Planning Sequences of Views for 3–D Object Recognition and Pose Determination *

Stanislav Kovačič, Aleš Leonardis, and Franjo Pernuš
Faculty of Electrical Engineering,
Faculty of Computer and Information Science
University of Ljubljana
Tržaška 25, 1001 Ljubljana, Slovenia
Tel. +386 (61) 1768310, Fax +386 (61) 1264630
E-mail: {stanek, feri}@.fe.uni-lj.si
E-mail: alesl@.fri.uni-lj.si

Abstract

We present a method for planning sequences of views for recognition and pose (orientation) determination of 3-D objects of arbitrary shape. The approach consists of a learning stage in which we derive a recognition and pose identification plan and a stage in which actual recognition and pose identification take place. In the learning stage, the objects are observed from all possible views and each view is characterized by an extracted feature vector. These vectors are then used to structure the views into clusters based on their proximity in the feature space. To resolve the remaining ambiguity within each of the clusters, we designed a strategy which exploits the idea of taking additional views. We developed an original procedure which analyzes the transformation of individual clusters under changing viewpoints into several smaller clusters. This results in an optimal next-view planning when additional views are necessary to resolve the ambiguities. This plan then guides the actual recognition and pose determination of an unknown object in an unknown pose.

Key words: Next-view planning, Multiple-views, View-based 3-D object representation, Object recognition, Pose determination.

*This work was supported in part by The Ministry for Science and Technology of the Republic of Slovenia (Projects J2-6187 and J2-7634) and by the Austrian national Fonds zur Förderung der wissenschaftlichen Forschung under grant S7002MAT.

1 Introduction

One of the major tasks of computer vision is the identification of an unknown 3-D object as a member of a set of known objects. In certain tasks, the identity of an object is known but the object has to be inspected (observed) from a particular view (direction), perhaps to detect defects or to measure dimensions, or it has to be re-oriented for further fine manipulation. In such cases, only the pose of an object, defined as the orientation and the position with respect to a predefined coordinate system, has to be determined. Therefore, vision systems must be able to reason about both identity and pose. Different approaches to the recognition and pose determination have been proposed, which mainly vary in the representation of 3-D objects and in the search techniques for matching data to models [6, 13].

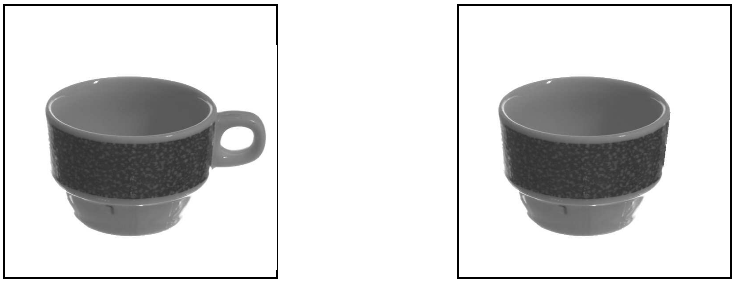
Object-centered or model-based representations usually store an explicit three-dimensional model (description) of each known object in some world coordinate frame [1, 2, 3, 6, 8]. Objects are modeled by 3-D volumetric or surface models, which are rotationally-invariant and viewpoint-independent. Even to acquire such a 3-D model is a problem by itself. There are two main approaches for model construction. In the first approach, models are obtained either manually or a computer aided design (CAD) system is used to interactively construct a CAD model. In the second approach, computer vision techniques are applied to learn models from multiple observations of actual objects using range and/or intensity images. The recognition in the framework of model-based approaches is performed by matching object data structure derived from the observed 2-D image(s) to the 3-D model data structure. 3-D to 2-D or 2-D to 3-D transformations must therefore be performed before one data structure can be matched to the other.

Viewer-centered or appearance-based representations use a set of images of an object, obtained from different views as its implicit description [6]. These representations can be acquired through an automatic learning phase. They have also the ability to deal with the combined effects of shape, reflectance properties, pose in the scene, and illumination conditions [12]. The approaches based on the appearance may be classified according to the degree of structure that is extracted from the image data and the number of views from which an unknown object must be observed in order to be recognized and its pose determined.

Concerning the structure, an extreme approach is to use explicit collections of 2-D images as object models. A similarity measure, like correlation, can then be used to determine how well the unknown image data matches the stored images of the models. Such an approach is very time consuming and requires large amounts of storage. The situation gets worse as the number of objects and views increases. Therefore, images have to be somehow compressed. A well-known image compression technique is the Karhunen-Loeve transform which is based on principal component analysis [4]. Murase and Nayar [12] used this transform to compress the image set to a low-dimensional subset. In this subset, called the eigenspace, an object is represented as a manifold. Given only one image of an unknown object in an unknown pose, the recognition system projects the image onto the eigenspace. After the object is recognized, the exact position of the projection on the manifold determines the object's pose.

Higher compression can be achieved by extracting different features from images. This approach models the objects as vectors of characteristic features, each of which corresponds to a point in the multidimensional feature space. Object identification and pose determination is then achieved by using pattern recognition techniques for finding the best match between an unknown object feature vector and the model feature vectors.

