IMPROVING STATISTICAL OBJECT RECOGNITION APPROACHES BY A PARAMETERIZATION OF NORMAL DISTRIBUTIONS

Ch. Gräßl, F. Deinzer, F. W. Mattern, H. Niemann

- DRAFT VERSION -

Christoph Gräßl
Chair for Pattern Recognition, University of Erlangen-Nuremberg
91058 Erlangen, Germany
phone: +49-9131-8527799, fax: +49-9131-303811
email: graessl@informatik.uni-erlangen.de

Frank Deinzer
Chair for Pattern Recognition, University of Erlangen-Nuremberg
91058 Erlangen, Germany
phone: +49-9131-8527874, fax: +49-9131-303811
email: deinzer@informatik.uni-erlangen.de

Frank W. Mattern
Computer Vision Group, FMI, Universität Passau, 94032 Passau, Germany
phone: +49-851-509-3009, fax: +49-851-3014
email: mattern@fmi.uni-passau.de

Heinrich Niemann
Chair for Pattern Recognition, University of Erlangen-Nuremberg
91058 Erlangen, Germany
phone: +49-9131-8527775, fax: +49-9131-303811
email: niemann@informatik.uni-erlangen.de

Y. Zhuravlev: Pattern Recognition and Image Analysis, 2004, Volume 14, Number 2,
IAPC Nauka / Interperiodica

Final version can be retrieved by sending an email to
graessl@informatik.uni-erlangen.de

IMPROVING STATISTICAL OBJECT RECOGNITION APPROACHES BY A PARAMETERIZATION OF NORMAL DISTRIBUTIONS

Ch. Gräßl[1], F. Deinzer[2], F. Mattern[3] and H. Niemann

Lehrstuhl für Mustererkennung, Universität Erlangen-Nürnberg,

Martensstraße 3, 91058 Erlangen, Germany

{graessl,deinzer}@informatik.uni-erlangen.de

Abstract. As statistical approaches play an important role in object recognition, we present a novel approach which is based on object models consisting of normal distributions for each training image. We show how to parameterize the mean vector and covariance matrix independently from the interpolation technique and formulate the classification and localization as a continuous optimization problem. This enables the computation of object poses which have never been seen during training. For interpolation, we present four different techniques which are compared in an experiment with real images. The results show the benefits of our method both in classification rate and pose estimation accuracy.

1 Introduction

Within the wide area of computer vision, object recognition is still one of the main topics of current research. Approaches for object recognition can mainly be divided into two directions. Firstly, segmentation based techniques which

[1]This work was partially funded by the European Commission 5th IST Programme - Project VAMPIRE

[2]This work was partially funded by the German Science Foundation (DFG) under grant SFB 603/TP B2. Only the authors are responsible for the content.

[3]This work was partially funded by the German Science Foundation (DFG) under grant DE 735/2-1. Only the authors are responsible for the content.

detect for example, geometric features can be used for object recognition [8]. But segmentation approaches suffer from the disadvantage that segmentation errors may occur which disturb the recognition process and from the general problem that often significant information is lost.

The second direction is that of appearance based approaches [13, 2, 10, 3, 12]. They avoid these disadvantages since they directly use the image data, e.g. pixel intensities, for the recognition process. There exist some well known approaches which uses multi-resolution wavelet features [13], Gaussian mixtures for classification [2] and the eigenspace approach [10] which was extended with a statistical component in [3, 12]. One primary disadvantage of most of those approaches is, that only those poses of objects are known that have been seen during the training process. The reason is that for each known pose a model feature is generated that represents exactly this one pose. That means for the statistical eigenspace approach that only poses can be computed which have been seen during the training.

This article discusses a general technique to parameterize the model features. From the application's point of view there is no longer a large collection of discrete model features — one for each known pose — but a continuous function that is able to provide a model feature for any desired pose. Our approach shows how one can efficiently parameterize the normal distributions that underlie the model features in a very general way. One major aspect of the discussion will be how to retain the necessary properties of normal distributions, like the positive definiteness of the covariance matrix. Based on this general parameterization technique we will show that a lot of different interpolation techniques can be used to implement that parameterization. The integration of the interpolation methods is thereby very easy as the parameterization does not make assumptions, for example, about the number of dimensions of the underlying pose space. Finally we will use this mathematical method to extend the statistical eigenspace to allow for an improved recognition and continuous localization.

Our experiments will show that our approach dramatically improves object recognition especially for the estimation of the pose of an object. We will also show and compare the different interpolation techniques and will discuss their major advantages and disadvantages with respect to runtime, memory consumption and recognition aspects.

2 The Discrete Statistical Eigenspace Approach

Traditionally, most object recognition methods do not work directly on pixel intensities of an image $\boldsymbol{f}$, but calculate features. One advantage of this technique is that the amount of data can be reduced drastically. Using graylevel images with a size of 256 x 256 pixels ($\hat{=}$65536 intensity values), the amount of data can be reduced to typically less than 20 feature values. Furthermore, features are normally more significant than the raw intensity values.

A very popular feature based method is the so-called *eigenspace approach* based on [10], which uses a *Karhunen-Loeve Transformation* [9] (also known as *Principal Components Analysis*) to obtain a linear system for computing a feature vector by

$$\boldsymbol{c} = \boldsymbol{\Phi}(\boldsymbol{f} - \overline{\boldsymbol{f}}), \quad \boldsymbol{\Phi} \in \mathbb{R}^{N_E \times N_P}. \tag{1}$$

The vector $\boldsymbol{f}$ contains all N_P intensities of the pixels of the object image. The matrix $\boldsymbol{\Phi}$ contains N_E eigenvectors (with the largest eigenvalues) of the covariance matrix of the training images. The average of all training images is denoted as $\overline{\boldsymbol{f}}$. Without loss of generality, we assume that the average image is subtracted from the object image in advance.

