ELSEVIER

Image and Vision Computing 18 (2000) 715–727

image AND vision COMPUTING

www.elsevier.com/locate/imavis

Appearance-based active object recognition ☆

H. Borotschnig*, L. Paletta, M. Prantl, A. Pinz

Institute for Computer Graphics and Vision, Technical University Graz, Münzgrabenstr. 11, A-8010 Graz, Austria

Received 9 December 1998; received in revised form 19 August 1999; accepted 25 October 1999

Abstract

We present an efficient method within an active vision framework for recognizing objects which are ambiguous from certain viewpoints. The system is allowed to reposition the camera to capture additional views and, therefore, to improve the classification result obtained from a single view. The approach uses an appearance-based object representation, namely the parametric eigenspace, and augments it by probability distributions. This enables us to cope with possible variations in the input images due to errors in the pre-processing chain or changing imaging conditions. Furthermore, the use of probability distributions gives us a gauge to perform view planning. Multiple observations lead to a significant increase in recognition rate. Action planning is shown to be of great use in reducing the number of images necessary to achieve a certain recognition performance when compared to a random strategy. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Action planning; Object recognition; Information fusion; Parametric eigenspace; Probability theory

1. Introduction

Most computer vision systems found in the literature perform object recognition on the basis of the information gathered from a single image. Typically, a set of features is extracted and matched against object models stored in a database. Much research in computer vision has gone in the direction of finding features that are capable of discriminating objects [1]. However, this approach faces problems once the features available from a single view are simply not sufficient to determine the identity of the observed object. Such a case happens, for example, if there are objects in the database which look very similar from certain views or share a similar internal representation (*ambiguous objects* or *object-data*); a difficulty that is compounded when we have large object databases.

A solution to this problem is to utilize the information contained in multiple sensor observations. *Active recognition* provides the framework for collecting evidence until we obtain a sufficient level of confidence in one object hypothesis. The merits of this framework have already been recognized in various applications, ranging from land-use classification in remote-sensing [16] to object recognition [2–4,15,18,19].

Active recognition accumulates evidence collected from a multitude of sensor observations. The system has to provide tentative object hypotheses for each single view.[1] Combining observations over a sequence of active steps moves the burden of object recognition slightly away from the process used to recognize a single view to the processes responsible for integrating the classification results of multiple views and for planning the next action.

In active recognition we have a few major modules whose efficiency is decisive for the overall performance (see also Fig. 1):

- The object recognition system (classifier) itself.
- The fusion task, combining hypotheses obtained at each active step.
- The planning and termination procedures.

Each of these modules can be realized in a variety of different ways. This article establishes a specific coherent algorithm for the implementation of each of the necessary steps in active recognition. The system uses a modified version of Murase and Nayar's [12] appearance-based object recognition system to provide object classifications for a single view and augments it by active recognition components. Murase and Nayar's method was chosen because it does not only result in object classifications but also gives reasonable pose estimations (a prerequisite for

☆ We gratefully acknowledge support by the Austrian 'Fonds zur Förderung der wissenschaftlichen Forschung' under grant S7003, and the Austrian Ministry of Science (BMWV Gz. 601.574/2-IV/B/9/96).

* Corresponding author.
E-mail address: borotschnig@ieee.org (H. Borotschnig).

[1] See for example Ref. [17] for a review on object recognition techniques capable of providing such view classifications.

PII: S0262-8856(99)00075-X

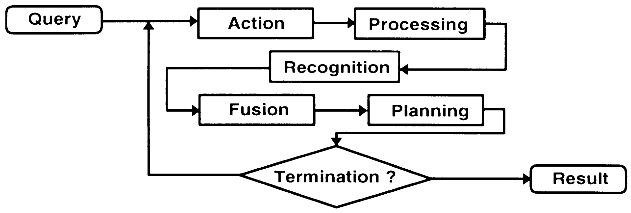


Fig. 1. The major modules involved in active object recognition. A query triggers the first action. The image data is processed and object hypotheses are established. After the newly obtained hypotheses have been fused with results from previous steps the most useful next action is planned and termination criteria are evaluated. The system will perform further active steps until the classification results have become sufficiently unambiguous.

active object recognition). It should be emphasized, however, that the presented work is not limited to the eigenspace recognition approach. The algorithm can also be applied to other view-based object recognition techniques that rely on unary, numerical feature spaces to represent objects.

2. Related research

Previous work in planning sensing strategies may be divided into off-line and on-line approaches [21]. Murase and Nayar, for example, have presented an approach for off-line illumination planning in object recognition by searching for regions in eigenspace where object-manifolds are best separated [11]. A conceptually similar but methodically more sophisticated strategy for on-line active view-planning will be presented below.

The other area of research is concerned with choosing on-line a sequence of sensing operations that will prove most useful in object identification and localization. These approaches are directly related to our work and some examples will be described here. The overall aim of all these methods is similar: given a digital image of an object, the objective is to actively identify this object and to estimate its pose. In order to fulfill this task the various systems change the parameters of their sensors and utilize the extra information contained in multiple observations. The systems differ in the way they represent objects and actions, the way they combine the obtained information and the underlying rational for planning the next observation.

Hutchinson and Kak [8] describe an active object recognition system based on Dempster–Shafer belief accumulation. Based on a set of current hypotheses about object identity and position, they evaluate candidate sensing operations with regard to their effectiveness in minimizing ambiguity. Ambiguity is quantified by a measure inspired by the entropy measure found in information theory but extended to Dempster–Shafer theory. The action that minimizes ambiguity is chosen next. Hutchinson and Kak mainly use range images as their input data and objects are represented by aspect graphs. They only present experiments in a blocks-world environment with very simple objects.

Callari and Ferrie [5] base their active object recognition system on model-based shape and position reconstructions from range data. Their system first tries to estimate the parameters of super-ellipsoid primitives approximating the range data and subsequently uses the uncertainty in these parameters to calculate a probability distribution over object hypotheses. The uncertainty in this distribution is measured by Shannon entropy and the system chooses those steps that minimize the expected ambiguity.

Sipe and Casasent's system [20] is probably the work most closely related to ours. They describe a system that uses an eigenspace representation for the objects in question. Individual views of an object are modeled as points in eigenspace and objects are represented by linear interpolation between these points. The resulting data structure is called a *feature space trajectory* (FST). View planning is accomplished by learning for each pair of objects the most discriminating viewpoint in an off-line training phase. A viewpoint is highly discriminating if the two FSTs of the inspected object pair are maximally separated. In contrast to our approach Sipe and Casasent do not model the non-uniform variance of the data points along the FST. This neglects the variability in the data and leads to suboptimal recognition performance. Furthermore, they do not fuse the obtained pose estimates, which can lead to wrong interpretations as will be demonstrated in Section 8.3.

Gremban and Ikeuchi [7] represent objects by a set of aspects. Each aspect is a set of views which are indistinguishable given the available features. If an input image is assigned to more than one such aspect, their system uses a tree search procedure to reposition the sensor and to reduce the set of possible aspects. As the tree of possible observations is far too large to be searched exhaustively, a heuristic search is used instead.

Kovačič et al. [9] cluster similar views in feature space. The system learns the changes in this clustering for each possible action and records that action which maximally separates views originally belonging to the same cluster. Doing this for all obtained clusters they pre-compile a complete *recognition–pose-identification* plan, a tree-like structure which encodes the best next view relative to the current one and is traversed during object recognition.

3. Object recognition in parametric eigenspace

Appearance-based approaches to object recognition, and especially the eigenspace method, have experienced a renewed interest in the computer vision community due to their ability to handle combined effects of shape, pose, reflection properties and illumination [12,13,22]. Furthermore, appearance-based object representations can be obtained through an automatic learning procedure and do