Thus far we have assumed that the recognition is based on a single image of an unknown object. The problem with this approach is that it implicitly assumes that the extracted features are sufficient to determine the identity and the pose of an object. In general, however, an image



(a) (b)

Figure 1: Two views of a cup. (a) The pose of a cup can be uniquely determined. (b) Without taking additional views, the exact pose of the cup can not be uniquely determined from this particular view.

obtained from a single view does not contain sufficient information and is therefore inadequate to uniquely solve the 3-D object- and/or pose-identification task. The reason is that some of the objects are inherently ambiguous, i.e., they differ only in the sequence of observations rather than in the value of any given observation, as shown in Fig. 1.

A natural way to resolve the ambiguities and to gain more information about an object is to observe it from additional views [5, 10, 11]. In this respect, the closest related approach to ours was reported by Gremban and Ikeuchi [5]. They presented a planning strategy for aspect resolution which limits the possible object poses to those consistent with the observed aspect. Therefore, they were interested only in rough localization determined by aspect classification. Their aspect resolution consists of searching through a tree structure (observation tree) of the collection of possible move sequences, looking for subtrees in which every aspect is resolved. Because observation trees are far too large to search exhaustively, they used heuristic search which minimized search time.
The questions that arise in the context of taking additional views are:

- How many additional views do we need to take? Observing an unknown object from all possible views would be very time consuming and also in most cases unnecessary; thus, only a smaller number of additional views should be used. The next question is therefore:

- Which views contain more information and enable a more reliable recognition? Those views that reduce the ambiguities more than other views should be applied first, which leads us to the last question:

- How do we determine the sequence of the additional views?

This paper is an attempt to answer these questions. The approach consists of a learning stage in which we derive a recognition and pose identification plan and a stage in which actual recognition and pose identification take place. In the learning stage, the objects are observed from all possible views and each view is characterized by an extracted feature vector. These vectors are then used to structure the views into clusters based on their proximity in the feature space. The clustering should satisfy the requirement that one can reliably distinguish between the views that belong to different clusters but not among the views that belong to the same cluster. Depending on the set of objects, different types of features that we choose to characterize the views, and the required reliability of the recognition and pose determination process, these clusters may contain only a few or many views. To resolve the remaining ambiguity within each of the clusters, we designed an original strategy which exploits the idea of taking additional views. We developed a procedure which analyzes the transformation of feature vectors in individual clusters under changing viewpoints, which results in an optimal next-view planning when additional views are necessary to resolve the ambiguities. This plan then guides the actual recognition and pose determination of an unknown object in an unknown pose. The proposed approach is a general framework, which offers a large variety of different possibilities regarding the recognition and pose determination problem with respect to different types of features that characterize the views and the reliability of feature classification.

The paper is organized as follows: In section 2 we formally define the problem. View planning based on cluster analysis is described in section 3. In section 4 we present the experimental results and conclude with a summary in section 5.

2 Problem formulation

Let us suppose that we are dealing with M different 3-D objects, and that each object m is represented by a set V_m of 2-D images obtained from an ordered set of N different views.

$$V_m = \{v_{m,1}, v_{m,2}, \ldots, v_{m,n}, \ldots, v_{m,N}\}, \quad (m = 1, 2, \ldots, M) \ ,$$

where $v_{m,n}$ is the image of object m from view n. The overall set V then contains the $M \times N$ available images

$$V = \cup_{m=1}^{M} V_m = \{v_{m,n};\ (m = 1, 2, \ldots, M;\ n = 1, 2, \ldots, N)\}.$$

Assume, that each 2-D image $v_{m,n}$ is described by I distinguishing characteristics or features, which form the feature vector $\mathbf{f}_{m,n}$:

$$\mathbf{f}_{m,n} = \left[{}^{1}f_{m,n}, {}^{2}f_{m,n}, \ldots, {}^{I}f_{m,n}\right].$$

In principle, the feature vector $\mathbf{f}_{m,n}$ can represent the entire image. The set F, containing all feature vectors, is therefore a representation of the set V of available 2-D images:

$$F = \{\mathbf{f}_{m,n};\ (m = 1, 2, \ldots, M;\ n = 1, 2, \ldots, N)\}.$$

Each feature vector can be represented as a point in the I-dimensional feature space. However, if the same object m is repeatedly observed from the same view n, the feature values will most probably differ from observation to observation, because features are subject to noise g:

$${}^{i}h_{m,n} = {}^{i}f_{m,n} + {}^{i}g_{m,n} \ , \ (i = 1, 2, \ldots, I).$$

Because our task is twofold, to identify an unknown object and to determine its unknown pose, let each object m being observed from view n represent a class which uniquely characterises the object and its position. Now suppose that we obtain an image w of an unknown object in an unknown pose and represent this image by a feature vector $\mathbf{x}$:

$$\mathbf{x} = \left[{}^{1}x, {}^{2}x, \ldots, {}^{I}x\right].$$

Based on the information contained in the feature vector $\mathbf{x}$, we want to determine to which class these data belong i.e., we want to reveal the identity of the object and its pose. Different techniques can be used to tackle this problem [4, 14]. However, in general it is not always possible for any recognition system to obtain a correct classification for certain views. There are two reasons for this:

1. some views are inherently ambiguous, and
2. the extracted features do not possess enough discriminative power.

