ARTICLE IN PRESS

PERGAMON

Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Pattern Recognition III (IIII) III–III

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Appearance-based recognition of 3-D objects by cluttered background and occlusions

Michael P. Reinhold*, Marcin Grzegorzek, Joachim Denzler, Heinrich Niemann

Chair for Pattern Recognition, University Erlangen-Nuremberg, Martensstr. 3, 91058 Erlangen, Germany

Received 30 September 2003; received in revised form 28 October 2004; accepted 28 October 2004

Abstract

In this article we present a new appearance-based approach for the classification and the localization of 3-D objects in complex scenes. A main problem for object recognition is that the size and the appearance of the objects in the image vary for 3-D transformations. For this reason, we model the region of the object in the image as well as the object features themselves as functions of these transformations. We integrate the model into a statistical framework, and so we can deal with noise and illumination changes. To handle heterogeneous background and occlusions, we introduce a background model and an assignment function. Thus, the object recognition system becomes robust, and a reliable distinction, which features belong to the object and which to the background, is possible. Experiments on three large data sets that contain rotations orthogonal to the image plane and scaling with together more than 100 000 images show that the approach is well suited for this task.

Keywords: Object recognition; Appearance-based; Object representation; Statistical modelling; Background model; 3-D transformation of objects

1. Introduction

For many tasks the recognition of objects in images is necessary, for example for visual inspection or for automatic detection of objects. In doing so, mostly the class as well as the pose of the object have to be estimated. One main aspect in object recognition is that the appearance as well as the size of the objects vary under 3-D transformations, i.e. scaling or rotations orthogonal to the image plane. An example is shown in Fig. 1. Therefore the appearance of the objects has to be stored for the different, possible viewpoints in a proper way. Especially the large data size has to be reduced. Furthermore, for real recognition tasks one has to deal with the following problems: often the illumination changes, the objects are situated in heterogeneous background and are partially occluded. A robust object recognition system has to handle these disturbances and has to guarantee a reliable recognition in spite of that.

1.1. Related work

There are two main approaches for object recognition. First, there exist approaches that apply a segmentation process and use geometric features like lines or vertices as features, e.g. Refs. [1–6]. But these methods suffer from segmentation errors, and they have problems to deal with objects that have no distinct edges. Therefore many authors, e.g. Refs. [7–14], prefer the second method, the appearance-based approach. Here, the features are directly calculated by the pixel intensities without a previous segmentation process.

* Corresponding author. Tel.: +49 9131 85 27775; fax: +49 89 4129 13055.
E-mail addresses: michael.p.reinhold@web.de (M.P. Reinhold), marcin.grzegorzek@informatik.uni-erlangen.de (M. Grzegorzek), joachim.denzler@uni-jena.de (J. Denzler), niemann@informatik.uni-erlangen.de (H. Niemann).

0031-3203/$30.00
doi:10.1016/j.patcog.2004.10.008

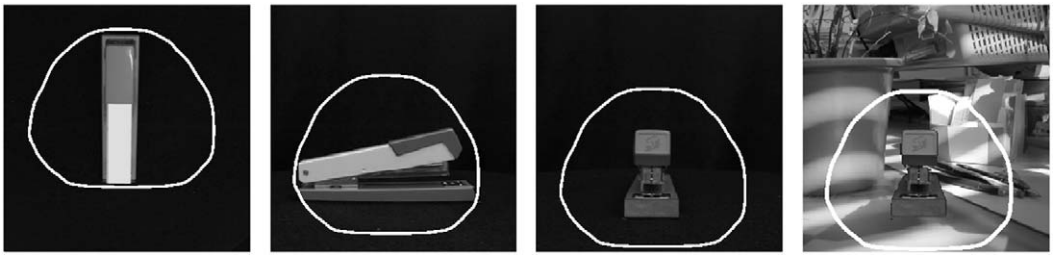

Fig. 1. Different viewpoints for a stapler. The fixed region that encloses the stapler for all viewpoints distributed on a hemisphere is plotted. For cluttered background a lot of background features are counted to the stapler.

The several appearance-based approaches differ in the way they handle 3-D transformations, in the following we will call these transformations *external transformations*. Some authors, who only want to classify objects, eliminate the pose information and model the varying features for example by *Gaussian mixtures* [10]. More often the so-called *view classes* are applied, e.g. Ref. [8]. Here adjacent viewpoints with similar appearance of the object are subsumed to one view class, and an object is represented by several view classes. Therefore, even for these methods the pose of the object cannot be estimated exactly. Only the respective view class can be estimated as for example for the *multidimensional reception field histograms* of Schiele and Crowley [8]. Besides, if the appearance of the object varies a lot due to the external transformations, many view classes are necessary. In contrast, there are only a few authors, e.g. Refs. [7,9,11,12,14], who model the appearance as well as the pose of an object. So, they can estimate the class and the pose of an object. The most famous method is the *parametric eigenspace* by Murase and Nayar [7]. Bischof et al. [14] improved its robustness for illumination changes by using gradient-based filters, Borotschnig et al. [9] and Gräßl et al. [12] extended it by a statistical framework.