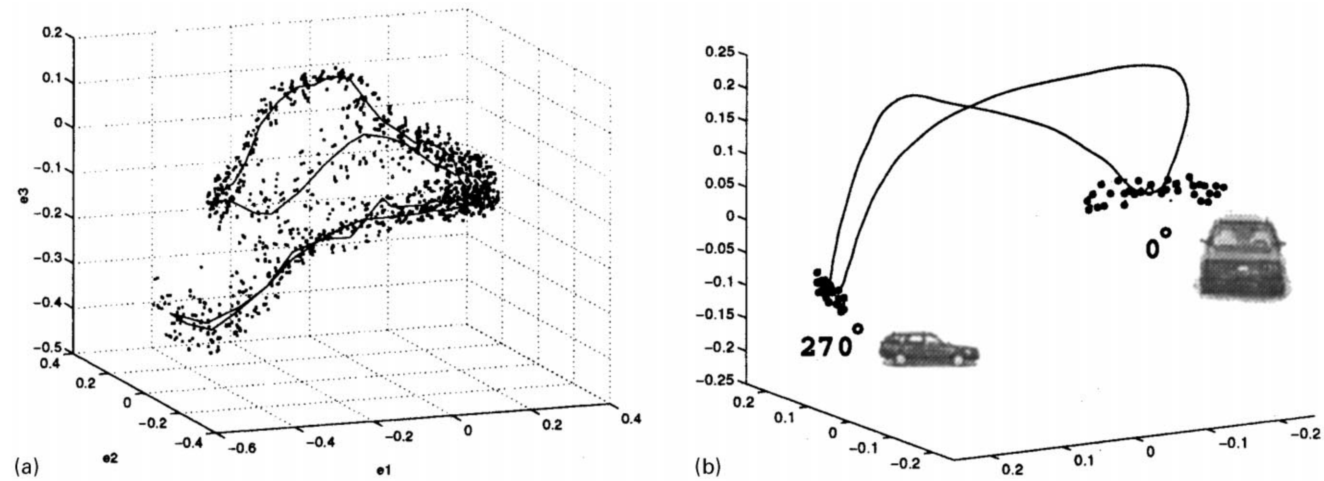


Fig. 2. (a) Exemplary eigenspace representation of the image set of one object used in the experiments to be described in Section 8.1, showing the three most prominent dimensions. (b) Illustrates even more explicitly how different views of an object give rise to likelihood distributions with different standard deviations in eigenspace. The dots indicate the positions of the learned samples for the views 270 and 0° of object o_1 used in the experiments described in Section 8.2.

not require the explicit specification of object models. Efficient algorithms are available to extend existing eigenspaces when new objects are added to the database [6]. As we will use the eigenspace object recognition method proposed by Murase and Nayar in the following we shall give a brief description of their approach (more details can be found in Ref. [12]).

The eigenspace approach requires an off-line learning phase during which images of all different views of the considered objects are used to construct the eigenspace (See for example, Fig. 6). In subsequent recognition runs the test images are projected into the learned eigenspace and assigned the label of the closest model point.

In a preprocessing step it is ensured that the images are of the same size and that they are normalized with regard to overall brightness changes due to variations in the ambient illumination or aperture setting of the imaging system. Each normalized image I can be written as a vector $\mathbf{x}(I)$ by reading pixel brightness values in a raster scan manner, i.e. $\mathbf{x} = (x_1, \ldots, x_N)^{\mathrm{T}}$ with N being the number of pixels in an image. $\mathbf{X} := (\mathbf{x}_{o_1,\varphi_1}, \mathbf{x}_{o_1,\varphi_2}, \ldots, \mathbf{x}_{o_{n_o},\varphi_{n_\varphi}})$ denotes a set of images with n_o being the number of models (objects) and n_φ being the number of views used for each model.[2] Next, we define the $N \times N$ *covariance matrix* $\mathbf{Q} := \mathbf{X}\mathbf{X}^{\mathrm{T}}$ and determine its eigenvectors $\mathbf{e}_i$ of unit length and the corresponding eigenvalues λ_i. See Ref. [12] for a discussion of various efficient numerical techniques which are useful in the given context. Since $\mathbf{Q}$ is real and symmetric, it holds that $\langle \mathbf{e}_i, \mathbf{e}_j \rangle = \delta_{ij}$, with $\langle \ldots, \ldots \rangle$ denoting the scalar product. We sort the eigenvectors in descending order of eigenvalues. The first k eigenvectors are then used to represent the image set $\mathbf{X}$ to a sufficient[3] degree of accuracy: $\mathbf{x}_{o_i,\varphi_j} \approx \sum_{s=1}^{k} g_s \mathbf{e}_s$, with $g_s = \langle \mathbf{e}_s, \mathbf{x}_{o_i,\varphi_j} \rangle$. We call the vector $\mathbf{g}_{o_i,\varphi_j} := (g_1, \ldots, g_k)^{\mathrm{T}}$ the projection of $\mathbf{x}_{o_i,\varphi_j}$ into the eigenspace. Under small variations of the parameters φ_j the images $\mathbf{x}_{o_i,\varphi_j}$ of object o_i will usually not be altered drastically. Thus for each object o_i the projections of consecutive images $\mathbf{x}_{o_i,\varphi_j}$ are located on piecewise smooth manifolds in eigenspace parameterized by φ_j.

In order to recover the eigenspace coordinates $\mathbf{g}(I)$ of an image I during the recognition stage, the corresponding image vector $\mathbf{y}(I)$ is projected into the eigenspace, $\mathbf{g}(I) = (\mathbf{e}_1, \ldots, \mathbf{e}_k)^{\mathrm{T}} \mathbf{y}(I)$. The object o_{m} with minimum distance d_{m} between its manifold and $\mathbf{g}(I)$ is assumed to be the object in question: $d_{\mathrm{m}} = \min_{o_i} \min_{\varphi_j} \|\mathbf{g}(I) - \mathbf{g}_{o_i,\varphi_j}\|$. This gives us both: an object hypothesis and a pose estimation.

4. Probability distributions in eigenspace

Before going on to discuss active fusion in the context of eigenspace object recognition we extend Murase and Nayar's concept of manifolds by introducing probability densities in eigenspace.[4] Let us assume that we have constructed an eigenspace of all considered objects. We denote by $p(\mathbf{g}|o_i, \varphi_j)$ the likelihood of ending up at point $\mathbf{g}$ in the eigenspace when projecting an image of object o_i with pose parameters φ_j. The parameters of the likelihood are estimated from a set of sample images with fixed o_i, φ_j but slightly modified imaging conditions. In our experiments we use multi- and univariate normal distributions and slightly change the viewing position and simulate small segmentation errors to obtain different sample images. In a more general setting the samples may capture not only inaccuracies in the parameters φ_j such as location and

[2] In order to simplify notation we assume $\mathbf{X}$ having zero mean.

[3] Sufficient in the sense of *sufficient for disambiguating various objects*.

[4] Moghaddam and Pentland also used probability densities in eigenspace for the task of face detection and recognition [10].

orientation of the objects but also other possible fluctuations in imaging conditions such as moderate light variations, pan, tilt and zoom errors of the camera and various types of segmentation errors. Fig. 2a depicts the point cloud in eigenspace corresponding to the full set of sample images of a specific object to be used in the experiments.

It is important to note that we interpret each captured image as a sample which is associated with a corresponding probability distribution. Accordingly capturing multiple images from (approximately) the same position amounts to sampling the underlying probability distribution for that position.

With the rule of conditional probabilities we obtain[5]

$$P(o_i, \varphi_j|\mathbf{g}) = \frac{p(\mathbf{g}|o_i, \varphi_j)P(\varphi_j|o_i)P(o_i)}{p(\mathbf{g})}. \tag{1}$$

Given the vector $\mathbf{g}$ in eigenspace the conditional probability for seeing object o_i is

$$P(o_i|\mathbf{g}) = \sum_j P(o_i, \varphi_j|\mathbf{g}). \tag{2}$$

Murase and Nayar's approach consists in finding an approximate solution for $o_{\mathrm{m}} = \arg\max_i P(o_i|\mathbf{g})$ by identifying the manifold lying closest to $\mathbf{g}$. We can restate this approach in the above framework and thereby make explicit the underlying assumptions. We obtain Murase and Nayar's algorithm if we

1. Estimate $P(o_i, \varphi_j|\mathbf{g}) = f(\|\mathbf{g}_{o_i,\varphi_j} - \mathbf{g}\|)$ with $f(x) > f(y) \Leftrightarrow x < y$. Thus they assume that the mean of the distribution lies at the one captured or interpolated position $\mathbf{g}_{o_i,\varphi_j}$. The distributions have to be radially symmetric and share the same variance for all objects o_i and all poses φ_j[6]. With this estimation the search for minimum distance can be restated as a search for maximum posterior probability:
$$\arg\max_{o_i,\varphi_j} P(o_i, \varphi_j|\mathbf{g}) = \arg\min_{o_i,\varphi_j} \|\mathbf{g}_{o_i,\varphi_j} - \mathbf{g}\|.$$
2. In the calculation of the object hypothesis the sum in Eq. (2) is approximated by its largest term:
$$P(o_i|\mathbf{g}) \approx \max_{\varphi_j} P(o_i, \varphi_j|\mathbf{g}) \Rightarrow \arg\max_{o_i} P(o_i|\mathbf{g})$$
$$= \arg\min_{o_i} \min_{\varphi_j} \|\mathbf{g}_{o_i,\varphi_j} - \mathbf{g}\|.$$

The first approximation is error-prone as the variance and shape of the probability distributions in eigenspace usually differ from point to point. This is exemplified in Fig. 2b where the point clouds for the views $\varphi = 270°$ and $\varphi = 0°$ indicate samples of the corresponding probability distributions. The experimentally obtained values for the standard deviations in this example are $\sigma_{270°} = 0.01$ and $\sigma_{0°} = 0.07$ which have to be compared to an average value of 0.04. The second approximation may lead to mistakes in case only a few points of the closest manifold lie near to $\mathbf{g}$ while a lot of points of the second-closest manifold are located not much further away.