Therefore, some points in the feature space will tend to lie very close one to another, causing the recognition to be at least unreliable if not even impossible. Nevertheless, in many cases this problem may be resolved by using multiple views, which is exactly the approach we are exploring.

3 View planning and recognition

In this section we show how the next views are planned for a reliable object identification and pose determination by cluster analysis. Feature vectors are first used to structure the views into equivalence classes or clusters based on their proximity in the feature space [9]. Depending on the set of objects, different types of features that we choose to characterize the views, and the required reliability of the recognition and pose determination process, these clusters may contain one, a few, or many views. To resolve the remaining ambiguity within each of the clusters, we designed an original strategy which exploits the idea that by taking an additional view, feature vectors within each cluster will be transformed into feature vectors which may form more than one cluster.

The clustering of a data set F, $F = \{\mathbf{f}_{m,n}\}$ is a partition C of F into a number of subsets, i.e., clusters $C_1, C_2, \ldots$, such that the distance or dissimilarity $d(C_i, C_j)$ between any two clusters is larger than some predefined value δ, $d(C_i, C_j) > \delta$; $\forall i, j,\ i \neq j$. The clusters together satisfy the requirements that each subset must contain at least one feature vector and that each feature vector must belong to exactly one subset. Suppose that a clustering method applied to partition the feature set F, $F = \{\mathbf{f}_{m,n}\}$ resulted in K clusters or equivalence classes $C_1, C_2, \ldots, C_j, \ldots, C_K$. Let each cluster C_j contain the feature subset F_j, such that $F = \cup_{j=1}^{K} F_j$ and $F_i \cap F_j = \emptyset$; $\forall i, j,\ i \neq j$. Depending on the set of objects, different types of features that we choose to characterize the views, and the clustering method, the number of clusters will vary and each cluster may contain one, a few, or many views. Fig. 2 (a) shows 30 feature vectors in a 2-D feature space representing 15 views of two objects and Fig. 2 (b) shows the clusters (K=9) formed by the feature vectors shown in Fig. 2 (a).

To reliably recognize an object and its pose we have to resolve the ambiguity within each of the clusters containing more than one feature vector. We apply the strategy which exploits the idea of taking additional views. Let the subset $F_j = \{\mathbf{f}_{o,p}\}$ form a cluster C_j containing more than one feature vector. We can observe the objects represented by these feature vectors from the k-th next view, where $k = 1, 2, \ldots, N-1$. By each new observation the feature vectors in F_j will be transformed to feature vectors forming the subset $F_j^k = \{\mathbf{f}_{o,p+k}\}$:

$$F_j = \{\mathbf{f}_{o,p}\} \longrightarrow F_j^k = \{\mathbf{f}_{o,p+k}\}, \quad (k = 1, 2, \ldots, N-1),$$

where o and p run over ambiguous objects and views and addition is taken modulo N. Each of the $N-1$ transformations of F_j under changing viewpoint can be analyzed by clustering F_j^k to derive the "best" next-view. For that purpose the feature space containing the transformed subset F_j^k is further partitioned into a number of clusters, as shown in Fig. 3 for subset F_8 and $k = 1$, see also Fig. 2.

Among the $N-1$ possible next views, the one yielding the largest number of clusters can be selected as the "best" one. If two or more views produce the same number of clusters, the one with the largest average intercluster dissimilarity is selected. As an illustration Fig. 4 shows the clusters formed by the features in the transformed subsets F_8^k, $k = 1, 2, 3,$ and 4 (the first 4

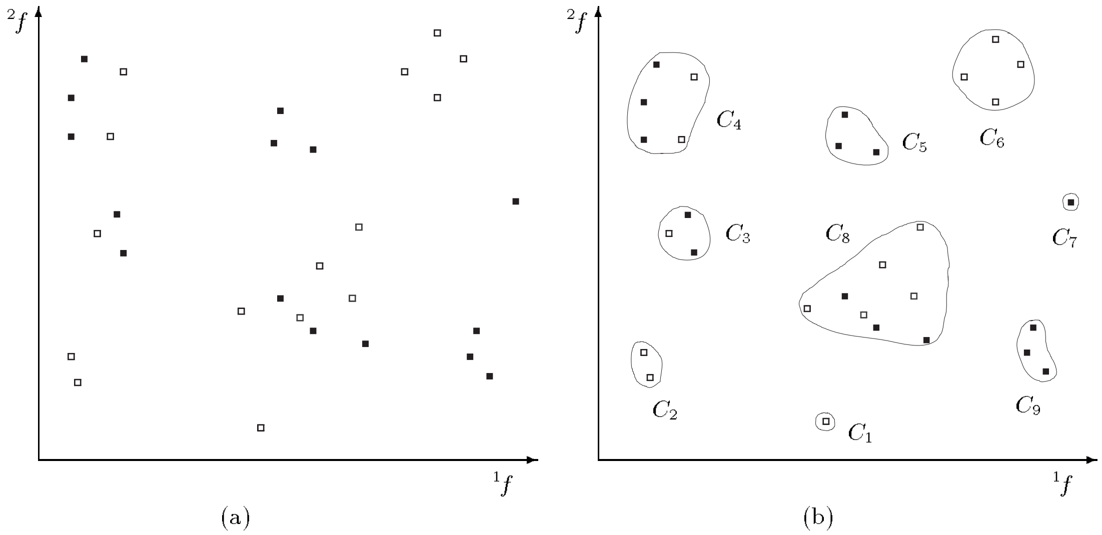


Figure 2: An example: (a) Feature vectors in a 2-D feature space representing $N = 15$ views of each of the two objects ($M = 2$). Views of the first object are depicted by open squares and views of the second object by closed squares; (b) Clusters formed by the feature vectors.