However, for real environments one has to consider that the objects are often situated in cluttered background and are partially occluded. Then, both the features at the border of the object and of the occluded part of the object change. The features at the border of the object vary, because the features are mostly calculated from the pixels of a small local region. Simple models for the object cannot handle these problems. For this reason, some authors, e.g. Ref. [15] for the eigenspace approach, try to find n (out of the totally N) object features that are not affected. For the recognition they only consider these n features and disregard the other $N-n$ features. Since for this method there is the risk to confuse similar-looking objects, other authors, e.g. Refs. [11,16], consider all features and employ an explicit background model with an assignment. For this purpose, many authors, e.g. Refs. [11,16], use a priori knowledge about the background during recognition. This might be an advantage, if the background is known a priori and varies only less. But for the recognition of objects in arbitrary environments, these conditions are rarely fulfilled.

But the described approaches share all the same problem: they model the varying appearance of the object, but they do not take into account that also the size of the object in the image varies due to the external transformations. Mostly, they employ a fixed bounding box or fixed arbitrary formed region. They choose its size so that the object resides for all external transformations inside this region. Further, they define that all features inside this region belong to the object, as for example for the eigenspace approach. But, for many viewpoints this region is much bigger than the object and encloses plenty of background features, as one can see in the right image of Fig. 1. In this case, a reliable recognition is not possible, even if the background is modelled explicitly. If only that region in the image is chosen that belongs for all external transformations to the object, e.g. Ref. [17], this region might be too small for a reliable recognition. The use of view-classes can reduce the problem, but cannot solve it.

1.2. Our approach

We model both the appearance of the object—represented by local feature vectors derived by the multiresolution analysis—and also its size in the image (in the following called *bounding region*) as functions of the external transformations [18]. So, the bounding region encloses the object tightly for all external transformations. In doing so, as many object features and as few background features as possible are considered for the object. Therefore, even if the size of the object varies a lot due to the external transformations, a reliable recognition is possible.

To formulate the dependence on the external transformations, we approximate the bounding region and the object features by sums of weighted continuous *basis functions*. This representation has a lot of advantages: we can handle also viewpoints between the trained viewpoints. A pose estimation is possible. Finally, by the use of trigonometric functions as basis functions the data size can be reduced strongly. So, we can deal with external rotations as well as a scaling.

To make the system robust with respect to camera noise and illumination changes, we apply a statistical framework: the object features are modelled statistically by normal distributions and the objects are represented by density functions.

To deal with cluttered background and partial occlusions, we model the background explicitly by a uniform distribution. Further, we define an *assignment function* that assigns each local feature vector inside the bounding region either to the object or to the background. For the background model and the assignment function no a priori knowledge is necessary, and each possible background can be handled. By this framework, even for complex scenes, a reliable localization and classification is possible.

In the following section we present our object model for homogeneous background. Particularly, we describe, how we model the bounding region and the features as functions of the external transformations, and how we integrate them into a statistical framework. In Section 3, we outline our background model, and in Section 4 we present experiments on three databases that comprehend two and three external transformations. The experiments are performed on homogeneous as well as on heterogeneous background and by partial occlusion. Finally, we end with a summary and an outlook in Section 5.

2. Object model

In the following subsections, firstly we will explain the model for one object class. If there are several object classes—like for example for the classification in Section 2.6—for each object class the respective parameters have to be trained. In that case we will mark these parameters with the index κ for the class.

2.1. Features

In our approach, we employ local feature vectors and represent an object by a set of local features. The main advantage of local feature vectors is that a local disturbance, e.g. noise or occlusion, only affects the local feature vectors in a small region around it. All the other local feature vectors are unchanged. In contrast to this, a global feature vector can totally change, if only one pixel in the image varies.

For the calculation of these feature vectors, we lay a grid with the grid size $r_s = 2^s$, whereby s is the index for the scale, on the quadratic image $\boldsymbol{f}$, as one can see in the left image of Fig. 2. In the following we will summarize these grid locations as $X = \{\boldsymbol{x}_{\widetilde{m}}\}_{\widetilde{m}=0,\ldots,M-1}$, $\boldsymbol{x}_{\widetilde{m}} \in \mathbb{R}^2$. On each grid point $\boldsymbol{x}_{\widetilde{m}}$ a two-dimensional local feature vector $\boldsymbol{c}(\boldsymbol{x}_{\widetilde{m}})$ is calculated. For this purpose we perform, corresponding to the chosen resolution r_s, s-times the wavelet multiresolution analysis [19] (see Fig. 3) using Johnston 8-TAP wavelets [11]. The coefficients of the local feature vectors $\boldsymbol{c}(\boldsymbol{x}_{\widetilde{m}})$ are computed by

$$\boldsymbol{c}(\boldsymbol{x}_{\widetilde{m}}) = \boldsymbol{c}_{\widetilde{m}} = \begin{pmatrix} c_{\widetilde{m},1} \\ c_{\widetilde{m},2} \end{pmatrix} = \begin{pmatrix} \ln|b_{s,\widetilde{m}}| \\ \ln(|d_{0s,\widetilde{m}}| + |d_{1s,\widetilde{m}}| + |d_{2s,\widetilde{m}}|) \end{pmatrix}. \quad (1)$$

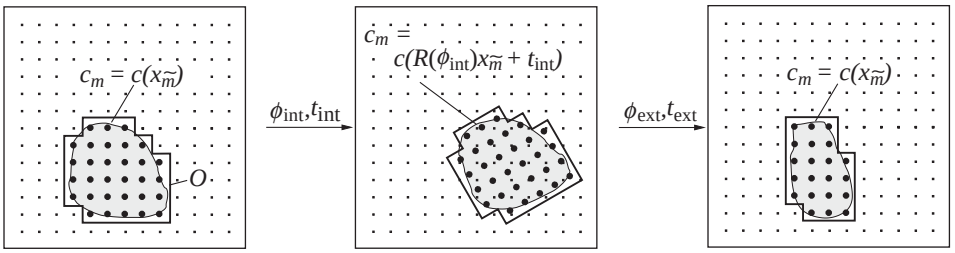

Fig. 2. Left: the image is covered by a grid for the local feature vectors $\boldsymbol{c}(\boldsymbol{x}_{\widetilde{m}})$, the bounding region O encloses the object tightly. Object grid and bounding region O for internal transformations ϕ_{int} and $\boldsymbol{t}_{\text{int}}$ (middle) and for external transformations $\boldsymbol{\phi}_{\text{ext}}$ and t_{ext} (right).

