Comparison of Appearance-Based Methods for Generic Object Recognition[1,2]

F. W. Mattern and J. Denzler

Computer Vision Group, Faculty of Mathematics and Computer Science,
Universität Passau, 94032 Passau, Germany
e-mail: mattern@fmi.uni-passau.de, denzler@fmi.uni-passau.de

Abstract—In this paper, we present a large-scale examination of different appearance-based, segmentation-free classification methods for their usability in generic object recognition. Generic object recognition is a method to handle the objects never seen before in classification by a hierarchical approach with a coarse-to-fine graduation. Unknown objects are only classified into coarse categories and rejected to be assigned to classes that are too specific. Comparison of PPCA, NN, and KPCA approaches is made on the basis of their recognition rate. The global generic recognition rate is computed for the best method, and its robustness according to different types of noise is examined. Our experiments show that the PCA-based method with nearest neighbor classification provides, in general, the best recognition rates, whereas the models based on principal component analysis outperform the other methods in computation time and model size. Gaussian KPCA models can be used for generic object recognition by varying the variance of the Gaussian kernel.

1. INTRODUCTION

3D object recognition is an important issue in image processing. Its objective is to classify 3D objects which were trained before. Depending on the kind of used model, small or distinct variances in images changed by illumination [1], disturbed by noise or occlusion [2] can be handled; whereas generic object recognition addresses the problem of classifying objects which are not known to the system. Novel objects are classified using hierarchically structured generic classes (or categories), where unknown objects should be assigned to a general class, and known objects should be additionally assigned to their object class. For instance, a generic object recognition system will classify all cups, also the unknown ones, into the category *cup*. In addition, a trained cup will be recognized as that specific cup. This type of classification is referred to as generic object modeling and recognition; it can be used when it is impossible to model all occurring objects, e.g., the variety of objects in an office environment.

Image retrieval techniques like those outlined in [3] also classify objects into generic classes but do not distinguish between known and unknown objects. In contrast to previous generic object modeling studies (e.g., [4–6]), we prevent image segmentation errors by using appearance-based methods which operate directly on the gray values of the image. These gray values are considered as a high-dimensional vector. Using the training samples, a principal component analysis (PCA) is computed to reduce the dimensions of the image vectors by projecting them into the eigenspace. The projected training vectors $\mathbf{z}$ are used to build a model in this eigenspace. Test image sets are transformed with the same projection, and the resulting vectors $\mathbf{c}$ are used to evaluate generic object approaches.

In this paper, we compare different generic object modeling and classification techniques of the eigenspace vectors. At first, we present the supervised generic object recognition system. In Section 3, we shortly describe the used classification techniques based on mixtures of probabilistic principal component analysis (MPPCA) [7], kernel principal component analysis (KPCA) [13], and nearest neighbor classification (NN) [8]. We show in the experiments (Section 4) that, in general, the NN approach performs best and point out the flaws of the approaches and possible improvements. We finish this paper with a conclusion in Section 5.

2. SUPERVISED GENERIC OBJECT RECOGNITION

A generic object recognition system should classify unknown objects into general categories. These general categories can be build automatically or can be given by a human. The algorithms for automatic categorization of the training samples show that generic classification is possible but generally differs from human-made categorizations [9]. This especially affects objects which vary more from different views than different objects vary from the same view. For example, it is easier to confuse a box and a car from the front view than the side view and the front view of a car. To gain meaningful generic classes, we have to use supervised approaches; otherwise, we can get categories which can

[1] This article was submitted by the authors in English.

[2] This work was financially supported by the German Science Foundation (DFG), grant no. DE 732/2-1.

Received December 1, 2003

Pattern Recognition and Image Analysis, Vol. 14, No. 2, 2004, pp. 255–261.