After computation of a feature vector using equation (1) it is possible to reconstruct the object image by

$$\hat{\boldsymbol{f}} = \boldsymbol{\Phi}^{+}\boldsymbol{c} \tag{2}$$

using the matrix pseudo inverse $\boldsymbol{\Phi}^+$ [17]. As the eigenspace matrix has the property of orthogonality, it is possible to use (2) to obtain a measurement of similarity. Using a distance measurement $d(\cdot,\cdot)$ (typically the L^1 or the L^2 norm is used) the relationship of similarity can be described as

$$\begin{aligned} d(\boldsymbol{f},\boldsymbol{f}') &\approx d(\hat{\boldsymbol{f}},\hat{\boldsymbol{f}}') \\ &= d(\boldsymbol{\Phi}^+\boldsymbol{c},\boldsymbol{\Phi}^+\boldsymbol{c}') \\ &= d(\boldsymbol{c},\boldsymbol{c}'). \end{aligned} \tag{3}$$

In Murase's traditional eigenspace approach, an object model is created using all training images of the objects. At first, for every object Ω_κ, all images $\boldsymbol{f}^{i,\kappa}$ from different viewpoints are used to calculate a single eigenspace matrix. Thus, the features $\boldsymbol{c}^{i,\kappa}$ can be calculated by equation (1). As the corresponding external pose parameter $\boldsymbol{\phi}_{i,\kappa}$ known for every image, classification and localization of an unknown object image $\boldsymbol{f}$ can be performed (using the similarity relationship of equation (3)) by

$$\begin{aligned} \boldsymbol{c} &= \boldsymbol{\Phi}\boldsymbol{f} \\ (\kappa^*, i^*) &= \operatorname*{argmin}_{(\kappa,i)} d(\boldsymbol{c},\boldsymbol{c}^{i,\kappa}) \\ \boldsymbol{\phi}^* &= \boldsymbol{\phi}_{i^*,\kappa^*}, \end{aligned} \tag{4}$$

where the result is class Ω_{κ^*} and pose $\boldsymbol{\phi}^*$. In figure 1 the first three components of the feature vectors of an object are shown. The distance measurement for an object image to be classified is also exemplified.

Fig. 1

An analysis of the feature vectors of noisy object images shows that the first components of $\boldsymbol{c}$ are less dispersed than the last ones. This is because the eigenvectors used for the eigenspace matrix $\boldsymbol{\Phi}$ are sorted (descending dependent on the corresponding eigenvalue). This means that the first eigenvectors represent significant features of the object and the last ones describe less significant details and noise. However, all components have the same influence on the distance measurement in (3).

A more robust classification was presented in [13] where $\boldsymbol{c}_\kappa^i$ is replaced by a normal distribution

$$p(\boldsymbol{c}|\boldsymbol{B}_\kappa^i) = \mathcal{N}(\boldsymbol{c}|\boldsymbol{\mu}_\kappa^i, \boldsymbol{\Sigma}_\kappa^i), \tag{5}$$

where $\boldsymbol{\mu}_{\kappa,i}$ denotes the mean vector and $\boldsymbol{\Sigma}_{\kappa,i}$ the covariance matrix. These two components build a *statistical model* $\boldsymbol{B}_{\kappa,i} = (\boldsymbol{\mu}_{\kappa,i}, \boldsymbol{\Sigma}_{\kappa,i})$ which is estimated by adding noise to the training image $\boldsymbol{f}^{i,\kappa}$ n-times. The result of the noise-adding processes are the new training images

$$\boldsymbol{f}^{i_1,\kappa}, \boldsymbol{f}^{i_2,\kappa}, \ldots \boldsymbol{f}^{i_n,\kappa}$$

and the corresponding feature vectors

$$\begin{aligned} \boldsymbol{c}^{i_1,\kappa} &= \boldsymbol{\Phi}\boldsymbol{f}^{i_1,\kappa} \\ \boldsymbol{c}^{i_2,\kappa} &= \boldsymbol{\Phi}\boldsymbol{f}^{i_2,\kappa} \\ &\vdots \\ \boldsymbol{c}^{i_n,\kappa} &= \boldsymbol{\Phi}\boldsymbol{f}^{i_n,\kappa} \end{aligned}$$

which are used to estimate the mean vector $\boldsymbol{\mu}_{\kappa,i}$ and covariance matrix $\boldsymbol{\Sigma}_{\kappa,i}$. Classification and pose estimation can be done in a similar way as in (4), but instead of the usage of the distance measurements $d(\cdot,\cdot)$, the probability (5) of the feature of the test image describes the similarity to a training image, which can be formulated as

$$\begin{aligned} \boldsymbol{c} &= \boldsymbol{\Phi}\boldsymbol{f} \\ (\kappa^*, i^*) &= \operatorname*{argmax}_{(\kappa,i)} p(\boldsymbol{c}|\boldsymbol{B}_{i,\kappa}) \\ \boldsymbol{\phi}^* &= \boldsymbol{\phi}_{i^*,\kappa^*}. \end{aligned} \tag{6}$$

3 Parameterization of Normal Distributions

The last section described the extension of the classical eigenspace approach by an statistical object model, which solves some of its problems. Nevertheless the poses, which can be estimated from the classifier, are restricted to poses of the training images. This means that exactly the pose of the most likely training image is returned, which leads to a systematical pose estimation error. As similar object images have similar features in feature space, [10] interpolates additional features using B-Splines. This is comprehensible as the features shown in figure 1 lead to the assumption that the curve in the feature space is continuous.