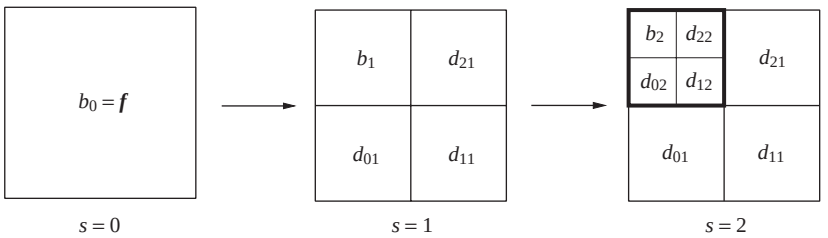

Fig. 3. An example of the wavelet multiresolution analysis. Here it is performed two times. Each time the left upper quadrant b_s that contains the low frequencies is filtered in a quadrant b_{s+1} with lower frequencies and three quadrants $d_{0\ldots2s+1}$ that contain the higher frequencies of b_s.

This means, the first component $c_{\tilde{m},1}$ of a local feature vector is derived by the low-pass coefficient of the wavelet transformation at the respective position $b_{s,\tilde{m}}$. The second component $c_{\tilde{m},2}$ is derived by the respective first high-pass values that contains information about discontinuities, e.g. edges. We disregard the higher frequencies; thus, the data size is reduced, and especially the noise that is mostly located at high frequencies is filtered out.

2.2. Object region in the image—bounding region O

For the object model we want to consider only those local feature vectors in the image that belong to the object and not to the background. Since the object normally takes only a small part of the whole, we define a tightly enclosing bounding region $O \subset X$. Subsequently, the N_O feature vectors inside this bounding region O are counted to the object. In the following they will be called *object feature vectors* $\boldsymbol{c}_{O,\overline{m}}$; the set of these object feature vectors is denoted as C_O. The training of the bounding region O will be described later in Section 2.5.

For the simpler case, when the object is only rotated by $\phi_{\text{int}} \in \mathbb{R}$ and translated by $\boldsymbol{t}_{\text{int}} \in \mathbb{R}^2$ inside the image plane, the appearance of the object does not change. For these transformations, in the following called *internal transformations*, the size of the bounding region O can be modelled as fixed and can be trained by one image of the object. The bounding region O is moved with the same transformations as the object itself (see image in the middle of Fig. 2). Also, the *object grid* inside the bounding region—marked by the bold points in Fig. 2—is transformed in the same way. The new positions $\boldsymbol{x}_m$ of the object grid are calculated by

$$\boldsymbol{x}_m = \boldsymbol{R}(\phi_{\text{int}})\boldsymbol{x}_{\tilde{m}} + \boldsymbol{t}_{\text{int}}, \tag{2}$$

whereby $\boldsymbol{R}(\phi_{\text{int}}) \in \mathbb{R}^{2\times 2}$ is the rotation matrix. If the positions $\boldsymbol{x}_m$ of the object grid do not coincide with the positions $\boldsymbol{x}_{\tilde{m}}$ of the image grid, the object feature vectors $\boldsymbol{c}_m$ on the transformed positions $\boldsymbol{x}_m$ are interpolated of the adjacent image feature vectors $\boldsymbol{c}_{\tilde{m}}$.

For the more difficult case, when the object is transformed by the external transformations $\boldsymbol{\phi}_{\text{ext}} \in \mathbb{R}^2$ and $t_{\text{ext}} \in \mathbb{R}$, the size of the object in the image varies; i.e. for some external transformations a feature vector $\boldsymbol{c}_m$ belongs to the object, for other external transformations it belongs to the background. Therefore, we model the size of the bounding region O as function of these external transformations (see right image in Fig. 2). Thus, it can be warranted that the bounded region encloses the object tightly for all transformations as postulated in the Introduction. To formulate this dependency mathematically, we define for each local feature vector $\boldsymbol{c}_m$ a function $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$. It assigns the feature vector $\boldsymbol{c}_m$ depending on the external transformations to the bounding region O, i.e. to the object, or to the background $X \backslash O$. These functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ are trained by images of the object for different viewpoints. To handle also viewpoints between the discrete training viewpoints and to reduce the data size, we model these functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ as continuous functions of N_ξ basis functions v_r with:

$$\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}}) = \sum_{r=0}^{N_\xi - 1} a_{\xi,m,r} v_r, \tag{3}$$

which will be explained in detail in Section 2.5.

Note, during the recognition phase the size of the bounding region O for a pose is calculated by these trained functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$. Therefore, no segmentation is necessary during the recognition. See Section 2.6.