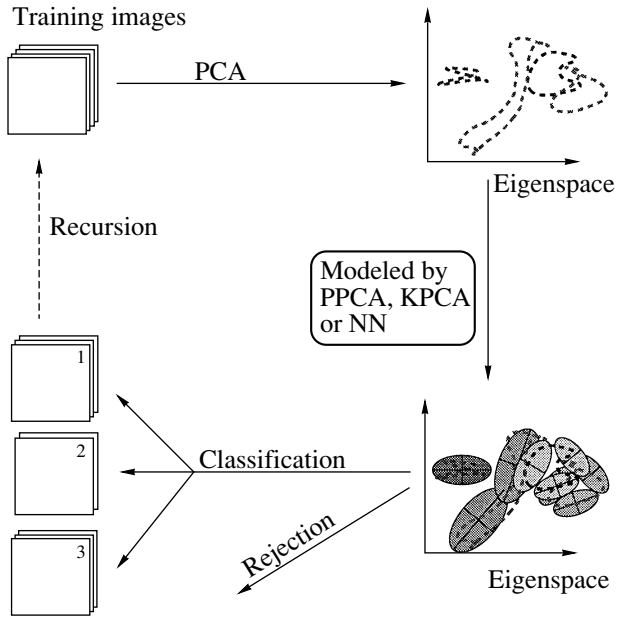

Fig. 1. Hierarchical approach for generic object recognition: dimensionality reduction with PCA, modeling with PPCA, KPCA, or NN, and classification on the basis of the used model.

be clearly distinguishable from each other but unimportant for solving the task of a robot or are not of interest for the user.

Supervised approaches need explicit assignments of objects to generic classes. This knowledge about the mapping of objects to generic classes can be used for feature selection with the Fisher transformation [11] or for building a nonlinear model for each generic class. Figure 1 shows a sketch of the generic object recognition approach.

In the training step, a PCA is performed for each image set. This transformation reduces the dimensions of the images. In contrast to the unsupervised case, a model is generated for each generic class and object class which is known to the system. Classification is done depending on the model type by maximum likelihood (ML) or NN classification. A rejection threshold is used to determine whether the objects are still represented by the generic model or not. If the object is not rejected, it will be classified in the next finer hierarchy level. More details for model building can be found in [9].

3. SUPERVISED GENERIC OBJECT MODELING WITH NONLINEAR METHODS

Using the supervised generic object recognition approach described in Section 2, we get a vector in the eigenspace for each training image. In general, the PCA-transformed image vectors of a generic class do not form a compact cluster but a nonlinear distribution. We use three methods to model these distributions: the MPPCA [7], KPCA [13], and NN [8] classification approach.

The NN approach just saves all trained eigenspace feature vectors $\mathbf{z}_{i,m} \in \Omega_m$ of each model Ω_m and calculates the nearest neighbor of the test feature vectors. If there is no trained eigenspace feature vector in the neighborhood of the test feature vector, the vector will not be assigned to any class. To decide whether or not a vector should be rejected, a threshold is experimentally determined. A rejection threshold can be chosen between zero and one using the distance measure

$$D(\mathbf{c}, \Omega_m) := \max_{\mathbf{z}_{i,m} \in \Omega_m} (\|\mathbf{c} - \mathbf{z}_{i,m}\| + 1)^{-1} \quad (1)$$

between the test vector $\mathbf{c}$ and the mth model Ω_m which contains the training vectors $\mathbf{z}_{i,m}\ \forall i \in 1, \ldots, n$.

The MPPCA approach tries to overcome the nonlinearity by combining linear PPCA models as described in [7, 9, 10]. A linear PPCA model is a generative model which explains the present feature vectors $\mathbf{z}_i \in \mathbb{R}^d$ as generated by a q-dimensional random vector $\mathbf{x}_i$ ($q < d$), like

$$\mathbf{z}_i = \mathbf{W}\mathbf{x}_i + \boldsymbol{\mu} + \boldsymbol{\epsilon}, \quad (2)$$

where $\mathbf{W}$ is the so-called factor loading matrix, $\boldsymbol{\mu}$ is a constant displacement vector, and $\boldsymbol{\epsilon}$ is a noise vector. It is assumed that $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$ as well as $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Psi})$ are zero mean Gaussian distributed random vectors, where $\mathbf{I}_q$ is a $q \times q$-dimensional identity matrix and $\boldsymbol{\Psi}$ is a $d \times d$-dimensional diagonal covariance matrix. Consequently, the observation $\mathbf{z}_i$ is also Gaussian-distributed.