Assuming that the estimated normal distributions of similar object images are also similar, these considerations are also applicable to the statistical eigenspace model. In figure 2 components of the mean vector and the covariance matrix of the object images of one class dependent on the pose parameter are shown, which verifies this assumption. Interpolation of some additional normal distributions in an offline step would improve the resolution of pose estimation of the statistical eigenspace classifier. The disadvantage of this method is the high demand of memory to store the additional normal distributions. Also the usage of more features still shows the same systematical pose estimation errors as only poses of training images and interpolated normal distributions can be returned. We propose the parameterization of the normal distribution

$$\boldsymbol{B}(\kappa, \boldsymbol{\phi}) = (\boldsymbol{\mu}(\kappa, \boldsymbol{\phi}), \boldsymbol{\Sigma}(\kappa, \boldsymbol{\phi})) \tag{7}$$

where κ describes the discrete class number and $\boldsymbol{\phi}$ denotes the continuous pose parameter. Since in equation (6) it is assumed that a discrete number of normal distributions are available, classification and pose estimation has now to be formulated as an optimization problem

$$\boldsymbol{c} = \boldsymbol{\Phi f}$$

$$(\kappa^*, \boldsymbol{\phi}^*) \quad = \quad \underset{(\kappa, \boldsymbol{\phi})}{\operatorname{argmax}}\, p\left(\boldsymbol{c} | \boldsymbol{B}(\kappa, \boldsymbol{\phi})\right). \tag{8}$$

The optimization is done in two steps, which is typical for such problems [11]. First the optimal pose parameter for every class is estimated. Then the class with the best result determines the overall result of the classification process. More formal, this can be described as

$$\begin{aligned} \boldsymbol{c} &= \Phi \boldsymbol{f} \\ \boldsymbol{\phi}^*(\kappa) &= \underset{(\kappa, \boldsymbol{\phi})}{\operatorname{argmax}}\, p\left(\boldsymbol{c} | \boldsymbol{B}(\kappa, \boldsymbol{\phi})\right) \\ \kappa^* &= \underset{\kappa}{\operatorname{argmax}}\, \boldsymbol{\phi}^*(\kappa). \end{aligned} \tag{9}$$

Fig. 2

It has been shown in figure 2, that the components of the mean vector and covariance matrix, which are dependent on the pose parameter, are continuous. Based on this fact, the parameterization is reduced to the interpolation of individual components. So the mean vector can be interpolated by

$$\boldsymbol{\mu}(\kappa, \boldsymbol{\phi}) = \begin{pmatrix} \mu_1(\kappa, \boldsymbol{\phi}) \\ \mu_2(\kappa, \boldsymbol{\phi}) \\ \vdots \\ \mu_{N_E}(\kappa, \boldsymbol{\phi}) \end{pmatrix}. \tag{10}$$

An obvious idea for interpolation of the covariance matrix is using

$$\boldsymbol{\Sigma}(\kappa, \boldsymbol{\phi}) = \begin{pmatrix} \Sigma_{1,1}(\kappa, \boldsymbol{\phi}) & \Sigma_{1,2}(\kappa, \boldsymbol{\phi}) & \cdots & \Sigma_{1,N_E}(\kappa, \boldsymbol{\phi}) \\ \Sigma_{2,1}(\kappa, \boldsymbol{\phi}) & \Sigma_{2,2}(\kappa, \boldsymbol{\phi}) & \cdots & \Sigma_{2,N_E}(\kappa, \boldsymbol{\phi}) \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{N_E,1}(\kappa, \boldsymbol{\phi}) & \Sigma_{N_E,2}(\kappa, \boldsymbol{\phi}) & \cdots & \Sigma_{N_E,N_E}(\kappa, \boldsymbol{\phi}) \end{pmatrix}, \tag{11}$$

which means that all components of $\boldsymbol{\Sigma}(\kappa, \boldsymbol{\phi})$ are interpolated independently from each other. This is not permitted, as the vital postulation of *positive definiteness*

$$\boldsymbol{x}^T \boldsymbol{\Sigma} \boldsymbol{x} > 0 \;\; \forall \boldsymbol{x} \neq 0 \tag{12}$$

may be violated. So two other ways of parameterization are suggested:

- **Discrete interpolation:** Use the covariance matrix of the corresponding training image whose pose $\boldsymbol{\phi}_{i,\kappa}$ has the lowest distance. This approach is maintainable, because the influence of the mean vector on the classification rate is more important.

- **Cholesky factorization:** Every positive definite matrix $\boldsymbol{\Sigma}$ can be factorized by $\boldsymbol{\Sigma} = \boldsymbol{L}\boldsymbol{L}^T$. The matrix $\boldsymbol{L}$ is a lower triangular matrix, which can be parameterized componentwise and is written as

$$\boldsymbol{L}(\kappa, \boldsymbol{\phi}) = \begin{pmatrix} L_{1,1}(\kappa, \boldsymbol{\phi}) & 0 & 0 & \cdots & 0 \\ L_{2,1}(\kappa, \boldsymbol{\phi}) & L_{2,2}(\kappa, \boldsymbol{\phi}) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ L_{N_E,1}(\kappa, \boldsymbol{\phi}) & L_{N_E,2}(\kappa, \boldsymbol{\phi}) & \cdots & \cdots & L_{N_E,N_E}(\kappa, \boldsymbol{\phi}) \end{pmatrix} . \quad (13)$$

The parameterized covariance matrix can be calculated by

$$\boldsymbol{\Sigma}(\kappa, \boldsymbol{\phi}) = \boldsymbol{L}(\kappa, \boldsymbol{\phi}) \left(\boldsymbol{L}(\kappa, \boldsymbol{\phi})\right)^T . \quad (14)$$

The product of a matrix and its transponent is always positive definite which can easily be shown by

$$\begin{aligned} \boldsymbol{x}^T \boldsymbol{\Sigma}(\kappa, \boldsymbol{\phi}) \boldsymbol{x} &= \boldsymbol{x}^T \boldsymbol{L}(\kappa, \boldsymbol{\phi}) (\boldsymbol{L}(\kappa, \boldsymbol{\phi}))^T \boldsymbol{x} \\ &= \underbrace{(\boldsymbol{x}^T \boldsymbol{L}(\kappa, \boldsymbol{\phi}))}_{\boldsymbol{v}^T} \underbrace{((\boldsymbol{L}(\kappa, \boldsymbol{\phi}))^T \boldsymbol{x})}_{\boldsymbol{v}} \\ &> 0 \ \ \forall \boldsymbol{x} \neq \boldsymbol{0}, \boldsymbol{L}(\kappa, \boldsymbol{\phi}) \neq \boldsymbol{0} \end{aligned}$$

Algorithms for calculating $\boldsymbol{L}$ are given in [6].