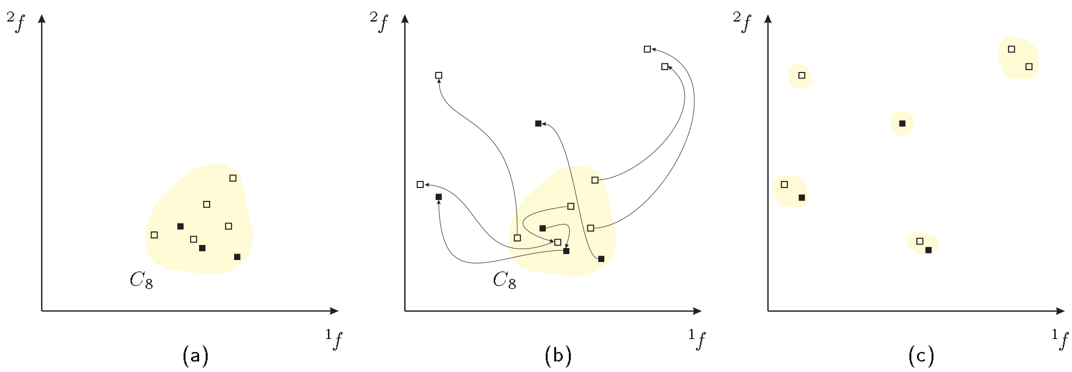


Figure 3: An example of mapping the subset F_8 under changing viewpoint. (a) The subset F_8 forming cluster C_8 (see also Fig. 2). (b) Transformation of F_8 for $k = 1$. (c) Clustering of the transformed subset F_8^1. Shaded areas represent the clusters.

next views, relative to the initial view). The largest number of clusters was obtained when the objects whose views are in subset F_8 were observed from the 4-th view relative to the current one. Therefore, this is the "best" next view of objects whose feature vectors fall into cluster C_8.

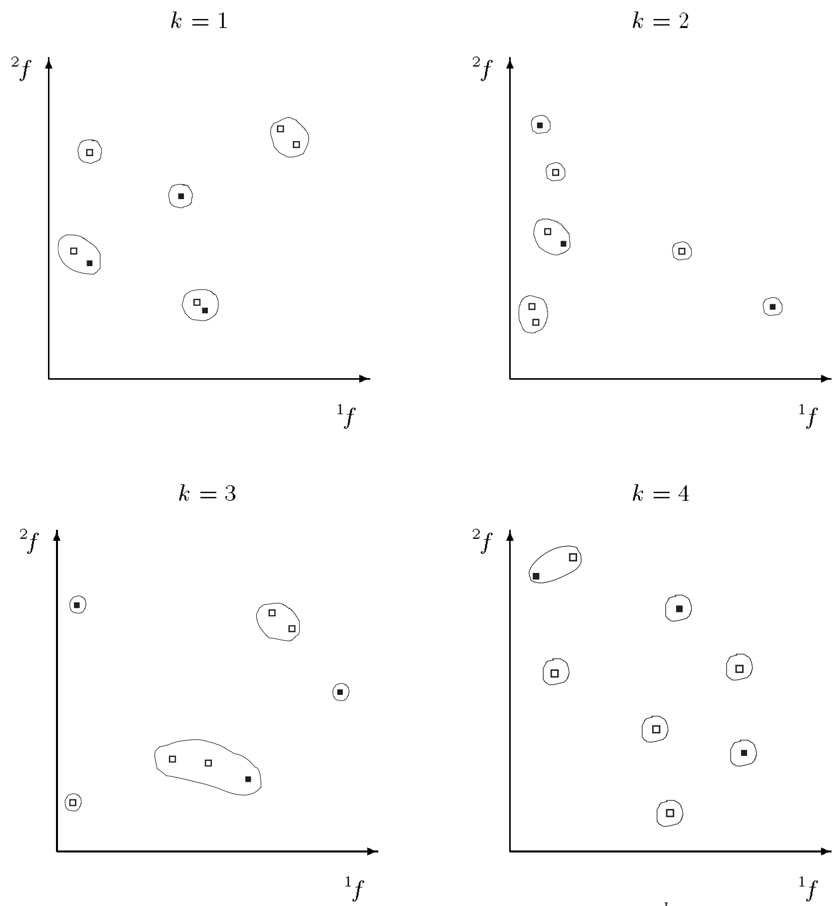


Figure 4: The clusters formed by the transformed subsets F_8^k, $k = 1, 2, 3$, and 4. Only the first four of 12 possible views relative to the current one are shown.

We recursively apply this strategy on all clusters containing more than one feature vector until only clusters with one feature vector remain, or a transformed cluster can not further be partitioned into more clusters. This procedure results in a recognition-pose-identification (RPI) plan which has the form of a tree. The root of the tree contains the set F and with each non-terminal node a subset of feature vectors and the "best" relative next view are associated. Each terminal node (leaf) may contain one or more views. In the ideal case each leaf in the tree encompasses only a single view and with each leaf, a class label indicating the object and its pose is associated. However, if a leaf contains a set of views, the identity of these views can not be determined by the selected features and clustering method.