The model from Eq. (2) can be easily extended to a mixture model of n Gaussian distributions. The observation vectors $\mathbf{z}_i$ are then modeled by

$$\mathbf{z}_i = \sum_{k=1}^{n} \omega_k(\mathbf{W}_k\mathbf{x}_i + \boldsymbol{\mu}_k + \boldsymbol{\epsilon}_k) \quad (3)$$

with $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$ and $\boldsymbol{\epsilon}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Psi}_k)$. The value ω_k is the weight of the kth mixture component, $\boldsymbol{\Psi}_k$ is again a diagonal covariance matrix of the observation noise. If a set of n observations $\mathbf{z}_i$ is given, the unknown parameters of the factor model $\mathbf{W}_k$, ω_k, $\boldsymbol{\mu}_k$, and $\boldsymbol{\Psi}_k$ can be estimated using the EM algorithm [7, 12]. For approximating the PCA, the diagonal covariance matrix $\boldsymbol{\Psi}$ is restricted to have identical elements ($\boldsymbol{\Psi} = \sigma^2\mathbf{I}_d$). Details are explained in [7]. A supervised approach is achieved using a mixture of PPCA for each generic class. On the basis of *a posteriori* probability, the models are compared and the rejection threshold is defined.

Another way to separate the nonlinear distributed models from each other is to project the vectors $\mathbf{c}$ with a nonlinear function $\varphi_m(\mathbf{c})$ into a higher-dimensional space in which the vectors can be separated linearly. The idea of the KPCA approach [13] is to avoid calculations of the projection by reducing the operations on

the high-dimensional vectors to scalar products $(\varphi_m(\mathbf{c}_x), \varphi_m(\mathbf{c}_y))$. These can be evaluated in the original space using the so-called kernel function $k(\mathbf{c}_x, \mathbf{c}_y) := (\varphi_m(\mathbf{c}_x), \varphi_m(\mathbf{c}_y))$. This evaluation is only valid for kernel functions for which the Mercer condition holds [13], like the polynomial kernel

$$k(\mathbf{c}_x, \mathbf{c}_y) := (\mathbf{c}_x, \mathbf{c}_y)^d, \quad d = 2, 3, \ldots, \tag{4}$$

and the Gaussian kernel

$$k(\mathbf{c}_x, \mathbf{c}_y) := \exp(-0.5\sigma^{-2}\|\mathbf{c}_x - \mathbf{c}_y\|^2). \tag{5}$$

For ML classification, we need the probability $p(\mathbf{c}|\Omega_m)$ that a feature vector $\mathbf{c}$ was generated by the nonlinear model Ω_m, which can be approximated by an energy value

$$E(\mathbf{c}, \Omega_m) \approx \ln(p(\mathbf{c}|\Omega_m)). \tag{6}$$

The energy value can be denoted by

$$E(\mathbf{c}, \Omega_m) := \tilde{\varphi}_m(\mathbf{c})^T \Sigma_{\tilde{\varphi}_m}^{-1} \tilde{\varphi}_m(\mathbf{c}) \tag{7}$$

with $\Sigma_{\tilde{\varphi}_m}^{-1}$ being the covariance matrix in the high-dimensional space and the nonlinear function

$$\tilde{\varphi}_m(\mathbf{c}) := \varphi_m(\mathbf{c}) - \frac{1}{k}\sum_{i=1}^{k} \varphi_m(\mathbf{z}_{i,m}), \tag{8}$$

centering the projected vectors $\varphi_m(\mathbf{c})$ by subtracting the mean of the projection of the training vectors $\mathbf{z}_{i,m}$. The calculation of the energy value can be rewritten [14], and the scalar products of the nonlinear function can be replaced by the kernel function $k(\mathbf{c}_x, \mathbf{c}_y)$, which leads to the following form:

$$E(\mathbf{c}, \Omega_m) = \sum_{s=1}^{r}\left(\left(\sum_{i=1}^{n} \alpha_{i,m}^s \tilde{k}_m(\mathbf{z}_{i,m}, \mathbf{c})\right)^2 (\lambda_{s,m}^{-1} - \lambda_{\perp,m}^{-1})\right) + \lambda_{\perp,m}^{-1}\tilde{k}_m(\mathbf{c}, \mathbf{c}), \tag{9}$$

where $\lambda_{s,m}$ for $s = 1, \ldots, r$ are the r largest eigenvalues and $\alpha_{i,m}^s$ are the components of the associated eigenvectors $\boldsymbol{\alpha}_m^s$. The centered kernel function $\tilde{k}_m(\cdot, \cdot)$ can be denoted by

$$\begin{aligned}\tilde{k}_m(\mathbf{c}_x, \mathbf{c}_y) &:= k(\mathbf{c}_x, \mathbf{c}_y) \\ &- \frac{1}{n}\sum_{j=1}^{n}(k(\mathbf{c}_x, \mathbf{z}_{j,m} + k(\mathbf{c}_y, \mathbf{z}_{j,m})) \\ &+ \frac{1}{n^2}\sum_{j=1}^{n}\sum_{l=1}^{n} k(\mathbf{z}_{j,m}, \mathbf{z}_{l,m}).\end{aligned} \tag{10}$$