4 Interpolation Techniques

In the last section, a method for parameterization of normal distributions which is independent from the interpolation technique was presented. There exist many methods for interpolating data. In the first part of this section, we restrict ourselves to pose parameters which lie on a regular grid. In the second part we also show two methods, based on trilinear interpolation and radial basis functions, which allow interpolation with scattered pose parameters.

4.1 Interpolation Techniques on Gridded Data

We use two 1-D interpolation methods for the components of the mean vector and the left triangle matrix and extend them to a n-D interpolation: Linear interpolation and *Catmull-Rom* spline (CRS) [1] interpolation. The linear interpolation is very fast but not continuously differentiable. The CRS interpolation is also fast because of its polynomial character and is continuously differentiable. A continuously differentiable interpolant is more realistic than one which is not. In contrast to other interpolation techniques, one does not have to solve a linear system, which makes CRS interpolation very flexible. In figure 3, two graphs of the first three components of $\boldsymbol{\mu}(\kappa = 1, \boldsymbol{\phi})$ using linear (left) and CRS interpolation (right) have been interpolated.

Fig. 3

The 1-D interpolation can be easily enhanced to n dimensions by using a dimension-descent technique as shown in figure 4. The upper left picture shows the dimension-descent for 2-D pose parameters for the linear interpolation. The filled points are the pose parameters of training images and the point *X* is to be interpolated. To do this by a linear 1-D interpolation, point p_1' has to be interpolated from p_1 and p_2. Similarly, point p_2' can be interpolated from p_3 and p_4. Now it is possible to interpolate the searched point *X* by using p_1' and p_2'. The upper right picture of figure 4 shows the dimension-descent for 3-D pose parameters. Of course, the dimension-descent can be adapted to CRS interpolation, which is exemplified (for 2-D interpolation) by the lower image.

Fig. 4

4.2 Interpolation Techniques on Scattered Data

Interpolation of scattered data is an intensively researched field in the area of computer graphics. Most of those techniques require a two dimensional triangle net for the parameterization which means that for our purpose we are limited to a 2-D pose parameter space. It is important that the mesh generator does not create acute angled triangles which would lead to bad interpolation results. Therefore we use a *Delaunay refinement* [15] to restructure an arbitrary triangle net which improves the quality of interpolation. Two examples for triangulation are given in figure 5.

One interpolation technique which uses triangle nets is the trilinear interpolation. The interpolant is defined as

$$\mu_n(\kappa, \boldsymbol{\phi}) = u_j \mu_{j,\kappa,n} + u_k \mu_{k,\kappa,n} + u_l \mu_{l,\kappa,n} \tag{15}$$

where u_j, u_k, u_l are the barycentric coordinates of $\boldsymbol{\phi}$, which is in the interior of the triangle with corner $\boldsymbol{\phi}_{\kappa,j}, \boldsymbol{\phi}_{\kappa,k}, \boldsymbol{\phi}_{\kappa,l}$ and $\mu_{\kappa,j,n}, \mu_{\kappa,k,n}, \mu_{\kappa,l,n}$ denotes the nth-component of the mean vector which belongs to the corresponding corners. The elements of matrix $\boldsymbol{L}$ in (13) can be treated the same way as the components of the mean vector.

Fig. 5

Another interpolation technique for scattered data which is applicable in parameter spaces of arbitrary size is the interpolation with radial basis functions [4], which have been intensively researched in numerical mathematics. The interpolation rule is

$$\mu_n(\kappa, \boldsymbol{\phi}_{i,\kappa}) = \mu_{i,\kappa,n} \forall i, \kappa \tag{16}$$

and the interpolant is defined as

$$\mu_n(\kappa, \boldsymbol{\phi}_{i,\kappa}) = \sum_i w_{i,\kappa} h(d_E(\boldsymbol{\phi}, \boldsymbol{\phi}_{i,\kappa})), \tag{17}$$

where $h : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is the so-called radial basis function, $d_E(\cdot, \cdot)$ is a distance measurement (we use Euclidian distance), and $w_{i,\kappa,n}$ is a weighting coefficient. For details of the computation of $w_{i,\kappa}$ consult [4]. There exist dozens of different radial basis functions, for our purpose we use

$$h(x) = \exp\left(-\frac{x^2}{\tau}\right), \tag{18}$$

because it depends on only one free adjustment parameter. If the value of τ is small, the elements of the normal distributions which have a large distance to $\boldsymbol{\phi}$ have a higher influence in comparison to a large τ (illustrated in figure 6).

Fig. 6

Note that any other interpolation techniques can be used for our purpose and the four examples are used to show how to use the presented method of parametrization of normal distributions. Doubtless there exist other methods of interpolation which could bring better classification rates.

5 Experiments and Results

We performed experiments on the *DIROKOL* image database [14] which consists of 13 real objects (1860 training images and 1860 test images per object) with a resolution of 256 x 256 shown in Fig. 7. For the image acquisition a turntable and a robot arm were used, which allows images to be taken from a hemisphere. Also the objects are illuminated at three different lighting conditions. As three different illumination conditions have been used, we limit ourselves to one third of the image set (all with the same illumination condition). Experiments with the full set have already been presented in [7]. We use a *Linux PC* with an *Athlon XP* (1.68 GHz) processor and 1GB memory for the experiments. We used an eight dimensional eigenspace for the PCA, and for the optimization problem in equation (8) we applied an adaptive random search algorithm [16], followed by a simplex step.