Fig. 5 shows the RPI plan for the feature vectors depicted in Fig. 2 (a). This plan guides the actual recognition and pose determination of an unknown object in an unknown pose. The RPI plan reveals that two poses can be identified from one view, that for majority of poses an additional view is required, and that two poses can be determined by taking two additional views.

Once the RPI plan is defined, an image w of an unknown object in an unknown pose which is

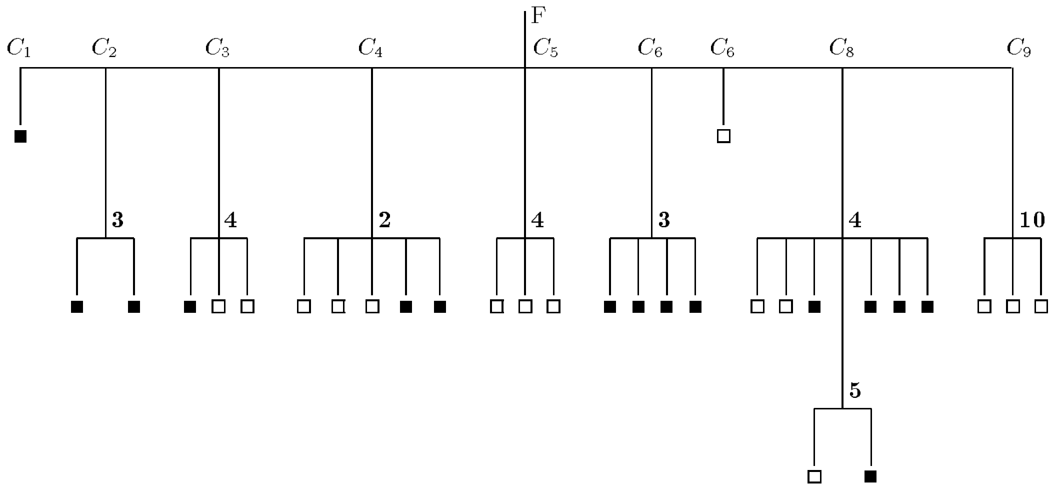


Figure 5: Recognition-pose-identification plan for the feature vectors shown in Figure 2. Each bold number gives the "best" next view relative to the current one.

represented by the feature vector $\mathbf{x}$ can be recognized on-line. This is accomplished by traversing the RPI tree until a leaf is reached. First the distance of the feature vector $\mathbf{x}$ to each of the K clusters of the first partition is determined. Among these distances the minimal one is determined and if it is smaller than some predefined threshold, $\mathbf{x}$ is assigned to the corresponding cluster. If the cluster contains more than one feature vector the object must be observed from the view associated with the corresponding node and the tree traversal is continued. When we end up in a leaf, the unknown object and its current pose can readily be determined. If necessary, the original pose can be revealed by summing up the relative moves made to acquire new views on the path from root to leaf and subtracting (modulo N) the sum from the final view.

If the features do not possess enough discriminating power, feature vectors will lie close one to the other in the feature space and clusters will only be formed for small dissimilarities δ. Because of the small intercluster distances the recognition of an unknown object in an unknown pose will not be very reliable. On the other hand, if the selected features are highly discriminative, the value of δ can be higher, the intercluster distances will be larger and this will result in more reliable recognition. Obviously there is a trade-off between the dissimilarity δ, which influences the reliability of the recognition and the number of additional views i.e., the complexity of the recognition-pose-identification plan.

4 Experiments

The procedure for building the RPI plan has been implemented and tested on various objects; four of them are shown in Fig. 7. One object at a time is placed on a motorized turntable, and its pose is varied about a single axis, namely, the axis of rotation of the turntable, see Fig. 6.

To explicitly show the power of the approach, we have *intentionally* chosen a very *restricted* set of features with which we model individual views. The features used in this experiment were based on the object silhouettes (Fig. 8).

After the silhouettes are segmented, the object contours get extracted and the two moment invariants are computed:

$$^1f = \mu_{20} + \mu_{02}, \quad ^2f = \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2},$$

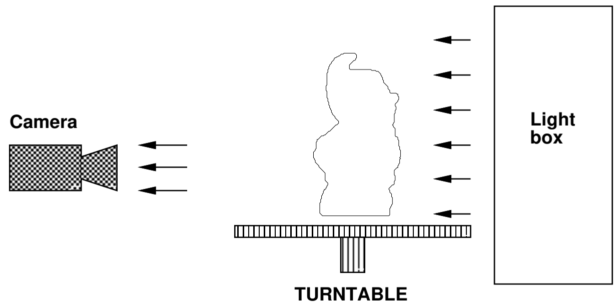


Figure 6: Our experimental system, shown schematically.

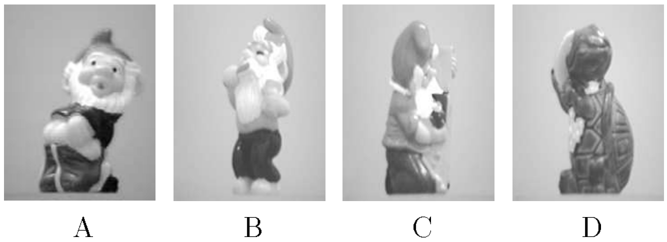


Figure 7: Four objects used in the experiments.