The parameter $\lambda_{\perp,m}$ is a regularization term which expresses the assumed variance in the orthogonal directions. To normalize the energy value of the Gaussian kernel, we set

$$\lim_{\|\mathbf{c} - \mathbf{z}_{j,m}\| \to \infty} k(\mathbf{c}, \mathbf{z}_{j,m}) = 0, \quad \forall \mathbf{z}_{j,m}, \tag{11}$$

which ensures that an infinitely distant feature vector will get the same energy value. We obtain the normalization factor κ_m as

$$\kappa_m = \sum_{s=1}^{r}\left(\left(\sum_{i=1}^{n} \alpha_{i,m}^s(\bar{\mu} - \mu_i)\right)^2 (\lambda_{s,m}^{-1} - \lambda_{\perp,m}^{-1})\right) + \lambda_{\perp,m}^{-1}(1 + \bar{\mu}) \tag{12}$$

with

$$\mu_i = \frac{1}{n}\sum_{j=1}^{n} k(\mathbf{z}_i, \mathbf{z}_j), \quad \bar{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mu_i. \tag{13}$$

4. EXPERIMENTAL EVALUATION AND RESULTS

We use the recognition rate at the first hierarchy to compare different generic object recognition approaches. Because of using a hierarchical approach, there are multiple definitions of a recognition rate. One is the recognition rate at a hierarchy level, which evaluates the classification into categories of that level. Another is the global generic recognition rate, where an object is classified correctly if a known object is classified into the right categories and the right object class, and an unseen object is classified into the right categories and are rejected in too specific categories where the object does not match. Thus, a global generic recognition rate cannot be better than the recognition rate at the first hierarchy level.

As outlined in Fig. 1, we reduced the input dimensions using PCA and examined the recognition rates of the NN classifier, the KPCA with a Gaussian kernel as described in Section 3, and a mixture of five one-dimensional PPCA models as described in [7, 9, 10]. For testing, we use the COIL-100 [15] database, which contains 7200 images of the size 128 × 128 of 100 objects. We divided the COIL-100 database into three test sets: the training set used for verification, a test set which contains untrained images of known objects, and a generic test set with images of untrained objects. Figure 2 shows some examples of the test sets of this database.

Comparing the best results (see Table 1), one can see that the NN approach is able to classify almost all objects correctly. The PPCA model does not completely represent the training set. The recognition rate of the training and test sets is over 90%. But it provides good results at the generic test set. The standard KPCA classifies the training set correctly and performs very well on the test set but fails on the generic test set. Increasing the variance σ^2 (see also Eq. (5)) of the Gaussian kernel improves the generalization of the

Fig. 2. Examples of test images: (left) test images where images from similar views are trained; (right) generic test images of objects which are omitted in the training.