Fig. 7

We compared classification rate and pose estimation accuracy of the linear, CRS, trilinear and RBF interpolation. All methods are tested with a discrete interpolation of the covariance matrix and Cholesky factorization (except for the trilinear interpolation, where only a discrete interpolation of $\mathbf{\Sigma}$ has been performed). For the RBF interpolation set the parameter of the Gaussian function $\tau = 100$. As the unit of the pose parameter of the DIROKOL database is *motor steps* [14] and not degree, the value of τ is here much larger than in figure 6. Experiments for using the discrete statistical eigenspace approach have also been done to show the improvement of the continuous model. The results, which are presented in table 1, show, except for the trilinear interpolation, that the continuous approach leads to better results than the discrete approach both in classification rate and pose estimation accuracy. The accuracy is given in the so called percentile 80 values which describe the maximal localization error if the classification is correct and only the 80% best localizations are taken into account. The objects *cup 1* and *cup 2* have been excluded from the calculation of the pose estimation error since there exist ambiguities. Disadvantageous is the high computational cost of the continuous method, because a lot of mean vectors and covariance matrices have to be interpolated for optimization of (8). Note that the classification time is linearly dependent on the number of classes. The best classification rate and pose estimation can be achieved by using using a Cholesky factorization for covariance matrix. Using a linear interpolation for the mean vector and a discrete interpolation for the covariance matrix also shows a very good result, but takes only $\sim$40% of the computation time. Furthermore, the results show that the trilinear interpolation is not well suited for the interpolation of normal distributions.

Tab. 1

6 Conclusion and Outlook

In this paper, we presented a novel approach for parameterization of normal distributions which is applicable to object recognition algorithms based on statistical normal distributed object models. As the componentwise parameterization of the covariance matrix is not allowed, because the positive definiteness may be lost, we proposed the usage of a Cholesky factorization. For interpolation we used Catmull-Rom splines, linear interpolation, radial basis functions and trilinear interpolation. The last two techniques are also applicable on object models where the pose parameters of the training images are scattered. Experiments which have been performed on an image database with real images show that the continuous model is superior to the discrete model. The advantage of radial basis functions is that they can also be used on scattered pose parameters of the training images.

Further research should concentrate on other methods for parameterization of normal distributions. Also interpolation techniques based on Beziér patches like the Clough-Tocher interpolation [5] should be evaluated. The usage of other radial basis functions like the inverse multiquadratic functions may be beneficial.

References

[1] E. Catmull and R. Rom. A Class of Local Interpolating Splines. *Computer Aided Geometric Design*, pages 317–326, 1974.

[2] J. Dahmen, R. Schlüter, and H. Ney. Discriminative Training of Gaussian Mixtures for Image Object Recognition. In *21. DAGM Symposium*

Mustererkennung, W. Förstner, J. Buhmann, A. Faber, P. Faber (eds.), pages 205–212, Bonn, Germany, September 1999.

[3] F. Deinzer, J. Denzler, and H. Niemann. On Fusion of Multiple Views for Active Object Recognition. In B. Radig and Stefan Florczyk, editors, *Mustererkennung 2001, 23. DAGM-Symposium, München*, pages 239–245, Berlin, September 2001. Springer.

[4] Nira Dyn and Charles A. Micchelli. Interpolation by sums of radial functions. *Numerische Mathematik, Springer-Verlag, Berlin*, 58(1):1–9, 1990.

[5] Gerald Farin. *Curves and Surfaces for Computer Aided Geometric Design*. Academic Press, Boston, 3 edition, 1993.

[6] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, Baltimore, MD, USA, 3rd edition, 1996.

[7] Ch. Gräßl, F. Deinzer, and H. Niemann. Continuous parametrization of normal distributions for improving the discrete statistical eigenspace approach for object recognition. In V. Krasnoproshin, S. Ablameyko, and J. Soldek, editors, *Pattern Recognition and Information Processing*, volume 1, pages 73–77, Minsk, 2003.

[8] J. Hornegger and H. Niemann. Statistical Learning, Localization, and Identification of Objects. In *Proceedings of the Fifth International Conference on Computer Vision (ICCV)*, pages 914–919, Boston, Massachusetts, USA, 1995. IEEE Computer Society Press.

[9] M. Loeve. *Probability Theory*. Springer, New York, 4th edition, 1978.

[10] H. Murase and S. Nayar. Visual Learning and Recognition of 3-D Objects from Appearance. *International Journal of Computer Vision*, 14:5–24, 1995.

[11] H. Niemann. *Pattern Analysis and Understanding, 2nd edition*. Springer-Verlag, Berlin, 1990.

[12] L. Paletta and A. Pinz. Active Object Recognition by View Integration and Reinforcement Learning. *Robotics and Autonomous Systems*, 31((1–2)):1–18, 2000.

[13] M. Reinhold, F. Deinzer, J. Denzler, D. Paulus, and J. Pösl. Active Appearance–Based Object Recognition Using Viewpoint Selection. In B. Girod, G. Greiner, H. Niemann, and H.-P. Seidel, editors, *VMV 2000*, pages 105–112. infix, Berlin, November 2000.

[14] M. Reinhold, Ch. Drexler, and H. Niemann. Image Database for 3-D Object Recognition. Technical Report LME-TR-2001-02, Informatik 5 (Lehrstuhl für Mustererkennung), Universität Erlangen-Nürnberg, 2001.

[15] J. R. Shewchuck. *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain*. Carnegie Mellon University, Pittsburgh, 1994.

[16] A. Törn and A. Žilinskas. Global Optimization. *Lecture Notes in Comuter Science, Springer-Verlag, Berlin*, 3350, 1989.

[17] L. Trefethen and D. Bau III. *Numerical Linear Algebra*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1997.