2.3. Statistical model

To handle illumination changes and low-frequency noise, we interpret the local feature vectors $\boldsymbol{c}_m$ as random variables and apply a statistical model. First, we assume that the object feature vectors $\boldsymbol{c}_{O,\overline{m}}$ inside the bounding region O are statistically independent of the features vectors outside the bounding region. Therefore we can disregard the feature vectors outside the bounding region O for the object model. Further, we suppose that the single object feature vectors $\boldsymbol{c}_{O,\overline{m}}$ and their components are statistically independent and normally distributed. We decided for this simple model, although in reality neighboring object feature vectors $\boldsymbol{c}_{O,\overline{m}}$ might be statistically dependent. But considering the full neighborhood relationship, e.g. by a Markov Random Field, leads to a very complex model. Modelling a dependency between neighboring object feature vectors in a row [11] gave worse results than the assumption of statistical independence. Besides, by the statistical independence non-uniform illumination changes can be handled very well, for example when the direction of the lighting varies and some parts of the object get brighter, whereas on the same time other parts get darker.

Thus, an object can be described by the probability density p to observe the object features $\boldsymbol{c}_{O,\overline{m}}$:

$$\begin{aligned} p(C_O|\boldsymbol{B}, \boldsymbol{\phi}, \boldsymbol{t}) &= \prod_{\boldsymbol{x}_m \in O} p(\boldsymbol{c}_m|\boldsymbol{\mu}_m, \boldsymbol{\sigma}_m, \boldsymbol{\phi}, \boldsymbol{t}) \\ &= \prod_{\boldsymbol{x}_m \in O} \prod_{q=1,2} p(\boldsymbol{c}_m|\mu_{m,q}, \sigma_{m,q}, \boldsymbol{\phi}, \boldsymbol{t}), \end{aligned} \tag{4}$$

where $\boldsymbol{\phi} = (\boldsymbol{\phi}_{\text{ext}}, \phi_{\text{int}})^{\text{T}}$, $\boldsymbol{t} = (t_{\text{ext}}, \boldsymbol{t}_{\text{int}})^{\text{T}}$ and the parameter $\boldsymbol{B}$ comprehends the trained means $\boldsymbol{\mu}_m = (\mu_{m,q})_{q=1,2}$ and trained standard deviations $\boldsymbol{\sigma}_m = (\sigma_{m,q})_{q=1,2}$ of the components $c_{m,q}$ of the feature vectors. In the following $p(C_O|\boldsymbol{B}, \boldsymbol{\phi}, \boldsymbol{t})$ in Eq. (4) will be called *object density p*. Note: Because of the flexible size of the bounding region O, it depends on the external transformations $\boldsymbol{\phi}_{\text{ext}}$ and t_{ext}, which feature vectors are taken into account for the object density $p(C_O|\boldsymbol{B}, \boldsymbol{\phi}, \boldsymbol{t})$.

If there are only internal transformations, the means $\mu_{m,q}$ and standard deviations $\mu_{m,q}$ of the feature vectors $\boldsymbol{c}_m$ are constant, because the appearance of the object does not

change. But under external transformations the appearance, also the means $\mu_{m,q}$ vary. So, we model $\mu_{m,q}$ as functions of the external transformations: $\mu_{m,q} = \mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$. Similar to the functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$, they are trained by images from different viewpoints, and they are represented as sum of N_μ weighted continuous basis functions:

$$\mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}}) = \sum_{r=0}^{N_\mu - 1} a_{\mu,m,q,r} v_r. \tag{5}$$

In contrast to $\mu_{m,q}$ we model the standard deviations $\sigma_{m,q}$ as constant: for the chosen features the standard deviation $\sigma_{m,q}$ is approximately independent of the external transformations, when the brightness of the illumination changes uniformly. Also for other illumination changes, this assumption gives good results.

2.4. Modelling the external transformations

In the last two subsections we mentioned that we model the bounding region functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ (Eq. (3)) and the means $\mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ (Eq. (5)) as functions of the external transformations. In this subsection we will explain this in detail. Although we have only discrete viewpoints for the training, we model ξ_m and $\mu_{m,q}$ as continuous functions, because in reality they are continuous. So, we can handle viewpoints between the trained viewpoints and estimate the respective transformations exactly. Besides, the data size can be reduced, because we only need to store the coefficients of the approximation functions v_r and not the single views.

We apply trigonometric functions as basis functions v_r. The trigonometric functions are well established in function approximation even for several degrees of freedom [20]. By these functions we can model a scaling t_{ext} as well as rotations $\boldsymbol{\phi}_{\text{ext}}$. Also, a periodic rotation like a turntable rotation can be represented. Furthermore, they are approved in image compression and coding [21]. Therefore, even complex objects can be described by a low number of basis functions.

For a periodic transformation, like a 360° turntable rotation ϕ_{table}, we use the sine–cosine-decomposition. So, for the basis functions v_r of Eqs. (3) and (5) are (in this example for one external degree of freedom denoted as z):

$$v_r(z) = \begin{cases} 1 & \text{for } r = 0, \\ \cos((r+1)/2 \cdot z) & \text{for } r = 2i - 1, \\ \sin(r/2 \cdot z) & \text{for } r = 2i, \end{cases} \tag{6}$$

with $i \in \mathbb{N}$ and $0 \leqslant r \leqslant N_\xi - 1$ (Eq. (3)), respectively, $0 \leqslant r \leqslant N_\mu - 1$ (Eq. (5)). For a non-periodic transformation like a scaling t_{ext} or a rotation $\phi < 180°$, we employ only the cosine-decomposition (also in this example for one external degree of freedom denoted as z):

$$v_r(z) = \cos(r\pi z / Z_T), \tag{7}$$

whereby Z_T is the maximal range of the transformation. This implies that the function is reflected on $z = 0$ and so gives an even function with a period of $2Z_T$.