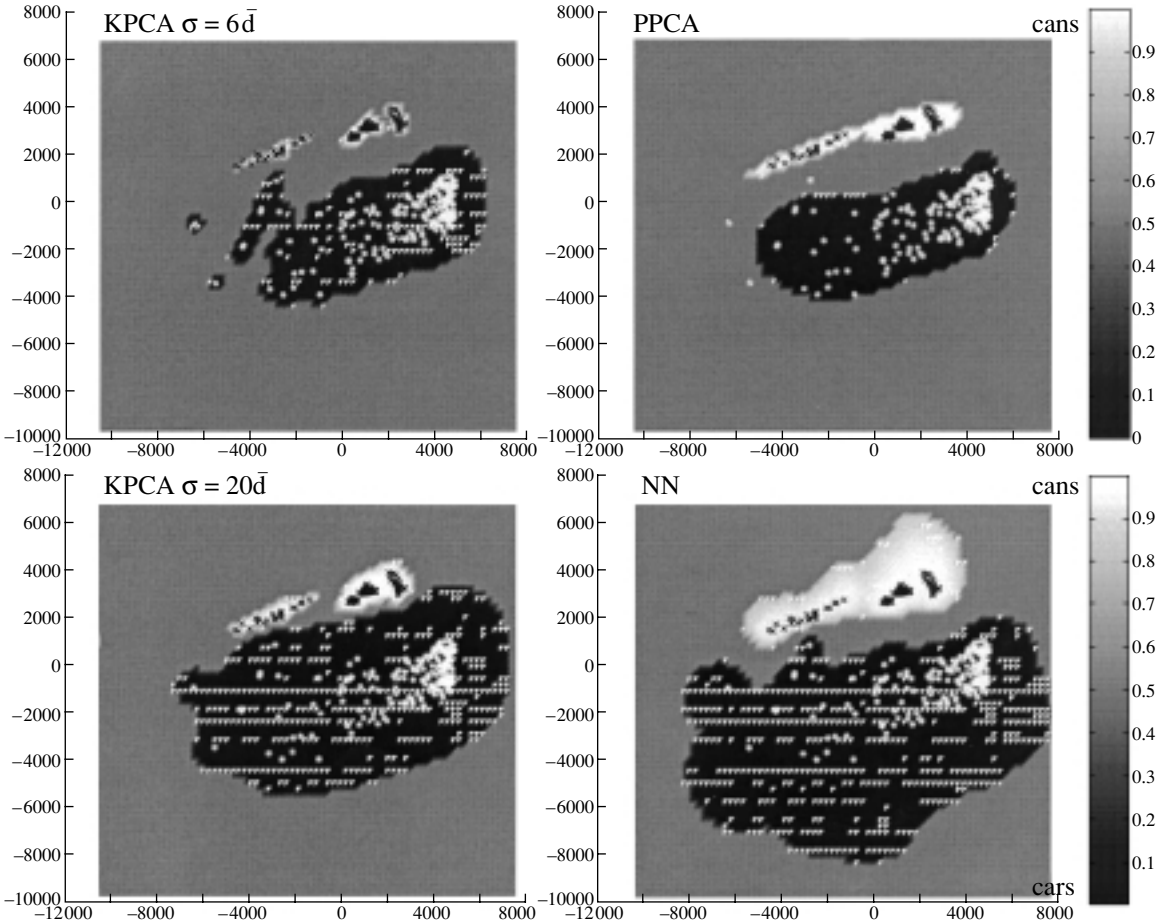

Fig. 3. Simple visual examples of KPCA, NN, and PPCA models: input images of cans (black crosses) and cars (white stars) are PCA projected in the 2D eigenspace; the model of the can and car classes are represented by the white and black area, respectively; the rejection class is represented by the gray area.

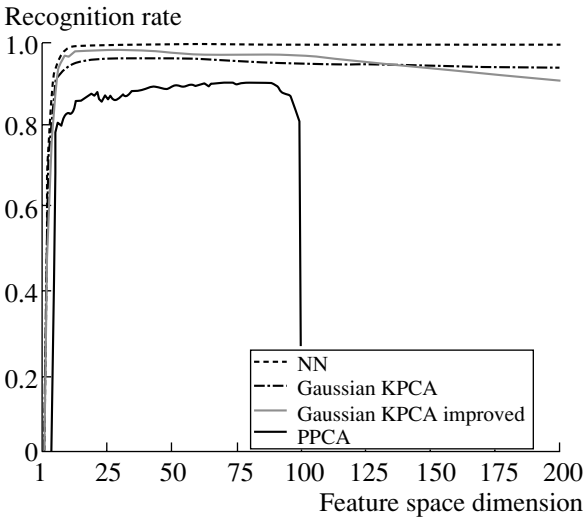

Fig. 4. Recognition rate at the first hierarchy level using the test set.

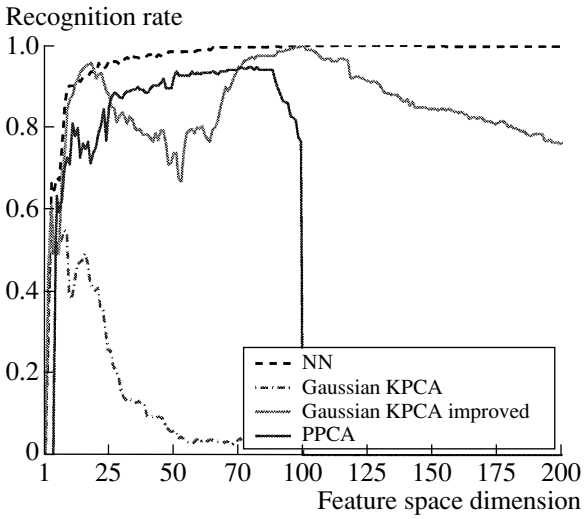

Fig. 5. Recognition rate at the first hierarchy level using the generic test set.