5. Active object recognition

Active steps in object recognition will lead to striking improvements if the object database contains objects that share similar views. The key process to disambiguate such objects is a planned movement of the camera to a new viewpoint from which the objects appear distinct. We will tackle this problem now within the framework of eigenspace-based object recognition. Nevertheless, the following discussion on active object recognition is highly independent of the employed feature space. In order to emphasize the major ideas only one degree of freedom (rotation around z-axis) is assumed. The extension to more degrees of freedom is merely a matter of more complicated notation and does not introduce any new ideas. We will present experiments for one and two degrees of freedom in Section 8.

5.1. View classification and pose estimation

During active recognition step number n a camera movement is performed to a new viewing position at which an image I_n is captured. The viewing position ψ_n is known to the system through $\psi_n = \psi_0 + \Delta\psi_1 + \cdots + \Delta\psi_n$ where $\Delta\psi_k$ indicates the movement performed at step k. Processing of the image I_n consists of figure-ground segmentation, normalization (in scale and brightness) and projection into the eigenspace, thereby obtaining the vector $\mathbf{g}_n = \mathbf{g}_n(I_n)$. When using other feature spaces we have a similar deterministic transformation from image I_n to feature vector $\mathbf{g}_n$, even though feature extraction may proceed along different lines.

Given input image I_n we expect the object recognition system on the one hand to deliver a classification result for the object hypotheses $P(o_i|I_n)$ while on the other hand a possibly separate pose estimator should deliver $P(\hat{\varphi}_j|o_i, I_n)$.[7] We obtain through Eq. (1) the quantity $P(o_i, \hat{\varphi}_j|I_n) := P(o_i, \hat{\varphi}_j|\mathbf{g}_n)$ from the probability distributions in the eigenspace of all objects. From that quantity we can calculate $P(o_i|I_n) := P(o_i|\mathbf{g}_n)$ as indicated by Eq. (2). The pose estimation for object o_i is given by

$$P(\hat{\varphi}_j|o_i, I_n) = \frac{P(o_i, \hat{\varphi}_j|I_n)}{P(o_i|I_n)}. \tag{3}$$

In order to ensure consistency when fusing pose estimations obtained from different viewing positions each pose estimation has to be transformed to a fixed set of coordinates. We use the quantity $P(\varphi_j|o_i, I_n, \psi_n)$ to denote the

[5] We use lower case p for probability densities and capital P for probabilities.

[6] This follows because $P(o_i, \varphi_j|\mathbf{g})$ is only a function of radial distance $\|\mathbf{g}_{o_i,\varphi_j} - \mathbf{g}\|$ from $\mathbf{g}_{o_i,\varphi_j}$ and because that function f is the same for all o_i,φ_j.

[7] The reason for the hat on $\hat{\varphi}_j$ will become evident below.

probability of measuring the pose φ_j at the origin of the fixed view-sphere coordinate system after having processed image I_n, which has been captured at the viewing position ψ_n. In our experiments the system is initially positioned at $\psi_0 = 0°$. Therefore $P(\varphi_j|o_i, I_n, \psi_n)$ indicates how strongly the system believes that the object o_i has originally been placed at pose φ_j in front of the camera. Since the current image I_n has been captured at position ψ_n this probability is related to $P(\hat{\varphi}_j|o_i, I_n)$ through

$$P(o_i, \varphi_j|I_n, \psi_n) := P(o_i, \hat{\varphi}_j + \psi_n|I_n). \tag{4}$$

It is $P(o_i, \varphi_j|I_n, \psi_n)$ that will be used for fusion. For ease of notation we shall omit the dependence on ψ_n in the following and write only $P(o_i, \varphi_j|I_n)$.

5.2. Information integration

The currently obtained probabilities $P(o_i|I_n)$ and $P(\varphi_j|o_i, I_n)$ for object hypothesis o_i and pose hypothesis φ_j are used to update the overall probabilities $P(o_i|I_1, \ldots, I_n)$ and $P(\varphi_j|o_i, I_1, \ldots, I_n)$. For the purpose of updating the confidences, we assume the outcome of individual observations to be conditionally independent given o_i and obtain:

$$\begin{aligned} P(o_i|I_i, \ldots, I_n) &= \frac{P(I_1, \ldots, I_n|o_i)P(o_i)}{P(I_1, \ldots, I_n)} \\ &= \frac{P(I_1, \ldots, I_{n-1}|o_i)P(I_n|o_i)P(o_i)}{P(I_1, \ldots, I_n)} \\ &= \frac{P(o_i|I_1, \ldots, I_{n-1})P(I_1, \ldots, I_{n-1})P(o_i|I_n)P(I_n)P(o_i)}{P(o_i)P(o_i)P(I_1, \ldots, I_n)} \end{aligned} \tag{5}$$

$$P(o_i|I_1, \ldots, I_n) \propto P(o_i|I_1, \ldots, I_{n-1})P(o_i|I_n)P(o_i)^{-1}.$$

In the first line we have exploited the assumed conditional independence while in the last line we have summarized all factors that do not depend on the object hypotheses in a single constant of proportionality. Similarly we obtain the update formulae for the pose hypotheses

$$P(\varphi_j|o_i, I_1, \ldots, I_n) \propto P(\varphi_j|o_i, I_1, \ldots, I_{n-1})P(\varphi_j|o_i, I_n)P(\varphi_j|o_i)^{-1}, \tag{6}$$

$$P(o_i, \varphi_j|I_1, \ldots, I_n) = P(\varphi_j|o_i, I_1, \ldots, I_n)P(o_i|I_1, \ldots, I_n). \tag{7}$$

The priors $P(\varphi_j|o_i)$ and $P(o_i)$ enter at each fusion step. In our experiments every object is placed on the turntable with equal probability and $P(o_i)$ is uniform. For the purpose of simplifying the calculations we have also assumed $P(\varphi_j|o_i)$ to be uniform even though in general rigid objects have only a certain number of stable initial poses.

The assumption of conditional independence leads to a very good approximative fusion scheme which works well in the majority of possible cases. Nevertheless counter-examples exist and lead to experimental consequences. We will discuss such a case in Section 8.3.

5.3. View planning

View planning consists in attributing a score $s_n(\Delta\psi)$ to each possible movement $\Delta\psi$ of the camera. The movement obtaining the highest score will be selected next:

$$\Delta\psi_{n+1} := \arg\max_{\Delta\psi} s_n(\Delta\psi). \tag{8}$$

The score measures the utility of action $\Delta\psi$, taking into account the expected reduction of entropy for the object hypotheses. We denote entropy by

$$H(O|\mathbf{g}_1, \ldots, \mathbf{g}_n) := -\sum_{o_i \in O} P(o_i|\mathbf{g}_1, \ldots, \mathbf{g}_n) \log P(o_i|\mathbf{g}_1, \ldots, \mathbf{g}_n) \tag{9}$$

where it is understood that $P(o_i|\mathbf{g}_1, \ldots, \mathbf{g}_n) = P(o_i|I_1, \ldots, I_n)$ and $O = \{o_1, \ldots, o_{n_o}\}$ is the set of considered objects. We aim at low values for H which indicate that all probability is concentrated in a single object hypothesis rather than distributed uniformly over many. Other factors may be taken into account such as the cost of performing an action or the increase in accuracy of the pose estimation. For the purpose of demonstrating the principles of active fusion in object recognition, let us restrict attention to the average entropy reduction using

$$s_n(\Delta\psi) := \sum_{o_i, \varphi_j} P(o_i, \varphi_j|I_1, \ldots, I_n)\Delta H(\Delta\psi|o_i, \varphi_j, I_1, \ldots, I_n). \tag{10}$$

The term ΔH measures the entropy loss to be expected, if o_i,φ_j were the correct object and pose hypotheses and step $\Delta\psi$ was performed. During the calculation of the score $s_n(\Delta\psi)$ this entropy loss is weighted by the probability $P(o_i, \varphi_j|I_1, \ldots, I_n)$ for o_i,φ_j being the correct hypothesis.

The expected entropy loss is again an average quantity given by

$$\Delta H(\Delta\psi|o_i, \varphi_j, I_1, \ldots, I_n) := H(O|\mathbf{g}_1, \ldots, \mathbf{g}_n) - \int_\Omega p(\mathbf{g}|o_i, \varphi_j + \psi_n + \Delta\psi)H(O|\mathbf{g}_1, \ldots, \mathbf{g}_n, \mathbf{g})\, d\mathbf{g}. \tag{11}$$

Here φ_j is the supposedly correct pose measured at the origin of the viewsphere coordinate system and $\psi_n + \Delta\psi$ indicates the next possible viewing position. The integration runs in principle over the whole eigenspace Ω (i.e. over a sub-manifold of Ω because the images are normalized). In practice, we average the integrand over randomly selected samples of the learned distribution $p(\mathbf{g}|o_i, \varphi_j + \psi_n + \Delta\psi)$.[8] Note that $H(O|\mathbf{g}_1, \ldots, \mathbf{g}_n, \mathbf{g})$ on the right hand side of Eq. (11) implies a complete tentative fusion step performed with the hypothetically obtained eigenvector $\mathbf{g}$ at position $o_i, \varphi_j + \psi_n + \Delta\psi$.