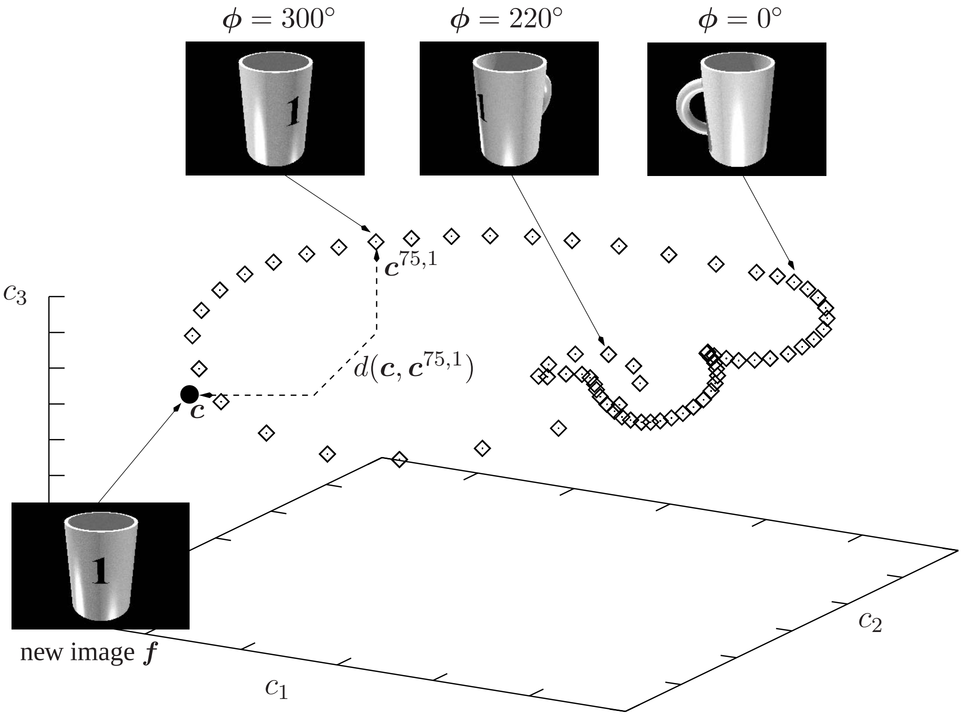


Fig. 1: Illustration of the first three components of feature vectors of one class and the distance measurement, which is used to classify and localize a new image

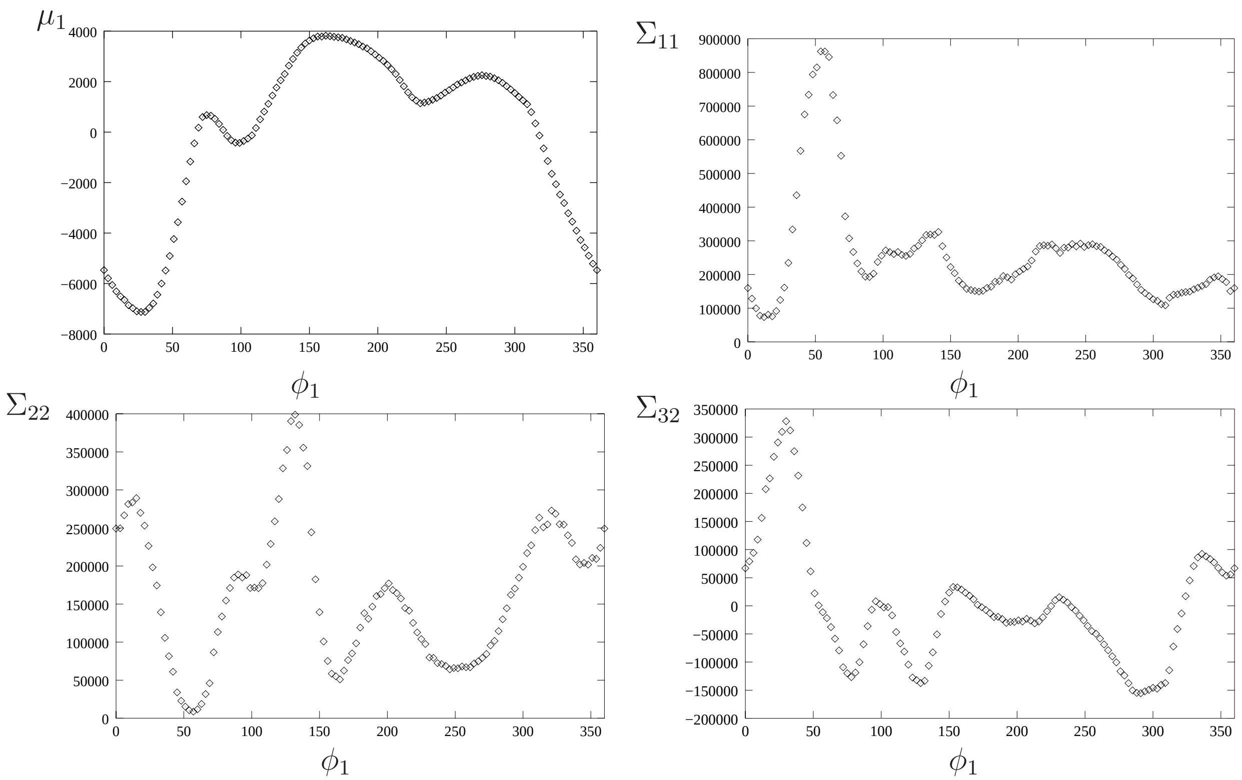


Fig. 2: The graph of elements of the mean vector $\boldsymbol{\mu}$ and of the covariance matrix $\boldsymbol{\Sigma}$. The parameter ϕ_1 describes the 1-D pose of an object which was placed on a turntable.