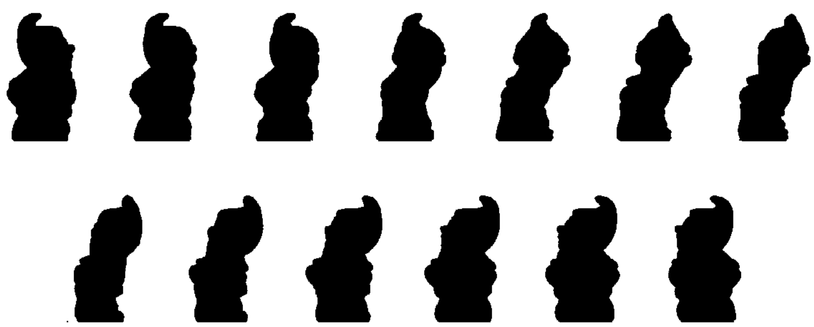


Figure 8: Silhouettes of different poses of object A depicted in Figure 7.

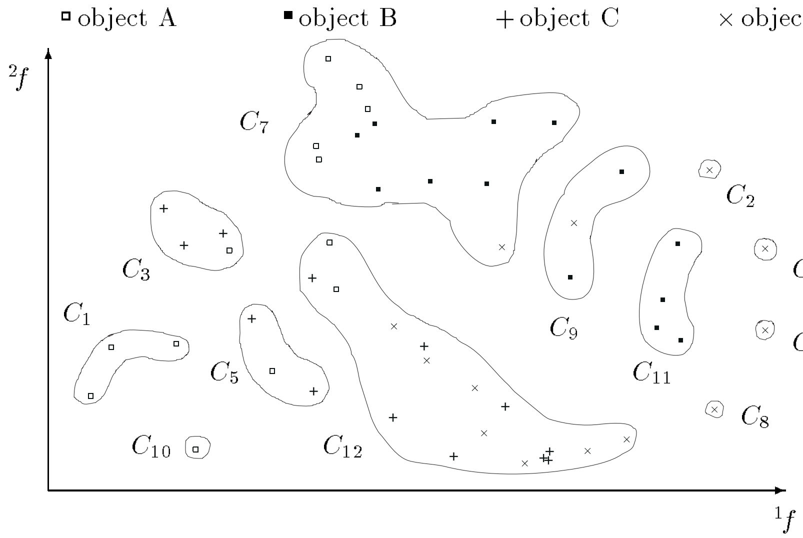


Figure 9: Feature vectors representing 13 views of each of the four objects. The outlined clusters were obtained by a minimal spanning tree clustering procedure. Edges longer than some prescribed distance δ_0 were deleted.

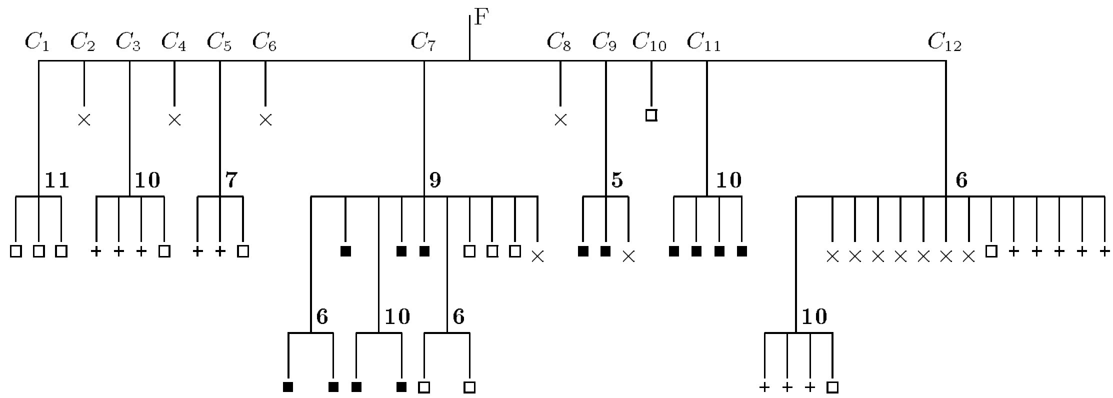


Figure 10: Recognition-pose-identification plan for δ_0. Bold numbers give the "best" relative next views.

where μ_{ij}, $(i+j=2)$ denote the central moments of second order [7]. The moments were computed from the contour points only. Fig. 9 shows the feature vectors $\mathbf{f}_{m,n}$,

$$\mathbf{f}_{m,n} = \left[{}^1f_{m,n},\, {}^2f_{m,n}\right],\ (m = 1, \ldots, 4; n = 1, 2, \ldots, 13)$$

representing 13 views of each object from Fig. 7. The outlined clusters were obtained by forming a minimal spanning tree and deleting the edges longer than a predefined distance δ, $\delta = \delta_0$.

Fig. 10 shows the RPI plan for the feature vectors depicted in Fig. 9. This plan then guides the actual recognition and pose determination of an unknown object in an unknown pose.