The score $s_n(\Delta\psi)$ can now be used to select the next camera motion according to Eq. (8). The presented

[8] Similarly we may also choose those samples which were used to estimate the parametric form of the likelihoods during the learning phase.

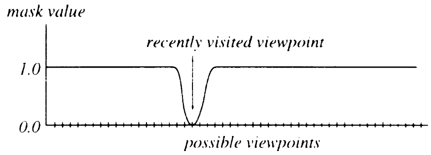


Fig. 3. The heuristic mask used to enforce global sampling.

view-planning algorithm greedily chooses the most discriminating next viewpoint. This output of the view-planning module is completely determined by the current probabilities for all object and pose hypotheses and the static probability distributions in eigenspace that summarize the learning data. Hence it is obvious that the algorithm will choose new viewing positions as long as the probabilities for object and pose hypotheses get significantly modified by new observations. Once the object–pose hypotheses stabilize the algorithm consistently and constantly favors one specific viewing position. As has already been stressed in Section 4, capturing multiple images from approximately the same viewing position does indeed give further information about the correct hypothesis. Each single image corresponds to only one specific sample while multiple images give a broader picture of the underlying probability distributions in eigenspace.

However, since inaccuracies will inevitably arise during the modeling process[9] we prefer to forbid the system to favor only one final viewing position. Thereby we diminish the potential influence of local errors in the learning data on the final object–pose hypothesis and increase the system's robustness by enforcing a global sampling. To this end the score as calculated through Eq. (10) is multiplied by a mask to avoid capturing views from similar viewpoints over and over again. The mask is zero or low at recently visited locations and rises to one as the distance from these locations increases. Using such a mask we attribute low scores to previous positions and force the system to choose the action originally obtaining second highest score whenever the system would decide to make no significant move (Fig. 3).

The process terminates if entropy $H(O|\mathbf{g}_1,\ldots,\mathbf{g}_n)$ gets lower than a pre-specified value or no more reasonable actions can be found (maximum score too low).

6. The complexity of the algorithm

In the following we denote by n_o the number of objects, by n_φ the number of possible discrete manifold parameters (total number of viewpoints) and by n_f the number of degrees of freedom of the setup. Since n_φ depends exponentially on the number of degrees of freedom we introduce n_v, the mean number of views per degree of freedom, such that $n_\varphi = n_v^{n_f}$. Finally, let us denote by n_a the average number of possible actions. If all movements are allowed we will usually have $n_a = n_\varphi$.

Before starting the discussion of the complexity of the algorithm it is important to realize that many of the intermediate results which are necessary during planning can be computed off-line. In Eq. (11) the quantity $H(O|\mathbf{g}_1,\ldots,\mathbf{g}_n,\mathbf{g})$ is evaluated for a set of sample vectors $\mathbf{g} = \hat{\mathbf{g}}_1,\ldots,\hat{\mathbf{g}}_{n_s}$ for each of the possible manifold parameters $\varphi_j + \psi_n + \Delta\psi$. We denote by n_s the number of samples per viewpoint used for action planning. The corresponding likelihoods $p(\hat{\mathbf{g}}_r|o_i,\varphi_j + \psi_n + \Delta\psi)$ and probabilities $P(o_i|\hat{\mathbf{g}}_r)$, $r = 1,\ldots,n_s$ are computed off-line such that only the fusion step in Eq. (5) has to be performed on-line before computing the entropy according to Eq. (9). Hence the complexity of calculating the score for a particular action $\Delta\psi$ is of order $O(n_o n_\varphi n_s n_o)$.

On the other hand, the complexity of calculating the score values for all possible actions is only of order

$$O(n_o n_\varphi n_s n_o + n_o n_\varphi n_a). \tag{12}$$

if a lookup table is calculated on-line. The first term $n_o n_\varphi n_s n_o$ expresses the order of complexity of calculating the fused probabilities (and the corresponding average entropies) for all the $n_o n_\varphi n_s$ possible samples that are used as potential feature vectors for view planning (n_s per view with $n_o n_\varphi$ being the total number of views). These average entropies can be stored in a lookup table and accessed during the calculation of the total average entropy reduction. Thus we need only $n_o n_\varphi n_a$ additional operations to compute all the scores $s_n(\Delta\psi)$ through Eqs. (10) and (11).

We can also take advantages of the fact that finally only hypotheses with large enough confidences contribute to action planning. This is due to Eq. (10) in which hypotheses with low confidences do not affect the calculation of the score. Hence only the n_l most likely compound hypotheses (o_i,φ_j) may be taken into account. The number n_l is either pre-specified or computed dynamically by disregarding hypotheses with confidences below a certain threshold. Usually $n_l \ll n_o n_\varphi$, for example $n_l = 10$ (taking $n_l = 2$ imitates the suggestion presented by Sipe and Casasent [20]). With this simplification we obtain the following estimate for the order of complexity of the algorithm:

$$O(n_o^2 n_\varphi n_s + n_l n_a) \propto O(n_v^{n_f}(n_o^2 n_s + n_l)). \tag{13}$$

This can be lowered again if not all possible actions are taken into account ($n_a < n_\varphi$). The above estimates explain why the algorithm can run in real-time for many conceivable situations even though the algorithm scales exponentially with the number of degrees of freedom. In fact, since the contributions of each sample and each action can be computed in parallel a great potential for sophisticated real-time applications exists. In the experiments to be described in Section 8 typical view planning steps take only about one second on a Silicon Graphics Indy workstation

[9] Outliers in the learning data or too strict parametric models for the likelihoods.

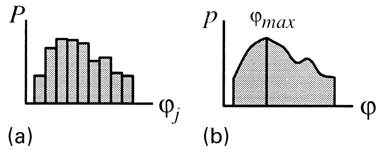


Fig. 4. In order to obtain a more accurate pose estimation, we can interpolate the discrete distribution (a) and obtain a continuous distribution (b). The pose parameter $\varphi_{\max}$ that maximizes the continuous distribution can then be regarded as a refined estimate of the true object pose and used, for example, to correct the camera position. (a) $P(\varphi_j|o_i,\mathbf{g})$. (b) $P(\varphi|o_i,\mathbf{g})$.

even though the code has been optimized towards generality rather than speed and none of the mentioned simplifications has been used.

7. Continuous values for the pose estimation

The fundamental likelihoods $p(\mathbf{g}|o_i,\varphi_j)$ are based upon learned sample images for a set of discrete pose parameters φ_j, with $j = 1\ldots n_\varphi$. It is possible to deal with intermediate poses if also images from views $\varphi_j \pm \Delta\varphi_j$ are used for the construction of $p(\mathbf{g}|o_i,\varphi_j)$. However, the system in its current form is limited to final pose estimations $P(\varphi_j|o_i,I_1,\ldots,I_n)$ at the accuracy of the trained subdivisions.

For some applications one is not only interested in recognizing the object but also in estimating its pose as precisely as possible. One way to increase the accuracy of the pose estimation is to generate additional intermediate views by generating artificial learning data through interpolation in eigenspace [12]. This does not necessitate any modifications of the presented algorithm but the new pose estimates are again limited to a certain set of discrete values.

In order to include truly continuous pose estimates it is possible to use a parametric form for the pose estimates $p(\varphi|o_i,\mathbf{g})$ instead of the non-parametric $P(\varphi_j|o_i,\mathbf{g})$. This can be achieved for example by fitting the parameters of a Gaussian mixture model to the discrete probabilities obtained from Eqs. (3) and (4) (see also Fig. 4):

$$p(\varphi|o_i,\mathbf{g}) := \sum_{m=1}^{M} P(m)\phi_{\mu_\mathrm{m},\sigma_\mathrm{m}}(\varphi|o_i,\mathbf{g}), \tag{14}$$

where the parameters $P(m)$ are the mixing coefficients and $\phi_{\mu_\mathrm{m},\sigma_\mathrm{m}}(\varphi|o_i,\mathbf{g})$ are basis functions containing parameters μ_m, σ_m. Defining the total error through

$$E(o_i,\mathbf{g}) := \sum_{\varphi_j}\left(P(\varphi_j|o_i,\mathbf{g}) - \sum_{m=1}^{M} P(m)\phi_{\mu_\mathrm{m},\sigma_\mathrm{m}}(\varphi_j|o_i,\mathbf{g})\right)^2$$

we estimate the parameters of the model through minimizing $E(o_i,\mathbf{g})$. In the most primitive case one uses a single Gaussian basis function and estimating the parameters becomes trivial.

