least uncertainty and largest plausibility are selected for pose re-calculation. The new "fine-tuned" pose result would in general agree with the initial one but with higher accuracy and with ill-condition problem avoided.

3 Applications and Results

3.1 System Configuration

The monocular 3-D object recognition system was implemented in the PAMI lab at the University of Waterloo to support intelligent robotics for automatic object manipulation and collision avoidance. It is part of the STEAR #3 research project sponsored by Thomson-CSF. As shown in Figure 7, the system consists of several software packages implemented on Sun Sparc 330 and a PUMA robot arm mounted with a CCD camera. The task of the robot is to move into a frame, engage known objects, one at a time, and transport them out of the frame. The role of the vision system is to provide the robot information on the recognized object and the frame as well as their pose. Through an *eye-hand coordination* algorithm, the intelligent robot system relates the spatial information of the objects and the frame from the camera to the end effector. Using this information, an online path planning algorithm plans a trajectory through inverse kinematics and control the movements of the robot arm.

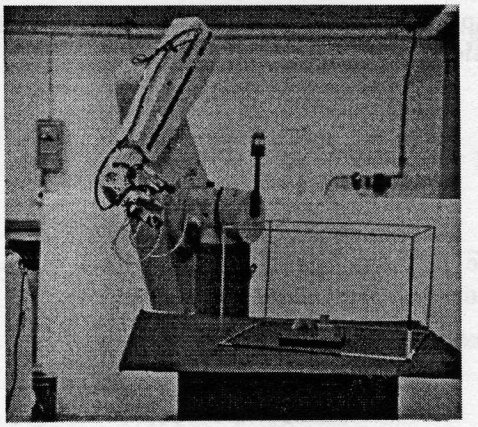


Figure 7: The PAMI intelligent robot system: a CCD camera mounted on the robot arm recognizes and determines the pose of the object and the frame. The robot arm then engages the recognized object and move it out of the frame

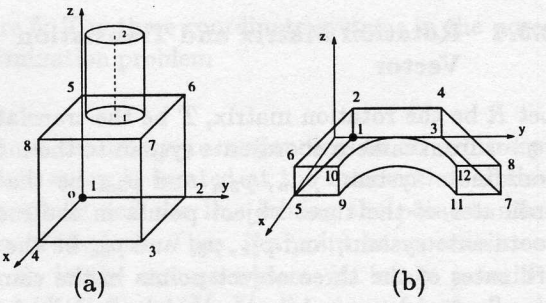


Figure 8: The CAD models for the two objects: (a) grenade and (b) bridge

3.2 Experimental Results

Experiments were conducted with two kinds of polyhedral objects, namely, "grenade" (Figure 8(a)) and "bridge" (Figure 8(b)). In the test image as shown in Figure 9, there are two "bridges" and one "grenade". In the current implementation, only grey-level information is considered while color and texture information in the 2-D image are not used.

The feature detection and grouping algorithm takes in the 2-D image from which it extracts straight lines and curve segments. The straight lines are fitted and grouped into feature groupings as shown in Figure 10 (a). The curve segments are fitted with ellipses as illustrated in Figure 10 (b). Pose hypothesis are generated according to the matching results. For each possible matchings, there could

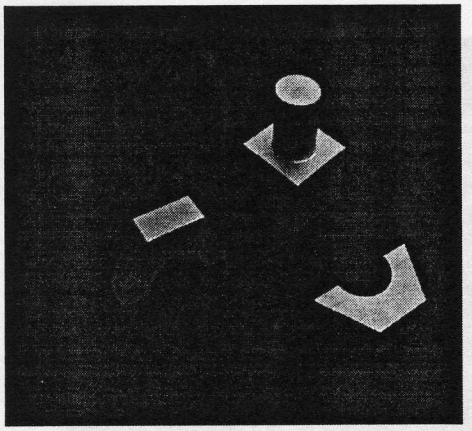


Figure 9: Two 3-D objects, a "bridge" and a "grenade", are used in the experiments of pose determination

be multiple pose solutions. The final decision is made from the plausibility measure and the reliability measure. Figure 11 presents the object recognition and pose determination result of the three objects by showing the back-projection of the CAD models, one by one, onto the 2-D image.