Since these decompositions are separable for several dimensions, it is easy to extend them to two or three dimensions. For example, if there is a full, i.e. 360°, turntable rotation ϕ_{table}, modelled by 5 basis functions, and a scaling t_{ext} with the maximal transformation range Z_T, modelled by 3 basis functions, we get the following $5 \times 3 = 15$ basis functions, concatenated as vector $\boldsymbol{v}$:

$$\begin{aligned} \boldsymbol{v} = (v_0 \quad v_1 \quad \cdots \quad v_{14})^{\mathrm{T}} \\ = \begin{pmatrix} 1 & \cos(\pi t_{\text{ext}}/Z_T) & \cos(2\pi t_{\text{ext}}/Z_T) \\ \cos(\phi_{\text{table}}) & \cos(\phi_{\text{table}}) \cdot \cos(\pi t_{\text{ext}}/Z_T) & \cos(\phi_{\text{table}}) \cdot \cos(2\pi t_{\text{ext}}/Z_T) \\ \sin(\phi_{\text{table}}) & \sin(\phi_{\text{table}}) \cdot \cos(\pi t_{\text{ext}}/Z_T) & \sin(\phi_{\text{table}}) \cdot \cos(2\pi t_{\text{ext}}/Z_T) \\ \cos(2\phi_{\text{table}}) & \cos(2\phi_{\text{table}}) \cdot \cos(\pi t_{\text{ext}}/Z_T) & \cos(2\phi_{\text{table}}) \cdot \cos(2\pi t_{\text{ext}}/Z_T) \\ \sin(2\phi_{\text{table}}) & \sin(2\phi_{\text{table}}) \cdot \cos(\pi t_{\text{ext}}/Z_T) & \sin(2\phi_{\text{table}}) \cdot \cos(2\pi t_{\text{ext}}/Z_T) \end{pmatrix}^{\mathrm{T}}. \end{aligned} \tag{8}$$

For three external degrees of freedom the extension is analogous.

To calculate the values of $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ and $\mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$, the basis functions v_r are employed in Eq. (3), respectively, Eq. (5). We apply the same basis functions v_r for all the functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ and $\mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$, only the coefficients $a_{\xi,m,r}$, respectively, $a_{\mu,m,q,r}$ vary. So, the function values $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ and $\mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ can be calculated fast: for a given pose $(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ of an object, the basis functions $v_r(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ have to be evaluated only once in advance, and for the single functions ξ_m and $\mu_{m,q}$ the respective coefficients $a_{\xi,m,r}$ and $a_{\mu,m,q,r}$ are multiplied by the already calculated values of the basis functions.

2.5. Training of the parameters

In the last subsections we explained the object model, now we will describe the estimation of the model parameters, especially the coefficients $a_{\xi,m,r}$ and $a_{\mu,m,q,r}$, by training images of the object. Firstly, the bounding region O will be trained, and subsequently, based on it, the statistical parameters of the object model are estimated. We will explain the training for the general case of arbitrary transformations. If there are only internal transformations ϕ_{int} and $\boldsymbol{t}_{\text{int}}$ and no external transformations t_{ext} and $\boldsymbol{\phi}_{\text{ext}}$, the method will simplify, because then ξ_m and $\mu_{m,q}$ are constants.

2.5.1. Training of the bounding region O

The coefficients $a_{\xi,m,r}$ (Eq. (3)) of the bounding region functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ are trained by $N_{t,\xi}$ images of the

object taken from different viewpoints with the respective transformation parameters $(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$. The viewpoints should be uniformly distributed over the whole transformation space, and the distance between two adjacent viewpoints should be small. In a first step, for each single viewpoint the decision is taken, which local feature vectors $\boldsymbol{c}_m$ belong to the object and which to the background. If, for example, the object is located in front of a darker background, this assignment can be performed by the following simple threshold operation:

$$\breve{\xi}_m(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)}) = \begin{cases} 0 \text{ (background)} & \text{for } c_{m,1}(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)}) < S_c, \\ 1 \text{ (object)} & \text{for } c_{m,1}(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)}) \geqslant S_c. \end{cases} \tag{9}$$

In Eq. (9), the surrogate function $\breve{\xi}_m(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ is only defined on the discrete training viewpoints $(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$. The threshold S_c is chosen manually, and it depends on the brightness of the background and the object.

Now, these discrete functions $\breve{\xi}_m(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ are approximated by the continuous functions $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}}) = \sum_{r=0}^{N_\xi - 1} a_{\xi,m,q} v_r$. The coefficients $a_{\xi,m,q}$ are computed by minimizing the squared approximation error for the training samples

$$\hat{\boldsymbol{a}}_{\xi,m} = \underset{\boldsymbol{a}_{\xi,m}}{\text{argmin}} \sum_{\rho=0}^{N_{t,\xi}-1} \left(\breve{\xi}_m(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)}) - \sum_{r=0}^{N_\xi - 1} a_{\xi,m,r} v_r^{(\rho)} \right)^2. \tag{10}$$

Note, the number N_ξ of basis functions is much smaller than the number $N_{t,\xi}$ of training samples.