Having established continuous pose estimations $P(\varphi|o_i,\mathbf{g})$ of higher accuracy we can use them in the usual way during fusion and view-planning. For the task of fusion Eq. (6) remains valid (without the subscript on φ_j). Various possibilities exist to let the higher accuracy of the pose estimates influence the view-planning phase. One solution is to rely on interpolated values for the quantities needed in Eqs. (10) and (11). Since this may make it difficult to assess the real quality of the pose estimate one can also consider the following alternative strategy. The system first recognizes the object using discrete and non-parametric pose estimations. After this stage, the pose is represented through Eq. (14). In order to estimate the pose more precisely the viewing position is adjusted for the most probable intermediate pose value such that the camera again captures images from views for which sample images have been learned. Subsequently view planning proceeds along the usual lines. Repeating this strategy the system accumulates a very precise value for the offset of the initial pose to the closest pose for which learning data exists.

8. Experiments

An active vision system has been built that allows for a variety of different movements (see Fig. 5). In the experiments to be described below, the system changes the vertical position of the camera, tilt, and the orientation of the turntable.

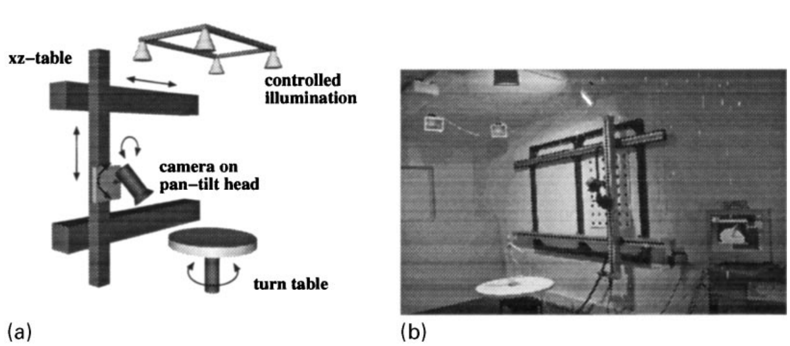


Fig. 5. A sketch plus a picture of the used active vision setup with 6 degrees of freedom and 15 different illumination situations. A rectangular frame carrying a movable camera is mounted to one side-wall. A rotating table is placed in front of the camera. (a) Sketch. (b) Setup.

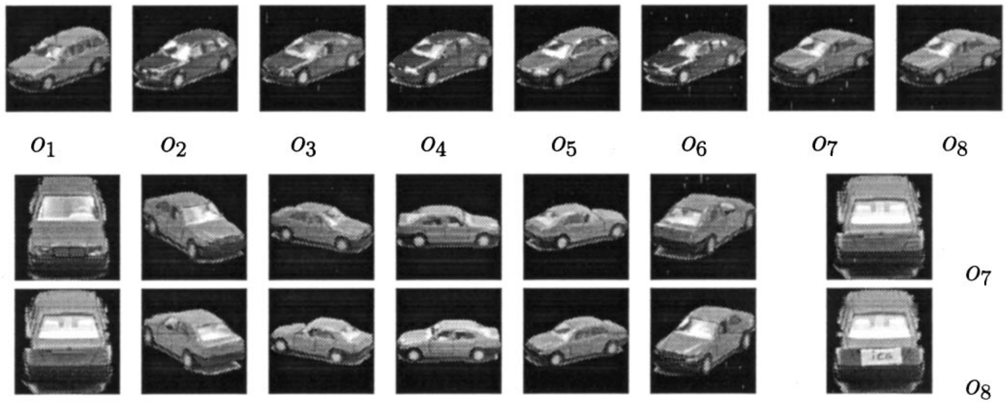


Fig. 6. Each of the objects (top row) is modeled by a set of 2D views (below, for object o_1). The object region is segmented from the background and the image is normalized in scale. The pose is shown varied by a rotation of 30° intervals about a single axis under constant illumination. A marker is attached to the rear side of object o_8 to discriminate it from object o_7 (bottom right).

8.1. Illustrative runs performed with eight toy cars

The proposed recognition system has first been tested with 8 objects (Fig. 6) of similar appearance concerning shape, reflectance and color. For reasons of comparison, two objects o_7 and o_8 are identical and can only be discriminated by a white marker which is attached to the rear side of object o_8. During the learning phase the items are rotated on a computer-controlled turntable at fixed distance to the camera by 5° intervals. The illumination is kept constant. The object region is automatically segmented from the background using a combined brightness and gradient threshold operator. Pixels classified as background are set to zero gray level. The images are then rescaled to 100×100 pixels and projected to an eigenspace of dimension 3 (see Section 9 for comments on the unusually low dimensionality). For each view possible segmentation errors have been simulated through shifting the object region in the normalized image in a randomly selected direction by 3% of the image dimension, as proposed in Ref. [11].

In Fig. 7a the overall ambiguity in the representation is visualized by the significant overlap of the manifolds of all objects, computed by interpolation between the means of pose distributions.

For a probabilistic interpretation of the data, the likelihood of a sample $\mathbf{g}$, $p(\mathbf{g}|o_i, \varphi_j)$, given specific object o_i and pose φ_j, has been modeled by a multivariate Gaussian density $N(\mu_{o_i,\varphi_j}, \Sigma_{o_i,\varphi_j})$, with mean μ_{o_i,φ_j} and covariance Σ_{o_i,φ_j} being estimated from the data that has been corrupted by segmentation errors. From this estimate both object (Eqs. (1) and (2)), and pose (Eqs. (1), (3) and (4)) hypotheses are derived, assuming uniform probability of the priors.

Table 1 depicts the probabilities for the object hypotheses in a selected run that finishes after three steps obtaining an entropy of 0.17 (threshold 0.2) and the correct object and pose estimations. Fig. 8a displays the captured images. Object o_7 has been placed on the turntable at pose 0°. Note that the run demonstrates a hard test for the proposed method. The initial conditions have been chosen such that the first image—when projected into the three-dimensional (3D) eigenspace—does not deliver the correct hypothesis. Consequently, object recognition relying on a single image

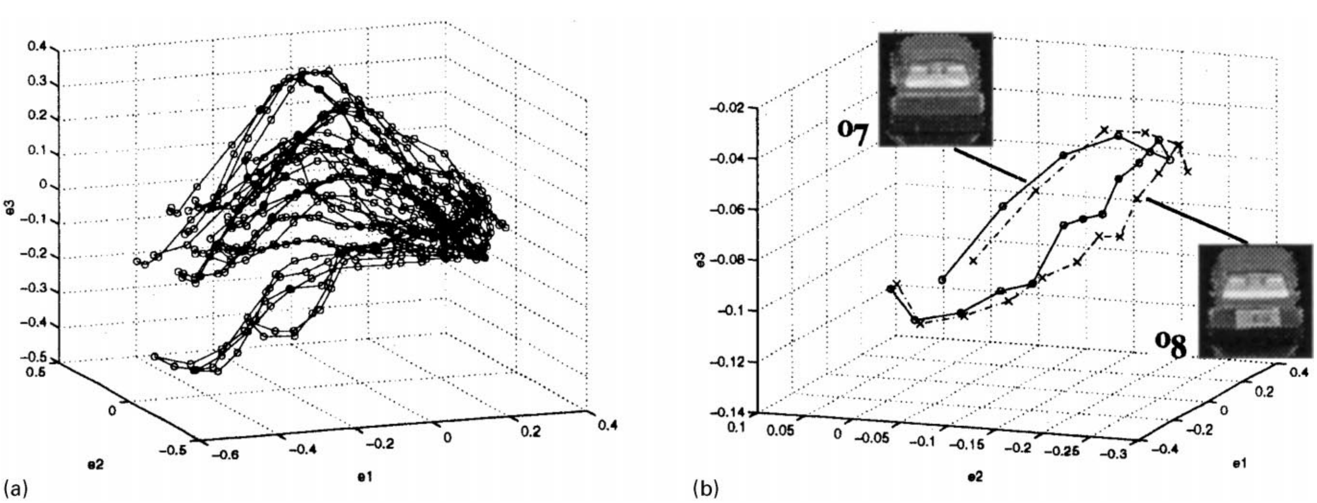


Fig. 7. Manifolds of all 8 objects (a) and distance between the manifolds of two similar objects introduced by a discriminative marker feature (b).