KPCA approach. Figure 3 visualizes this effect. Images of cans and cars are PCA-transformed to 2D vectors and displayed as black crosses (can) and white stars (car). The models of the different approaches are displayed by the area in which the vectors are located that would be assigned to a model. The *can* model and the *car* models are colored white and black, respectively. The gray area in between represents the rejection class. With the illustrative examples of the black car model, the mentioned problems can be shown. The KPCA model at the top left corner overfit to the trained data, whereas the KPCA model with the larger variance of the Gaussian kernel function (5) generalizes well. Also, the PPCA approach build a well formed model although some training images are rejected. The NN approach tends to model the borders between models very precisely and to spread in the other directions.

The progression of the curves of recognition rates depending on the eigenspace dimension (Fig. 4) shows that the PPCA model breaks down at about 100 dimensions, which is due to curse of dimensionality while building the model which is described in [9]. The NN and the KPCA model achieve a high recognition rate on the test set with 15 dimensions and more. On the generic test set (Fig. 5), the NN and the PPCA model with 70 dimensions demonstrate good results while the KPCA model breaks down. This is caused by a too specialized model that does not generalize well for generic object classification.

Regarding the computation time (Table 2), we determine a linear increase for the PPCA and the NN approach and an almost constant computation time for the KPCA approach. But for 50 dimensions, the computation time of the NN approach is 26 times faster than the KPCA approach and the PPCA approach is even 400 times faster.

We also checked the robustness of the NN approach by disturbing the image sets with Gaussian and pixel noise and calculating the recognition rates (Table 3). We determined no significant degradation in recogni-

Table 1. Best recognition rates (in %) with different nonlinear approaches using different test sample sets

Approach \ Set	training	test	generic test
PPCA	91.5%	90.4%	94.8%
NN	100%	99.9%	100%
KPCA	100%	96.4%	54.9%
KPCA improved	100%	98.6%	100%

Table 2. Mean computation time per test image (in s) on 50, 99, and 200 eigenspace dimensions

Approach \ Dimension	50	99	200
PPCA	3.61×10^{-4}	7.36×10^{-4}	–
NN	5.52×10^{-3}	1.35×10^{-2}	3.21×10^{-2}
KPCA	1.45×10^{-1}	1.51×10^{-1}	1.58×10^{-1}

Table 3. Maximum degradation of recognition rate (in %) due to Gaussian or pixel noise using the NN approach (using eigenspace dimensions ≥10)

Noise \ Set	training	test	generic test
Gaussian	0.00%	0.26%	0.69%
pixel	0.52%	1.00%	2.08%

tion rate when used more than ten feature dimensions. The best global generic recognition rate of 97.1% is achieved by the NN approach.

5. CONCLUSIONS

The objective of generic object recognition is to classify unknown objects in general categories and known ones in their specific classes. The result shows that the supervised generic object recognition is suitable for generalizing from learned examples of categories, even though the preprocessing of the images should be further improved. We suggested three methods for representing the generic class models: KPCA, NN, and PPCA.

As was shown, the KPCA approach copes very well with known objects. If the variance is too small, the model overfits to the trained objects. To generalize to unknown generic objects, the optimal variance parameter σ should be well chosen. An optimally chosen parameter leads to very good classification results for the KPCA approach. Unfortunately, the method depends on the number of training images; therefore, this approach is not practical for large databases.

Also, the NN approach performs excellent. The generalization is good and robust against noise as shown in Section 4. However, this approach has the same disadvantages as the KPCA approach: it depends on the number of training images, because all of them must be stored. Quantization of training vectors or leaving out unimportant vectors may solve this problem.

The classification rate of PPCA is worse than that of the other methods. This can be improved by using models with higher feature space dimensions and more submodels. For both, we need enough training vectors to obtain a suitable model. The limited number of images per object of the COIL-100 database prevent such an improvement. The approach generalizes well and is the fastest among examined. It is ten times faster than the NN approach and does not depend on the number of training images. The model size and the computational time depend only linearly on the number of objects; thus, it can handle large databases. Additionally, this method provides a probability measure.