By the function approximation the values of $\xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ are no longer restricted to the discrete values 0 and 1 of Eq. (9), but each can take a value between 0 and 1. Therefore we define a threshold S_ξ and use the following assignment for calculating the bounding region O for a given pose $(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$:

$$\boldsymbol{x}_m \in \begin{cases} X \backslash O \text{ (background)} & \text{for } \xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}}) < S_\xi, \\ O \text{ (object)} & \text{for } \xi_m(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}}) \geqslant S_\xi. \end{cases} \tag{11}$$

A possible choice for the threshold S_ξ could be $S_\xi = 0.5$, the mean of the original values 0 and 1 in Eq. (9). In the experiments in Section 4 we choose the lower value 0.35 for S_ξ, because so even for objects whose bounding region is "difficult" to approximate the complete object reside inside the bounding region.

This learned bounding region O (for the chosen S_ξ) is used for the training of the means $\boldsymbol{\mu}_m$ and standard deviations $\boldsymbol{\sigma}_m$ as well as during the recognition process.

2.5.2. Training of the statistical parameters

After the training of the bounding region O the statistical parameters, i.e. the means $\boldsymbol{\mu}_m$, concatenated written as $\boldsymbol{\mu}$, and standard deviations $\boldsymbol{\sigma}_m$, concatenated written as $\boldsymbol{\sigma}$, can be estimated. For that purpose $N_{t,\mu}$ images from different viewpoints are taken. As before, the viewpoints should be uniformly distributed, and the distance between neighboring viewpoints should be small. For each viewpoint two or more images with different illuminations should be used to estimate the statistical parameters, especially the standard deviation.

For each viewpoint $(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ the respective trained bounding region $O(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ is calculated, and the density about all observations is maximized:

$$(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\sigma}}) = \underset{(\boldsymbol{\mu},\boldsymbol{\sigma})}{\text{argmax}} \prod_{\rho=0}^{N_{t,\mu}-1} p(C_O^{(\rho)} | \boldsymbol{B}, \boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)}). \tag{12}$$

Since the single feature vectors $\boldsymbol{c}_m$ as well as their components are assumed to be statistically independent, each mean $\mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}})$ and standard deviation $\sigma_{m,q}$ can be calculated independently and Eq. (12) can be transformed to

$$\begin{aligned} &(\hat{\mu}_{m,q}, \hat{\sigma}_{m,q}) \\ &\quad = \underset{(\mu_{m,q},\sigma_{m,q})}{\text{argmax}} \prod_{\rho} p(c_{m,q}^{(\rho)} | \mu_{m,q}, \sigma_{m,q}, \boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)}) \\ &\quad \forall \rho : \boldsymbol{x}_m^{(\rho)} \in O(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)}). \end{aligned} \tag{13}$$

Note: Mostly, a feature vector $\boldsymbol{c}_m$ does not belong to the object for all external transformations, so normally, the number $N_{t,\mu,m}$ of training samples of this feature vector $\boldsymbol{c}_m$ is smaller than the total number $N_{t,\mu}$ of training images.

With $\mu_{m,q}(\boldsymbol{\phi}_{\text{ext}}, t_{\text{ext}}) = \sum_{r=0}^{N_\mu - 1} a_{\mu,m,q,r} v_r$ we can transform Eq. (13) and get the following term for estimating the coefficients $a_{\mu,m,q,r}$, concatenated written as vector $\boldsymbol{a}_{\mu,m,q}$:

$$\begin{aligned} &\hat{\boldsymbol{a}}_{\mu,m,q} = \underset{\boldsymbol{a}_{\mu,m,q}}{\text{argmin}} \sum_{\rho} \left\{ c_{m,q}^{(\rho)} - \sum_{r=0}^{N_{\mu,m}-1} a_{\mu,m,q,r} v_r^{(\rho)} \right\}^2 \\ &\quad \forall \rho : \boldsymbol{x}_m^{(\rho)} \in O(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)}). \end{aligned} \tag{14}$$

The number $N_{\mu,m}$ of basis functions for $\boldsymbol{\mu}_m$ has to be reduced, if the number $N_{t,\mu,m}$ of training samples for a feature vector $\boldsymbol{c}_m$ is very small.

The standard deviation $\sigma_{m,q}$ can be estimated by the following maximum likelihood estimation:

$$\begin{aligned} &\sigma_{m,q}^2 = \frac{1}{N_{t,\mu,m}} \sum_i \left\{ c_{m,q}^{(\rho)} - \mu_{m,q} \left(\boldsymbol{\phi}_{\text{ext}}^i, t_{\text{ext}}^i \right) \right\}^2 \\ &\quad \forall \rho : \boldsymbol{x}_m^{(\rho)} \in A \left(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)} \right). \end{aligned} \tag{15}$$

2.6. Localization and classification

By the described framework objects can be localized and classified in images. For the classification, for each object

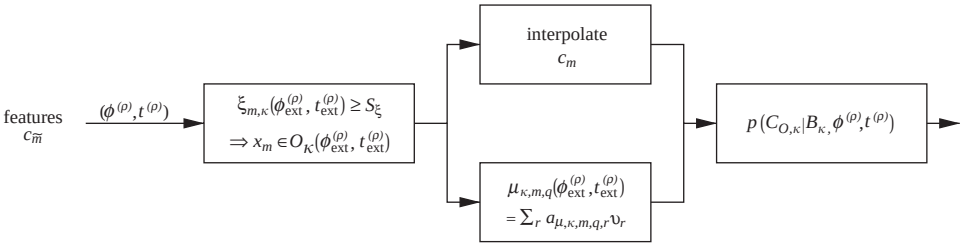

Fig. 4. Evaluation of the density function for one pose hypotheses $(\boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)})$.

class κ, with $\kappa = 1, \ldots, K$, an own object model is learned as depicted in the last subsection. It comprises the bounding region O_κ (i.e. the functions $\xi_{\kappa,m}$) and the statistical parameter $\boldsymbol{B}_\kappa$ (i.e. $\boldsymbol{\mu}_{\kappa,m}$ and $\boldsymbol{\sigma}_{\kappa,m}$). Consequently, each object is represented by its density function $p(C_{O,\kappa}|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})$.