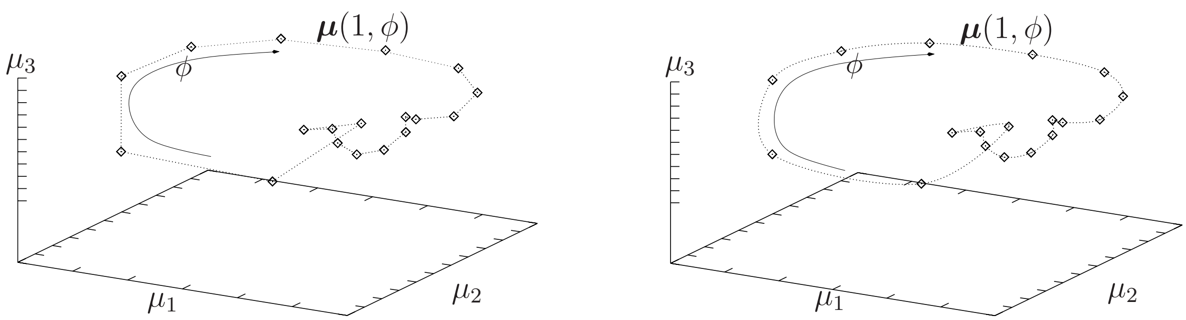
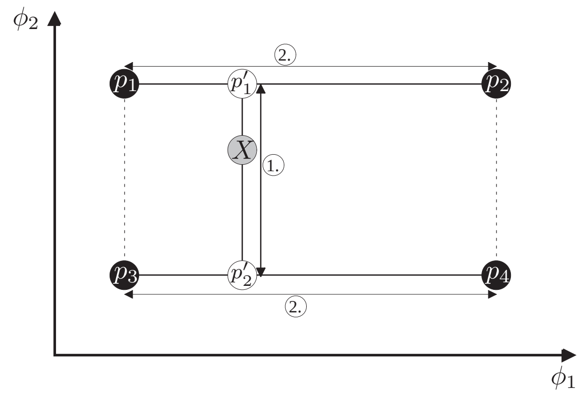
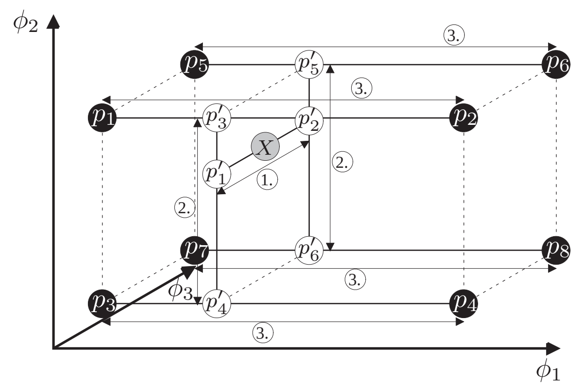


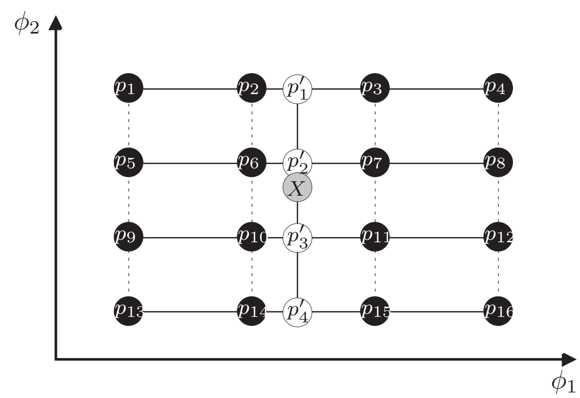
Fig. 3: Example of a linear (left) and CRS interpolation (right) of a graph of the first three components of $\boldsymbol{\mu}(\kappa = 1, \boldsymbol{\phi})$ dependent of a one dimensional pose parameter ϕ. A total of $N_T = 16$ training images – corresponding mean vectors are marked as diamonds – are the control points of the interpolation.



(a) Linear 2-D-Interpolation



(b) Linear 3-D-Interpolation



(c) CRS 2-D-Interpolation

Fig. 4: Illustration of a reduction of an n-dimensional to a one-dimensional interpolation. (a) and (b) show this recursive procedure in the case of a linear interpolation, (c) in the case of CRS Interpolation.

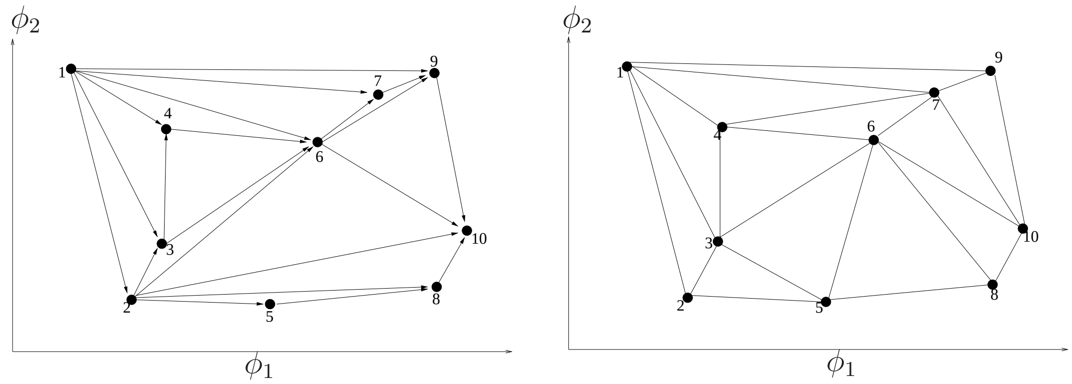
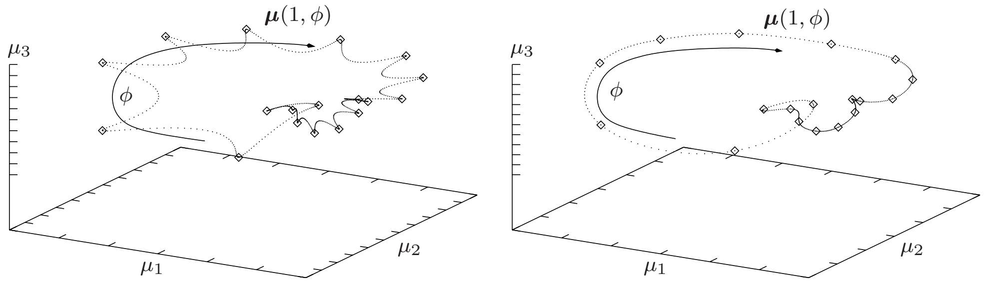


Fig. 5: Left: Triangle net using an awkward triangulation algorithm. Right: Triangle net with delaunay refinement [15] which is more suitable for interpolation. Dots mark pose parameters of training images.