ACKNOWLEDGMENTS

We thank C. Schnörr for stimulating discussions and for pointing us to density estimation methods in Kernel space.

REFERENCES

1. D. Koubaroulis, J. Matas, and J. Kittler, "Illumination Invariant Object Recognition Using the MNS Method," in *Proc. of the 10th European Signal Processing Conf., 2000*, pp. 2173–2176.
2. M. Reinhold, D. Paulus, and H. Niemann, "Appearance-Based Statistical Object Recognition by Heterogeneous Background and Occlusions," in *Proc. of the 23rd DAGM Symp. of Pattern Recognition, 2001*, pp. 254–261.
3. Q. Iqbal and J. K. Aggarwal, "Retrieval by Classification of Images Containing Large Manmade Objects Using Perceptual Grouping," Pattern Recognit. J. **35** (7), 1463–1479 (2002).
4. G. Medioni and A. Francois, "3D Structures for Generic Object Recognition," Computer Vision Image Anal. **1**, 30–37 (2000).
5. R. C. Nelson and A. Selinger, "A Cubist Approach to Object recognition," in *Proc. of the 6th Int. Conf. on Computer Vision (ICCV), 1998*, pp. 614–621.
6. S. Dickinson, R. Bergevin, I. Biederman, *et al.*, "The Potential of Geons for Generic 3D Object Recognition: Panel Report," Image Vision Comp. **15** (4), 277–292 (1997).
7. M. E. Tipping and C. M. Bishop, "Mixtures of Probabilistic Principal Component Analyzers," Neural Comp. **11** (2), 443–482 (1999).
8. T. M. Cover and P. E. Hart, "Nearest Neighbor Pattern Classification," IEEE Trans. Inform. Theory **13**, 21–27 (1967).
9. C. Drexler, F. Mattern, and J. Denzler, "Appearance-Based Generic Object Modeling and Recognition Using Probabilistic Principal Component Analysis," in *Proc. of the 24th DAGM Symp. of Pattern Recognition, 2002*, pp. 100–108.
10. C. Drexler, F. Mattern, and J. Denzler, "Generic Hierarchic Object Models and Classification Based on Probabilistic PCA," in *IAPR Workshop on Machine Vision Applications, 2002*, pp. 435–438.
11. R. Fisher, "The Use of Multiple Measures in Taxonomic Problems," Ann. Eugen. **7**, 179–188 (1936).
12. Z. Ghahramani and G. Hinton, "The EM Algorithm for Mixtures of Factor Analyzers," Technical Report, (Department of Computer Science, University of Toronto, 1997).
13. B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," Neural Comp. **10**, 1299–1319 (1998).
14. D. Cremers, T. Kohlberger, and C. Schnörr, "Nonlinear Shape Statistics in Mumford-Shah Based Segmentation," in *European Conf. on Computer Vision, 2002*, pp. 93–108.
15. S. A. Nene, S. K. Nayar, and H. Murase, "Columbia Object Image Library (COIL-100)," Technical Report No. CUCS-006-96 (Department of Computer Science, Columbia University, 1996).

Frank W. Mattern. Born 1976. Graduated from the University of Erlangen in 2001. Currently, is working as a scientific assistant at the Computer Vision Group, Faculty of Mathematics and Computer Science, University of Passau. His research interests are focused on generic object recognition and computer vision for autonomous systems. Author or coauthor of four scientific articles.

Joachim Denzler. Born 1967. Obtained his Diplom-Informatiker and Dr.-Ing. degrees from the University of Erlangen in 1992 and 1997, respectively. Currently, Professor for computer science and head of the Computer Vision Group, Faculty of Mathematics and Computer Science, University of Passau. His research interests comprise active computer vision, object recognition and tracking, 3D reconstruction and plenoptic modeling as well as computer vision for autonomous systems. Author and coauthor of over 70 journal papers and technical articles. Member of the IEEE, IEEE Computer Society, and GI. Was a recipient of the DAGM best paper awards in 1996, 1999, and 2001, respectively.

SPELL: 1. eigenspace, 2. q-dimensional, 3. overfit, 4. overfits, 5. submodels