The density $p(C_{O,\kappa}|\boldsymbol{B}_\kappa, \boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)})$ for a certain class and a pose hypotheses $(\boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)})$ is computed as shown in Fig. 4: first, the feature vectors $\boldsymbol{c}_{\tilde{m}}$ are calculated as described in Section 2.1. Subsequently, the respective bounding region $O_\kappa(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ is computed by the functions $\xi_{m,\kappa}(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$. Afterwards, the local feature vectors $\boldsymbol{c}_m$ are interpolated by the feature vectors $\boldsymbol{c}_{\tilde{m}}$ according to the internal transformations $(\phi_{\text{int}}^{(\rho)}, \boldsymbol{t}_{\text{int}}^{(\rho)})$, and the means $\mu_{\kappa,m,q}(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ are calculated by the basis functions v_r according to the external transformations $(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$. Finally, the density $p(C_{O,\kappa}|\boldsymbol{B}_\kappa, \boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)})$ can be evaluated.

For a reliable localization and classification, it has to be considered that the number N_{O_κ} of object vectors $\boldsymbol{c}_{O_\kappa,\overline{m}}$ depends on the object and the viewpoint, i.e. it can vary much. For the example in Fig. 1 in the Introduction, the stapler seen from the side takes about 8400 pixels in the image, whereas seen from the front it only takes 4000 pixels. So a simple maximum likelihood estimation on the density function $p(C_{O,\kappa}|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})$ does not work: the density $p(\boldsymbol{c}_m|\boldsymbol{\mu}_{\kappa,m}, \boldsymbol{\sigma}_{\kappa,m}, \boldsymbol{\phi}, \boldsymbol{t})$ of a single object feature vector $\boldsymbol{c}_{O_\kappa,\overline{m}}$ is normally smaller than 1. For this reason objects and viewpoints with a small number N_{O_κ} of object feature vectors are wrongly preferred, as we showed in Ref. [18]. Therefore, we normalize the density function by the N_{O_κ}th root, i.e. the geometric mean of the densities of the single object feature vectors $\boldsymbol{c}_{O_\kappa,\overline{m}}$. For the localization, when the class of the object is known, we perform a maximum likelihood estimation over all possible transformations on the normalized density function:

$$
\begin{aligned}
&(\hat{\boldsymbol{\phi}}_\kappa, \hat{\boldsymbol{t}}_\kappa) \\
&\quad = \underset{(\boldsymbol{\phi},\boldsymbol{t})}{\text{argmax}} \ \sqrt[N_{O,\kappa}]{p(C_{O,\kappa}|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})} \\
&\quad = \underset{(\boldsymbol{\phi},\boldsymbol{t})}{\text{argmax}} \ \sqrt[N_{O,\kappa}]{\prod_{\boldsymbol{x}_m \in O,\kappa} p(\boldsymbol{c}_m|\boldsymbol{\mu}_{\kappa,m}, \boldsymbol{\sigma}_{\kappa,m}, \boldsymbol{\phi}, \boldsymbol{t})}. \qquad (16)
\end{aligned}
$$

For the classification, for each class κ the potential pose $(\hat{\boldsymbol{\phi}}_\kappa, \hat{\boldsymbol{t}}_\kappa)$ is estimated analogous to Eq. (16), and the decision is taken for the class κ with highest density value:

$$
\begin{aligned}
(\kappa, \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{t}}) &= \underset{\kappa}{\text{argmax}} \ \sqrt[N_{O,\kappa}]{p(C_{O,\kappa}|\boldsymbol{B}_\kappa, \hat{\boldsymbol{\phi}}_\kappa, \hat{\boldsymbol{t}}_\kappa)} \\
&= \underset{\kappa}{\text{argmax}} \left\{ \underset{(\boldsymbol{\phi},\boldsymbol{t})}{\text{argmax}} \ \sqrt[N_{O,\kappa}]{p(C_{O,\kappa}|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})} \right\}. \qquad (17)
\end{aligned}
$$

2.6.1. Search algorithm

Normally, Eqs. (16) and (17) cannot be solved analytically. Therefore we apply a search algorithm. To speed it up, the estimation of the potential pose $(\hat{\boldsymbol{\phi}}, \hat{\boldsymbol{t}})$ of each object class is performed hierarchically. The algorithm starts with a *global search* on a coarse resolution r_{s_c}, continued by a *local search*. The result is refined on a finer resolution r_{s_f}.