We performed the additional experiments to demonstrate the tradeoff between the reliability (defined by the dissimilarity δ) and the speed of recognition (number of necessary additional views). Figures 11 and 12 show the clusters obtained for $\delta = 1.15\,\delta_0$ and $\delta = 0.85\,\delta_0$, respectively.

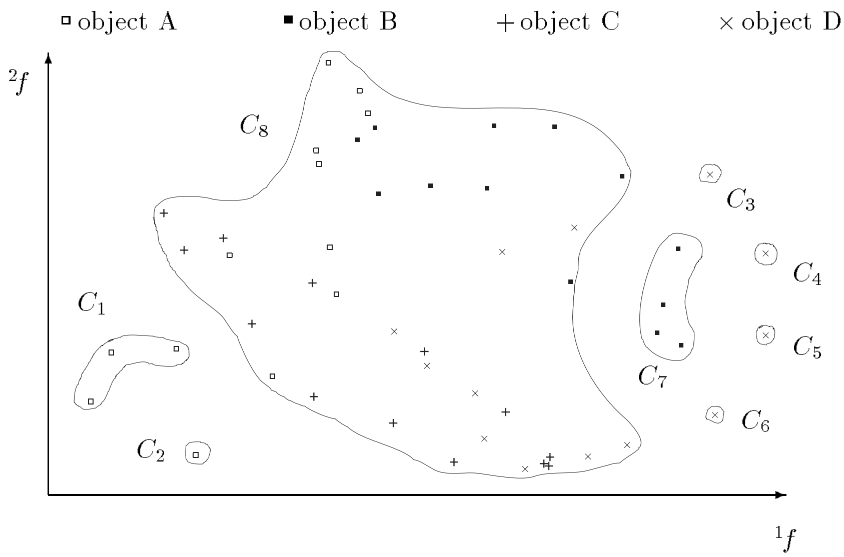


Figure 11: The initial clusters obtained for $\delta = 1.15\,\delta_0$.

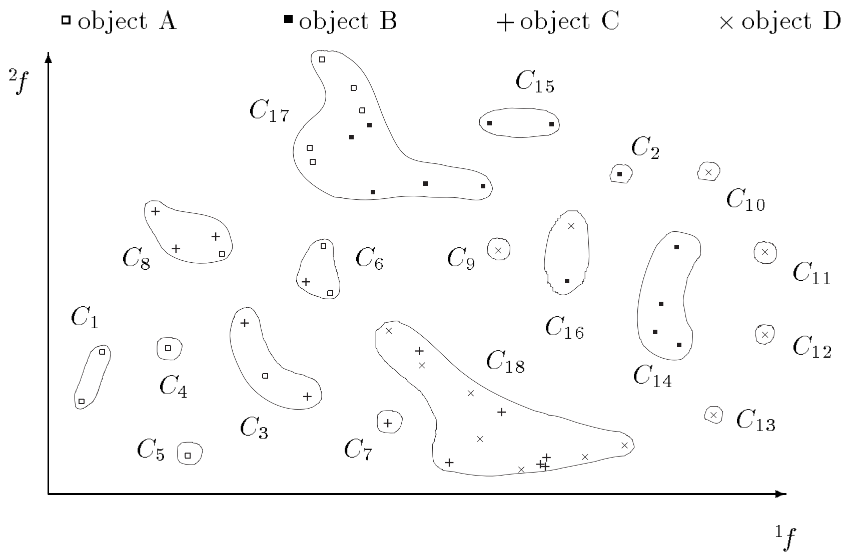


Figure 12: The initial clusters obtained for $\delta = 0.85\,\delta_0$.

The corresponding RPI plans are shown in Figures 13 and 14. If δ is increased/decreased the recognition is more/less reliable, but more/fewer additional views must be taken. The experiments with 3 different δ-s are summarised in Table 1. As δ increases, in general, less poses can be identified from one view. To identify the more ambiguous ones, additional views must be taken. For instance, for $\delta = 1.15\,\delta_0$, three additional views are needed to identify the 7 most ambiguous poses.

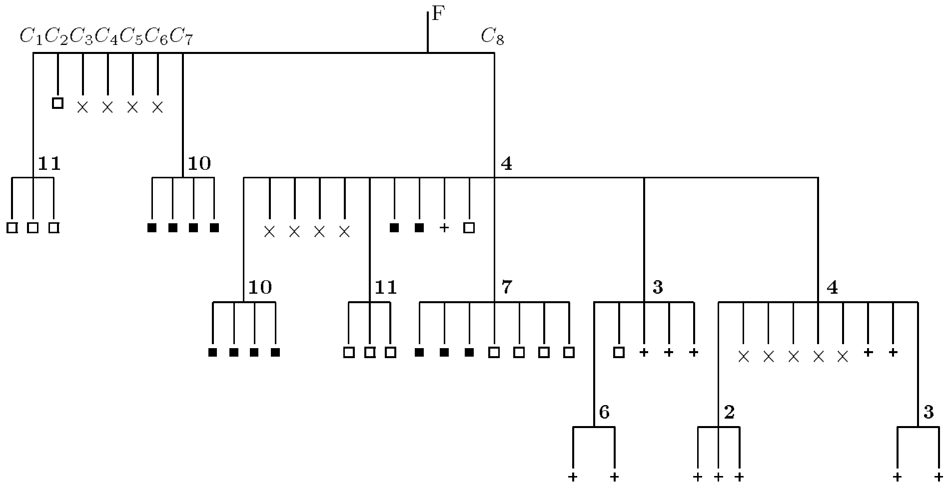


Figure 13: RPI plan for $\delta = 1.15\,\delta_0$. Bold numbers give the "best" relative next views.