Table 1
Probabilities for object hypotheses in an exemplary run. See also Fig. 8a. P_f are the fused probabilities $P(o_i|\mathbf{g}_1,\ldots,\mathbf{g}_n)$. Object o_7 is the object under investigation

o_i	$\psi_0 = 0°$		$\psi_1 = 290°$		$\psi_2 = 125°$		$\psi_3 = 170°$	
	$P(o_i\|\mathbf{g}_0)$	P_f	$P(o_i\|\mathbf{g}_1)$	P_f	$P(o_i\|\mathbf{g}_2)$	P_f	$P(o_i\|\mathbf{g}_3)$	P_f
1	0.001	0.001	0.000	0.000	0.139	0.000	0.000	0.000
2	0.026	0.026	0.000	0.000	0.000	0.000	0.000	0.000
3	0.314	0.314	0.097	0.203	0.055	0.074	0.091	0.013
4	0.027	0.027	0.096	0.017	0.097	0.011	0.002	0.000
5	0.000	0.000	0.098	0.000	0.335	0.000	0.032	0.000
6	0.307	0.307	0.015	0.031	0.009	0.001	0.224	0.000
7	0.171	0.171	0.354	0.403	0.224	0.597	0.822	0.967
8	0.153	0.153	0.338	0.344	0.139	0.315	0.032	0.019

would erroneously favor object o_3 at pose $\varphi = 0°$ (pose estimations are not depicted in Table 1). Only additional images can clarify the situation. The next action places the system to position 290° and the initial probability for object o_3 is lowered. Objects o_7 and o_8 are now the favored candidates but it still takes one more action to eliminate object o_3 from the list of possible candidates. In the final step the system tries to disambiguate only between objects o_7 and o_8. Thus the object is looked at from the rear where they differ the most.

The results of longer test runs are depicted in Fig. 8b where the number of necessary active steps to reach a certain entropy threshold are depicted for both a random strategy and the presented look-ahead policy. The obtained improvements in performance will also be confirmed in more detail in the following experiment.

8.2. Experiments performed with 15 objects and 2 degrees of freedom

In a second experiment we have used 15 different objects (Fig. 9) and two degrees of freedom (Fig. 10) in camera motion. For each of the 15 objects 12 poses are considered at three different latitudinal positions amounting to a total of 540 different viewpoints. For each of the 540 viewpoints, 40 additional images have been captured from slightly varying viewing positions. Using these samples the likelihoods $p(\mathbf{g}|o_i, \varphi_j)$ have been modeled by univariate Gaussian distributions. The mean and variance have been estimated for each viewpoint separately.

An extensive set of 1440 test runs has been performed during which each object has been considered for runs with initial poses close to the learned poses (±5°). For each initial condition the system's behavior has been observed over 15 steps. The experiment has been repeated with eigenspaces of dimensions 3, 5 and 10. Each complete run has been performed twice, one time with view planning switched on, the other time relying on random motions of the camera. The recognition module has analyzed a total of 21600 images.

The results of the experiments performed with the whole database of model objects are depicted in Fig. 11 where recognition rate over the number of active recognition steps is shown for 3, 5 and 10 dimensions of the eigenspace and for planned vs. random runs. The following observations can be made:

- A static system that stops after the first observation reaches recognition rates of 30% (3D), 57% (5D), 60% (10D). These values have to be compared to 84% (3D), 96% (5D), 98% (10D) which are finally achieved through fusing the results from multiple observations.
- The final recognition level that can be obtained with a 3D eigenspace (84%) lies beyond the recognition rate of a system that relies on a single observation and is using a 10D eigenspace (69%). Thus multiple observations allow the use of much simpler recognition modules to reach a certain level of performance.
- When comparing the algorithm relying on planned actions and the use of a random strategy, attention has to be paid to the increase in recognition rate, especially during the first few observations. The system is able to

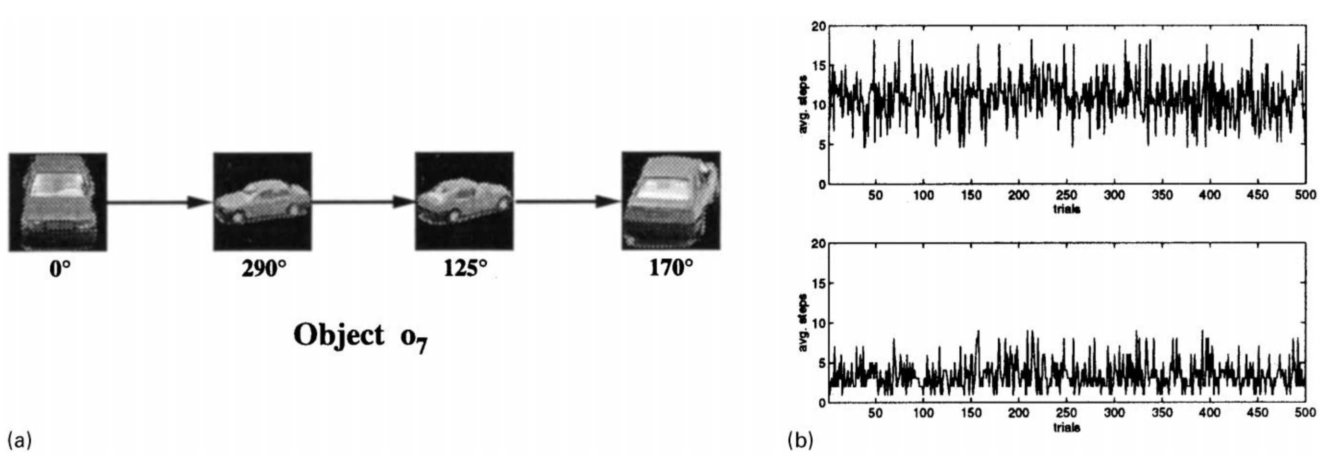


Fig. 8. (a) Sample pose sequence actuated by the planning system (see Table 1). A comparison of the number of necessary active steps (b) using a random (top) and the presented look-ahead policy (below) illustrates the improved performance.

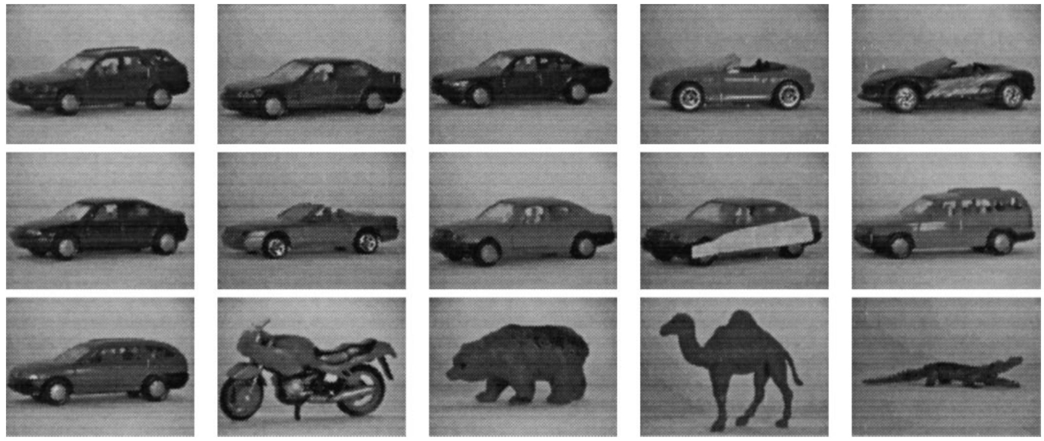


Fig. 9. Extended database consisting of 15 objects (some cars, a bike and animals). Top row (left to right): objects $o_1..o_5$, middle: $o_6...o_{10}$, bottom $o_{11}...o_{15}$. Objects o_8 and o_9 are identical except for a white marker.

come very close to its final recognition rate already after 2 to 3 steps if it plans the next action. In that region the achieved recognition rate lies more than 10% above the level obtained for the random strategy which usually needs 6 or more steps to reach its final recognition rate no matter how many dimensions of the eigenspace are used. The beneficial effect of planning can also be inferred from the much faster decrease in average entropy indicating that the system reaches a higher level of confidence already at earlier stages. In our experiment the time cost of calculating where to move (≈ 1 s) is well below the time needed to maneuver (≈ 4 s). Hence, the directed strategy is also faster than the random strategy.

- The above results can also be used to compare our approach to a static multi-camera setup. A static system is not able to perform the right movement already at the beginning of the recognition sequence but rather has to hope that it will capture the decisive features with at least one of the cameras. We have seen that using a random strategy the system needs usually 6 or more steps to reach its final recognition level. This fact translates to the assertion that a multi-camera system with randomly but statically placed cameras will need on the average 6 or more cameras to obtain a recognition rate comparable to our active system for the used set of objects.

These observations are even more conclusive when

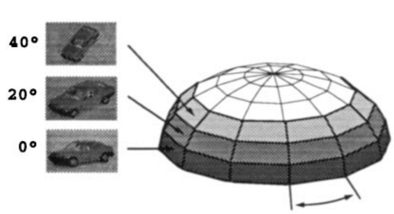


Fig. 10. Top half of the view sphere of 2D rotation about the object (at sphere center). Images are captured at three latitudinal levels (0, 20, 40°) and at 30° longitudinal intervals.