4 Conclusions

In this paper, we introduce a new system to perform 3-D object recognition and pose determination through a single CCD camera. The prominence of our methodology lies in (1) the use of spatial and topological feature groupings and (2) an automatic pose verification and refinement algorithm. The system has been integrated into an intelligent robot system to guide a robot arm to engage objects while avoiding obstacles along its path.

Our future work is to adopt a new algorithm of pose determination under development that is more stable and tolerant to noise. We would also incorporate an automatic camera calibration algorithm to minimize lens distortions. Finally, this system will be integrated with a model synthesis system to obtain 3-D models of polyhedral or cylindrical objects.

References

[1] P. J. Besl and R. C. Jain, "Three-dimensional Object Recognition", *Computing Survey*, Vol. 17, No. 1, pp. 74-145, 1985.

[2] I. Chakravarty and H. Freeman, "Characteristic Views as a Basis of Three-Dimensional Object Recognition", *Proc. of SPIE 336 (Robot Vision)*, 1982, pp.37-45.

[3] H. H. Chen, "Pose Determination from Line-to-plane Correspondences: Existence Condition and Closed Form Solution", *IEEE Trans. on PAMI*, Vol. 13, No. 6, 1991, pp. 530-541.

[4] M. A. Eshera & K. S. Fu, "A Graph Distance Measure for Image Analysis", *IEEE Trans. on SMC*, Vol. SMC-14, no, 3, pp 398-408, 1984.

[5] M.A. Fischler and R.C. Bolles, "Random Sample Consensus: a Paradigm for Model Fitting Applications to Image Analysis and Automated Cartography", *ACM Communications*, Vol. 24, 1981, pp. 381-395.

[6] Q. C. Gao and A.K.C. Wong, "A Curve Detection Approach Based on Perceptual Organization", *Pattern Recognition*, Vol. 26, No. 7, pp. 1039-1046, August 1993.

[7] R.M. Haralick, "Pose Estimation From Corresponding Point Data", *IEEE Trans. on SMC*, Vol. 19, No. 6, 1989, pp. 1426-1446.

[8] T. S. Huang, A. N. Netravali and H. H. Chen, "Motion and Pose Estimation Using Algebraic Methods", *Time-Varying Image Processing and Moving Object Recognition*, Cappellini, Ed., Amsterdam, The Netherlands: Elsevier, 1990, pp. 243-249.

[9] D. Marr, *Computer Vision*, 1982.

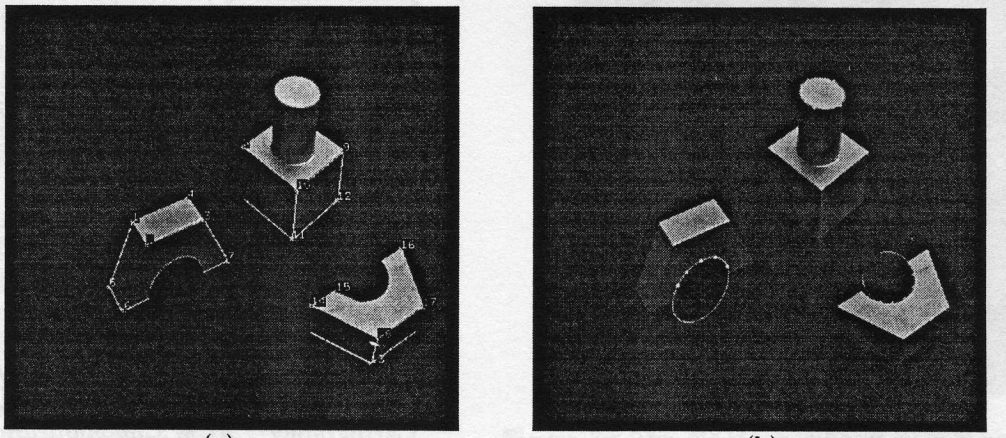
[10] A. Rosenfeld and M. H. Thurston, "Edge and Curve Detection for Visual Scene Analysis", *IEEE Trans. Comput.* C-20, pp. 562-568, 1971.