(a) RBF-Interpolation with $\tau = 10$ (b) RBF-Interpolation with $\tau = 20$

Fig. 6: Example for an interpolation using radial basis functions (RBF) of a three-dimensional mean vector $\boldsymbol{\mu}(\kappa = 1, \boldsymbol{\phi})$*. A total of* $N_T = 16$ *training images – corresponding mean vectors are marked as diamonds – are the control points of the interpolation. The influence of the parameter* τ *of the Gaussian function* $K(x) = \exp(-x^2/\tau)$ *is clearly visible.*

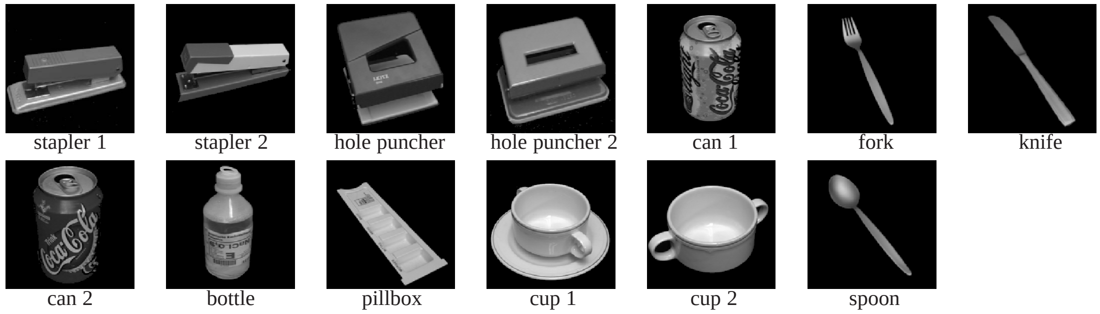


Fig. 7: The DIROKOL image database [14]

Method	**Classification rate**	**Pose estimation accuracy**	**Classification time**
discrete statistical	86.1%	$9.61°$	$< 0.01s$
$\boldsymbol{\mu}$: linear, $\boldsymbol{\Sigma}$: discrete	90.0%	$6.79°$	$1.6s$
$\boldsymbol{\mu}$: linear, $\boldsymbol{\Sigma}$: linear	92.6%	$6.18°$	$3.7s$
$\boldsymbol{\mu}$: CRS, $\boldsymbol{\Sigma}$: discrete	91.1%	$6.59°$	$3.4s$
$\boldsymbol{\mu}$: CRS, $\boldsymbol{\Sigma}$: CRS	92.5%	$6.00°$	$6.5s$
$\boldsymbol{\mu}$: trilinear, $\boldsymbol{\Sigma}$: discrete	75.1%	$45.28°$	$4.5s$
$\boldsymbol{\mu}$: RBF, $\boldsymbol{\Sigma}$: discrete, $\tau = 100$	90.5%	$6.06°$	$9.7s$
$\boldsymbol{\mu}$: RBF, $\boldsymbol{\Sigma}$: RBF, $\tau = 100$	92.5%	$6.00°$	$20.0s$

Table 1: Classification rate, accuracy of pose estimation (percentile 80 values) of successfully classified images and optimization time using the DIROKOL image database.

Christoph Gräßl, born 29 August 1975, got the degree 'Diplom-Informatiker' from the University of Erlangen in the year 2002. Currently, he is working as a scientific assistant at the chair for pattern recognition at the institute of computer science, University of Erlangen. His research interests is focused on object recognition and object tracking. He is author of 4 scientific articles. His work was funded by the European Commission 5th IST Programme - Project VAMPIRE.

Frank Deinzer, born 1 September 1972, got his degree 'Diplom-Informatiker' from the University of Erlangen in the year 1998. Currently, he is working as a scientific assistant at the chair for pattern recognition at the institute of computer science. His research interests is focused on active approaches for object recognition and statistical fusion of sensor data. He is author and co-author of 10 articles. His work on sensor fusion for active object recognition was awarded with DAGM best paper award in 2001.

Frank W. Mattern, born 15 March 1976, got the degree 'Diplom-Informatiker' from the University of Erlangen in the year 2001. Currently, he is working as a scientific assistant at the computer vision group, Faculty of Mathematics and Computing, University of Passau. His research interests is focused on generic object recognition and computer vision for autonomous systems. He is author or co-author of 4 scientific articles. His work was funded by the German Science Foundation (DFG) under grant DE 732/2-1.

Heinrich Niemann obtained the degree of Dipl.-Ing. in Electrical Engineering and Dr.-Ing. from Technical University Hannover, Germany. He worked with Fraunhofer Institut für Informationsverarbeitung in Technik und Biologie, Karlsruhe, and with Fachhochschule Giessen in the department of Electrical Engineering. Since 1975 he has been Professor of Computer Science at the University of Erlangen-Nürnberg, where he was dean of the engineering faculty of the university from 1979 – 1981. Since 1988 he is head of the research group "Knowledge Processing" at the Bavarian Research Institute for Knowledge Based Systems (FORWISS). Since 1998 he is speaker of a "special research area" (SFB) entitled "Model–Based Analysis and Visualization of Complex Scenes and Sensor Data" which is funded by the German Research Foundation (DFG). His fields of research are speech and image understanding and the application of artificial intelligence techniques in these fields. He is on the editorial board of Signal Processing, Pattern Recognition Letters, Pattern Recognition and Image Analysis, and Journal of Computing and Information Technology. He is the author or coauthor of 7 books and about 400 journal and conference contributions as well as editor or coeditor of 24 proceedings volumes and special issues. He is a member of DAGM, ISCA, EURASIP, GI, IEEE, and VDE and a Fellow of IAPR.