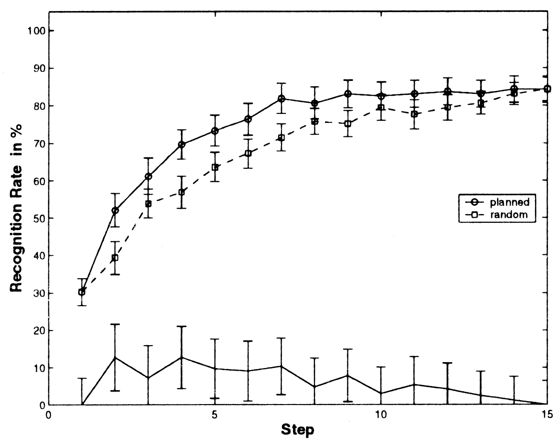
comparing the results obtained only with the two Mercedes cars o_8 and o_9. The cars are identical except for the marker on o_9. Even using a 7D eigenspace the difference in average recognition rate between planned actions and random strategy reaches a maximum beyond 30% at the second step. As the dimensionality of the eigenspace increases to 10 the maximum difference is still above 10%.

8.3. A counter-example for conditional independence in Eq. (5)

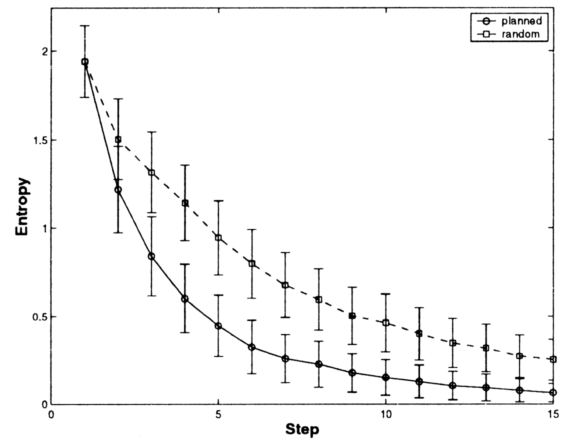
The case of the Mercedes cars is noteworthy for another reason. We can see from Fig. 12 that o_9 can be recognized without efforts using only a 3D eigenspace. This is in sharp contrast to o_8 which very often *cannot* be recognized when using a 3D eigenspace. The situation changes as the dimensionality of the eigenspace increases. The explanation of this effect leads to a deeper insight into the fusion process in Eq. (5).

The above effect occurs because the car without the marker appears to be symmetric under rotations of 180° if one is using only a 3D eigenspace. In other words, there is a significant overlap of $p(\mathbf{g}|o_8, \varphi)$ and $p(\mathbf{g}|o_8, \varphi + 180°)$ since the system does not resolve finer details at this level.

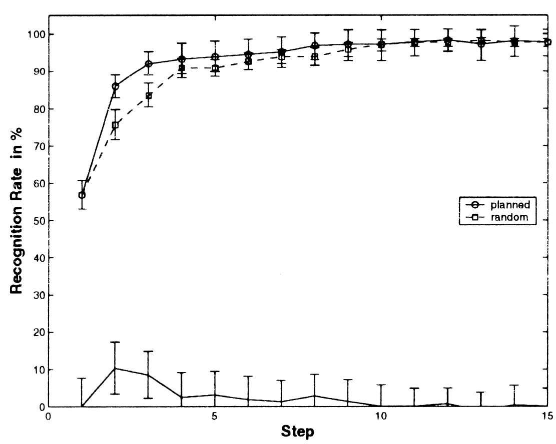
If the object database contains two identical objects that appear to be symmetric under rotation of e.g. 180° (for example two blocks) and one of the objects carries a marker on one side then fusing probabilities according to Eq. (5) will fail to integrate results correctly when trying to recognize the object without the marker. This can be understood easily if one imagines a static system with an arbitrary number of cameras placed all over the view-sphere observing the object without the marker. Each separate observation will produce equal confidences for both considered objects because each single view may stem from either of the two objects. But the whole set of observations is only possible for the object without the marker because no marker can be found even though images from opposite views have been taken. However, if fusion is based upon



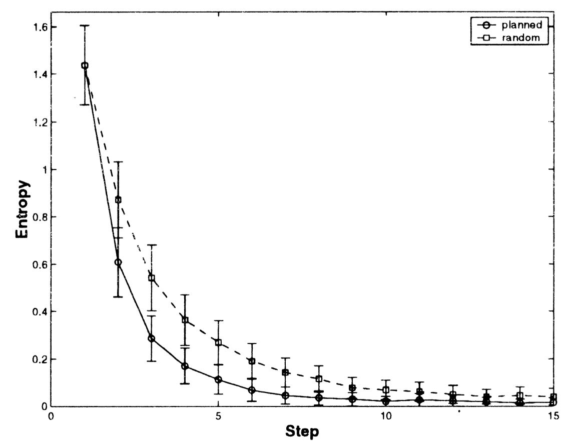
(a)



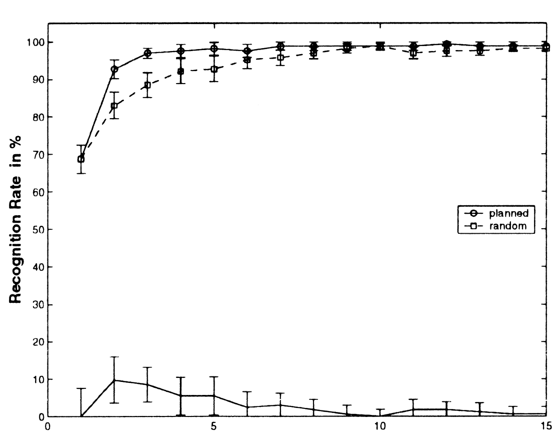
(b)



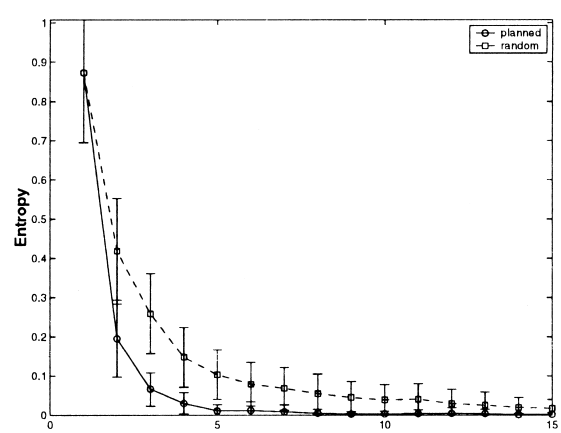
(c)



(d)

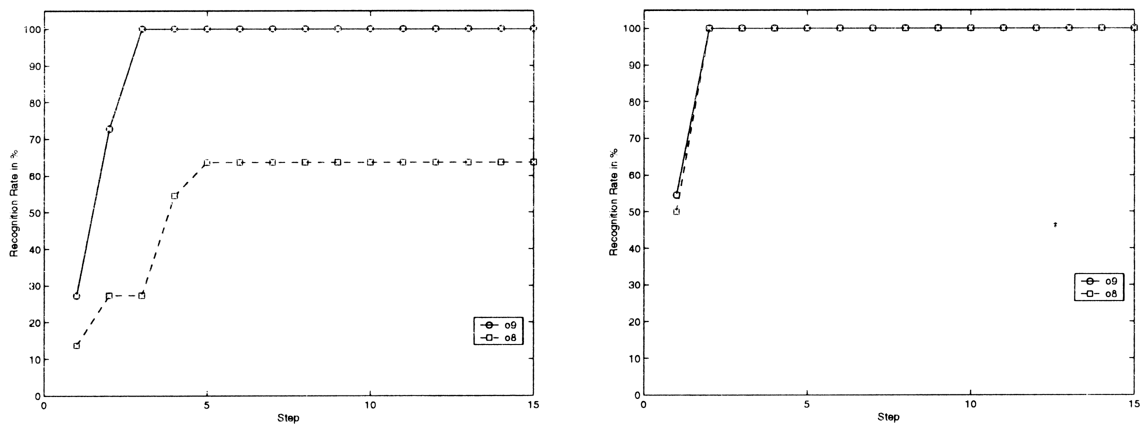


(e)



(f)

Fig. 11. Results obtained with the whole database of toy objects depicted in Fig. 9. Average recognition rate (left column) and entropy (right column) over number of steps (1…15). Each of the figures in the left column contains three plots: the average recognition rate for runs with action planning switched on (upper plot), for runs relying on a random strategy (middle plot) and the difference of the two recognition rates (lower plot). The number of dimensions of the eigenspace increases from top to bottom. In the right column the average entropy of the probability distribution $P(o_i|I_1, \ldots, I_n)$ is depicted for each step n. Each of the figures shows the entropy for runs with action planning (lower plot) and without action planning (upper plot). Again the number of dimensions of the eigenspace increases from top to bottom. (a) Rec. Rate 3d. (b) Entropy 3d. (c) Rec. Rate 5d. (d) Entropy 5d. (e) Rec. Rate 10d. (f) Entropy 10d.