[11] A.K.C. Wong and R. Salay, "An Algorithm for Constellation Matching", *Proc. of 8th Int. Conf. on Pattern Recognition*, pp. 546-554, Oct. 1986.

[12] A. K. C. Wong, S. W. Lu & M. Rioux, "Recognition and Shape Synthesis of 3-D Object Based on Attributed Hypergraph", *IEEE Trans. PAMI*, Vol. 11, pp. 279-290, 1990.

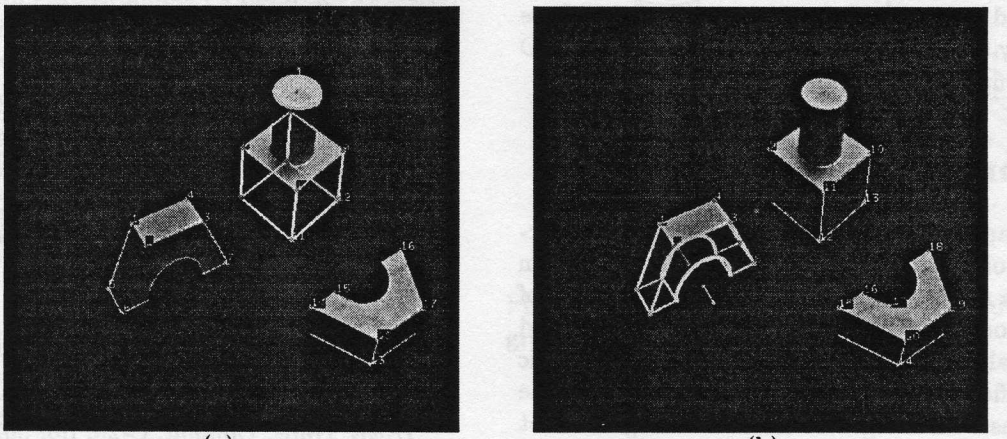
[13] A.K.C. Wong, G.R. Heppler, D.N.C. Tse and K.D. Rueb, "Robotic Vision Technology for Space Station and Satellite Applications", *Acta Astronauticas Journal*, Vol. 29, No. 12, pp. 911-930, (1993).

[14] A. K. C. Wong, "3D Vision and Modeling", *Proc. of 2nd Int'l Conf. on Mechatronics and Machine Vision in Practice*, Hong Kong, pp. 39-48, Sept. 12-14, 1995.

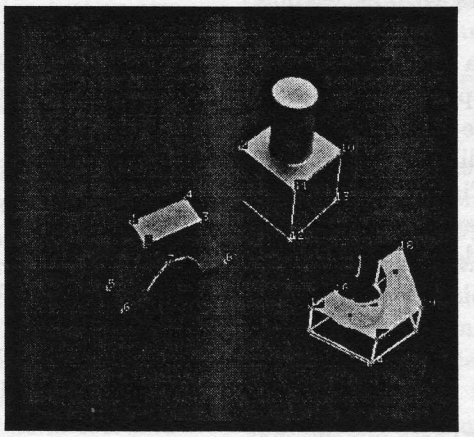


(a) (b)

Figure 10: The results of feature grouping: (a) the feature groupings consist of fitted straight lines (b) the fitted ellipses



(a) (b)



(c)

Figure 11: Back-projection of the recognition and pose determination result: (a) the grenade (b) the left bridge (c) the right bridge