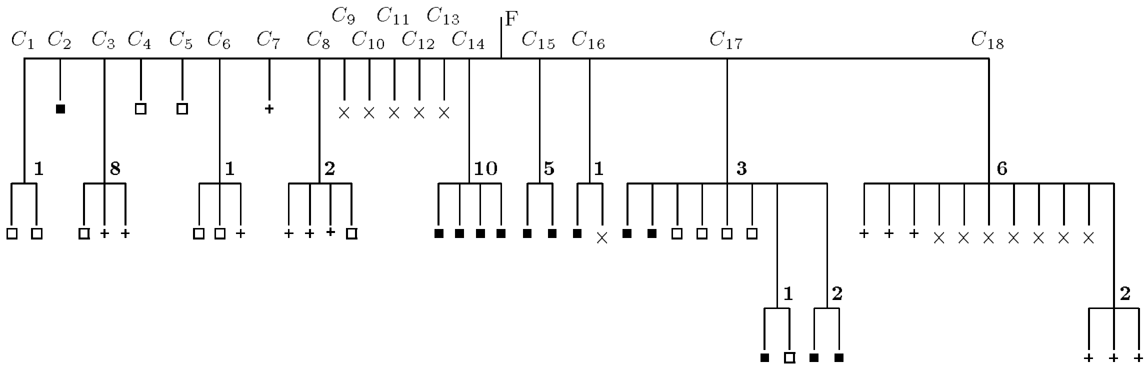


Figure 14: RPI plan for $\delta = 0.85\,\delta_0$. Bold numbers give the "best" relative next views.

	Additional views			
δ	0	1	2	3
$0.85\,\delta_0$	9	36	7	0
δ_0	5	37	10	0
$1.15\,\delta_0$	5	15	25	7

Table 1: Number of poses which can be recognized as a function of additional views for different values of δ.

5 Conclusions

We presented a method for automatic planning of additional views for recognition and pose (orientation) determination of 3-D objects of arbitrary shape. We demonstrated that ambiguities, inherent or caused by the selection of features, present in a single image can be resolved through the use of additional sensor observations. To demonstrate the approach we restricted ourselves to a rather simple situation where the objects have only one degree-of-freedom and extracted features are simple features. However, we view the major strength of the proposed work in its generality since there are no conceptual barriers to increase the number of degrees-of-freedom of the sensor and/or of the object. Besides, currently used features can be replaced with more sophisticated ones without affecting other components of the system.

References

[1] F. Ardman and J.K. Aggarwal, Model-based object recognition in dense-range images-A review, *ACM Computing Surveys*, **25/1**, 5-43 (1993).

[2] P.J. Besl and R.C. Jain, Three-dimensional object recognition, *ACM Computing Surveys*, **17/1**, 75-145 (1985).

[3] R.T. Chin and C.R. Dyer, Model-based recognition in robot vision. *ACM Computing Surveys*, **18/1**, 67-108 (1986).

[4] K. Fukunaga, Introduction to statistical pattern recognition. AP London (1990).

[5] K.D. Gremban and K. Ikeuchi, Planning multiple observations for object recognition, *International Journal of Computer Vision*, **12**, 137–172 (1994).

[6] M. Hebert, J. Ponce, T.E. Boult, A. Gross and D. Forsyth (Eds), Report of the NSF/ARPA workshop on 3D object representation for computer vision (1994).

[7] M.K. Hu, Visual pattern recognition by moment invariants. *IRE Trans. on Information Theory*, **8**, 179-187 (1962).

[8] K. Ikeuchi and P.J. Flynn (Guest Editors), Special issue on model-based vision, *Computer Vision and Image Understanding*, **61/3**, 293-474 (1995).

[9] L. Kaufman and P.J. Rousseeuw, Finding groups in data. An introduction to cluster analysis. John Wiley & Sons, New York (1990).

[10] A. Leonardis, S. Kovačič, F. Pernuš, Recognition and pose determination of 3-D objects using multiple views, V. Hlaváč and R. Šára (Eds.), *CAIP'95 Proceedings*, LNCS 970:778–783, Springer-Verlag, Berlin Heidelberg (1995).

[11] C.H. Liu and W.H. Tsai, 3D curved object recognition from multiple 2D camera views, *Comput. Vision, Graphics, Image Processing*, **50**, 177–187 (1990).

[12] H. Murase and S.K. Nayar, Visual learning and recognition of 3-D objects from appearances, *International Journal of Computer Vision*, **14**, 5-24 (1995).

[13] J. Ponce, A. Zisserman, M. Hebert (Eds.), Object representation in computer vision II, *ECCV'96 International workshop, Cambridge*, LNCS 1144. Springer-Verlag, Berlin Heidelberg (1996).

[14] J.T. Tou and R.C. Gonzalez, Pattern recognition principles. Addison-Wesley (1974).