(a) (b)

Fig. 12. The average recognition rate achieved for the two Mercedes cars o_8 and o_9 (with marker) using 3D and 5D eigenspaces. In both figures the upper plot corresponds to o_9, the lower plot to o_8. (a) Rec. Rate 3d. (b) Rec. Rate 5d.

Eq. (5) then this fact will not be accounted for. Instead, even after fusing all single results both object hypotheses will achieve equally high probabilities.

The naive Bayesian fusion operator has been applied widely by different authors working on active recognition tasks [4,5,15,20] since it allows for efficient information integration. We have shown that in some cases the considered fusion scheme will fail to integrate all the encountered hints. The necessary conditions for this to happen may seem to be artificial. All conceivable features for one object must also be possible for another object. But it should not be overlooked that what really counts is *not* the actual visual appearance of the objects but rather the *internal* representation (see Fig. 13a and b). This can also be concluded from the above experimental example in which the real car is *not* symmetric under rotations of 180° but its internal representation is symmetric if the eigenspace has only three dimensions. Therefore, the effect disappears if the eigenspace has enough dimensions to capture the data in greater detail (see Fig. 12b).

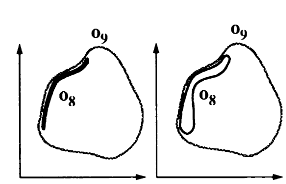
To resolve the above difficulties within the presented framework one may exploit the fact that the pose estimation for o_9 becomes more and more uniform while the pose of o_8 can still be estimated precisely (modulo the rotational symmetry).[10] A more elegant solution can be found using a more sophisticated fusion scheme which requires the explicit consideration of performed action sequences [18].

9. Conclusions

We have presented an active object recognition system for single-object scenes. Depending on the uncertainty in the current object classification the recognition module acquires new sensor measurements in a planned manner until the confidence in a certain hypothesis obtains a predefined level or another termination criterion is reached. The well-known object recognition approach using eigenspace representations was augmented by probability distributions in order to capture possible variations in the input images. These probabilistic object classifications can be used as a gauge to perform view planning. View planning is based on the expected reduction in Shannon entropy over object hypotheses given a new viewpoint. The algorithm runs in real time for many conceivable situations. The complexity of the algorithm is polynomial in the number of objects and poses and scales exponentially with the number of degrees of freedom of the hardware setup.

The experimental results lead to the following conclusions:

1. The number of dimensions of the feature space can be lowered considerably if active recognition is guiding the object classification phase. This opens the way to the use of very large object databases. Static methods are more



(a) (b)

Fig. 13. Manifolds in feature space in case each separate observation for object o_8 could as well stem from object o_9. Fig. (a) depicts the case of a practically complete overlap of possible feature values for object o_8 with object o_9. Object o_8 has to be symmetric to produce a manifold, each point of which corresponds to two (or more) views of the object. Fig. (b) illustrates the case in which o_8 is not fully symmetric, i.e. feature vectors for different views are not equal but only very similar. This can also happen if the chosen feature space is not appropriate for resolving finer details of different views (artificial symmetry due to internal representation).

[10] This work-around is only possible because we fuse the pose estimates (Eq. (6)). It cannot be applied within the algorithm suggested by Sipe and Casasent [20].

likely to face problems if the dimensionality of the feature space is too low relative to the number of objects represented (due to overlapping manifolds).
2. Even objects sharing most of their views can be disambiguated by an active movement that places the camera such that the differences between the objects become apparent. The presented view planning module successfully identifies those regions in feature space where the manifold representations of competing object hypotheses are best separated.
3. The planning phase has been shown to be necessary and beneficial as random placement of the camera leads to distinctively worse experimental results both in terms of steps and time needed for recognition. In general, a fair comparison of the directed strategy to a random strategy has to take into account the cost of additional moves (in terms of time, energy, risks,…) compared to the cost of planning. Even though these factors will strongly depend on the considered application, it can be anticipated that planning will outperform random strategies in many other settings as well.

The presented work has focused on demonstrating the principles of an active vision algorithm within a well-defined setting. We consider multi-sensor planning and planning for multi-objects scene analysis to be among the most interesting possible extensions. On the other hand, it is possible to extend the range of applications (e.g. different feature spaces and sensing techniques) without changing fundamental parts of the algorithm. For example, the currently used recognition modules are foiled by changes in lighting. Nevertheless, such changes can be embraced by recognition modules which either rely on illumination invariant features or which are built upon learning data that reflects the effect of all possible important changes in illumination [14,23].

References

[1] I. Biederman, Recognition-by-components: a theory of human image understanding, Psychological Review 2 (94) (1987) 115–147.
[2] H. Borotschnig, Uncertain information fusion in active object recognition, Number 127 in "Schriftenreihe der OCG": A. Pinz (Ed.), Computer Vision and Graphics Dissertations, Oldenbourg, Wien, München, 1999. See also http://www.icg.tu-graz.ac.at/~borotschnig and http://www.ocg.at/srtitel.
[3] H. Borotschnig, L. Paletta, M. Prantl, A. Pinz, Active object recognition in parametric eigenspace, Proceedings of the Ninth British Machine Vision Conference 2 (1998) 629–638.
[4] H. Borotschnig, L. Paletta, M. Prantl, A. Pinz, A comparison of probabilistic, possibilistic and evidence theoretic fusion schemes for active object recognition, Computing 62 (1999) 293–319.
[5] F.G. Callari, F.P. Ferrie, Autonomous recognition: driven by ambiguity, Proceedings of the International Conference on Computer Vision and Pattern Recognition (1996) 701–707.
[6] S. Chandrasekaran, B.S. Manjunath, Y.F. Wang, J. Winkeler, H. Zhang, An eigenspace update algorithm for image analysis, Graphical Models and Image Processing 59 (5) (1997) 321–332.
[7] K.D. Gremban, K. Ikeuchi, Planning multiple observations for object recognition, International Journal of Computer Vision 12 (2/3) (1994) 137–172.
[8] S.A. Hutchinson, A.C. Kak, Multisensor strategies using Dempster–Shafer belief accumulation, in: M.A. Abidi, R.C. Gonzalez (Eds.), Data Fusion in Robotics and Machine Intelligence, Academic Press, New York, 1992, pp. 165–209 chap 4.
[9] S. Kovačič, A. Leonardis, F. Pernuš, Planning sequences of views for 3-D object recognition and pose determination, Pattern Recognition 31 (10) (1998) 1407–1417.
[10] B. Moghaddam, A. Pentland, Probabilistic visual learning for object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 696–710.
[11] H. Murase, S.K. Nayar, Illumination planning for object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (12) (1994) 1219–1227.
[12] H. Murase, S.K. Nayar, Visual learning and recognition of 3-D objects from appearance, International Journal of Computer Vision 14 (1) (1995) 5–24.
[13] S. Nayar, H. Murase, S. Nene, General learning algorithm for robot vision. In SPIE, Neural and Stochastic Methods in Image and Signal Processing, volume 2304, July 1994.
[14] S.K. Nayar, H. Murase, Dimensionality of illumination manifolds in eigenspace, Technical Report CUCS-021-94, Computer Science Department, Columbia University, 1995. Revised in September.
[15] L. Paletta, A. Pinz, Active object recognition by view integration and reinforcement learning, Robotics and Autonomous Systems 31 (1–2) (2000) 1–18.
[16] A. Pinz, M. Prantl, H. Ganster, H.K. Borotschnig, Active fusion—a new method applied to remote sensing image interpretation, Pattern Recognition Letters (Special Issue on Soft Computing in Remote Sensing Data Analysis) 17 (13) (1996) 1349–1359.
[17] A.R. Pope, Model-based object recognition: a survey of recent research, Technical Report 94-04, Department of Computer Science, The University of British Columbia, January 1994.
[18] M. Prantl, Active object recognition. PhD thesis, Institute for Computer Graphics and Vision, Technical University Graz, Austria, 1999. See also http://www.icg.tu-graz.ac.at.
[19] M. Prantl, H. Borotschnig, H. Ganster, D. Sinclair, A. Pinz, Object recognition by active fusion, Intelligent Robots and Computer Vision XV: Algorithms, Techniques, Active Vision, and Materials Handling, SPIE 2904 (1996) 320–330.
[20] M.A. Sipe, D. Casasent, Global feature space neural network for active computer vision, Neutral Computation and Applications 7 (3) (1998) 195–215.
[21] K.A. Tarabanis, P.K. Allen, R.Y. Tsai, survey of sensor planning in computer vision, IEEE Transactions on Robotics and Automation 11 (1) (1995) 86–104.
[22] M. Turk, A. Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience 3 (1) (1991) 71–86.
[23] L. Zhao, Y.-H. Yang, Theoretical analysis of illumination in PCA-based vision systems, Pattern Recognition 32 (1999) 547–564.