For the global search the expressions in Eqs. (16) and (17) are evaluated on discrete points of the n-dimensional transformation space ($n \leqslant 6$) spanned by the possible rotations $\boldsymbol{\phi} = (\phi_{\text{int}}, \boldsymbol{\phi}_{\text{ext}})^{\text{T}}$ and translations $\boldsymbol{t} = (\boldsymbol{t}_{\text{int}}, t_{\text{ext}})^{\text{T}}$. The computationally expensive global search can be accelerated. On the one hand, the search algorithms is very robust; so the *search grid* can be chosen very coarsely, for example for a 360° turntable rotation ϕ_{table} a distance between the discrete points $\Delta\phi_{\text{table}} = 10°$ is sufficient. On the other hand, the algorithm can be strongly sped up by reusing already calculated values [18]: the size of the bounded region O_κ, i.e. the values $\xi_{\kappa,m}$, and the values of the means $\mu_{m,q}$ of the local feature vectors $\boldsymbol{c}_m$ depends only on the external transformations $\boldsymbol{\phi}_{\text{ext}}$ and t_{ext} and are independent of the internal transformations ϕ_{int} and $\boldsymbol{t}_{\text{int}}$. Whereas the interpolation of the feature vectors $\boldsymbol{c}_m$ depends only on the internal transformations ϕ_{int} and $\boldsymbol{t}_{\text{int}}$. Further, for the internal translations $\boldsymbol{t}_{\text{int}}$ we translate the object grid according to the rotated coordinates axes in steps respective to the resolution r_s. So, each interpolated feature vector can be used for many internal translations and all external transformations, as visible in the right image of Fig. 5. Consequently, we interpolate the required area of the grid for each internal rotation ϕ_{int} only once and store it. Then, we calculate the size of the bounded region O and the means $\boldsymbol{\mu}_m$ of the local feature vectors $\boldsymbol{c}_m$ for each external transformation once and combine it with the stored values of the interpolated grid. In

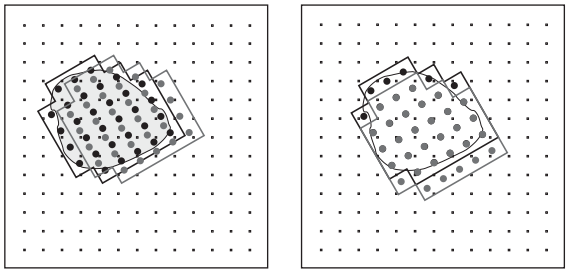

Fig. 5. Left: "naive" algorithm: for each possible internal transformation all the feature vectors have to been interpolated; right: "improved" algorithm: translating the object grid according the rotated coordinates axes in steps respective to the resolution r_s, the most feature vectors can be reused.

doing so, the global search can be accelerated, for example by the factor 50–100 for the experiments in Section 4.

The result of this global search (on the discrete points of the transformation space) is refined by a local search (Downhill–Simplex algorithm [22]), first on the coarse resolution r_{s_c} and then on the finer resolution r_{s_f}. Because of the continuous basis function v_r, even every viewpoint between the trained viewpoints can be estimated.

3. Background model

The simple object model of the last section works well as long as the objects are located in homogeneous background and are not occluded. But for real recognition tasks, these conditions are rarely fulfilled: the objects mostly reside in cluttered background, and very often they are partially occluded, as one can see in Fig. 6. Because of these reasons, the object feature vectors $\boldsymbol{c}_{O,\overline{m}}$ at the border of the object as well as of the occluded part of the object are changed. Therefore, the object model of Section 2 does not fit for these feature vectors, and the assumption that all feature vectors $\boldsymbol{c}_m$ inside the bounding region O_κ belong to the object is violated. Because of this reason we extend the object model, to handle heterogeneous background and partial occlusion.

3.1. Background model and assignment function

The main points of this extension are the explicit *background model* $\boldsymbol{B}_0$ and the *assignment function* $\boldsymbol{\zeta}_\kappa \in \{0,1\}^{N_{O_\kappa}}$ that assigns each feature vector $\boldsymbol{c}_m$ inside the bounding region O either to the background ($\zeta_{\kappa,m}=0$) or to the object ($\zeta_{\kappa,m}=1$).

The background is modelled as uniform distribution over all possible values of the feature vectors. The two main advantages of this model are: firstly, a priori, i.e. during the training of the objects, nothing has to be known about the background in the recognition phase. Secondly, every possible background can be handled by the same background

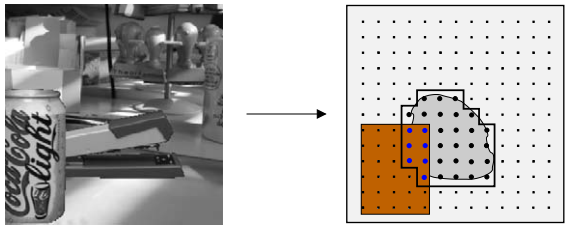

Fig. 6. Example of heterogeneous background and occlusion.

model. Besides, because of the uniform distribution, the background density $p(\boldsymbol{c}_m|\boldsymbol{B}_0)$ is identical for all positions, and thus it is independent of the transformations $\boldsymbol{\phi}$ and $\boldsymbol{t}$. The simple density function $p(C_{O,\kappa}|\boldsymbol{B}_\kappa,\boldsymbol{\phi},\boldsymbol{t})$ for an object (Eq. (4)) of the last section is extended, and now it comprises also the background model $\boldsymbol{B}_0$:

$$p(C_{O,\kappa}|\overline{\boldsymbol{B}}_\kappa,\boldsymbol{\phi},\boldsymbol{t}) = p(C_{O,\kappa}|\boldsymbol{B}_0,\boldsymbol{B}_\kappa,\boldsymbol{\phi},\boldsymbol{t}). \tag{18}$$

For the assignment function $\boldsymbol{\zeta}_\kappa$ we assume that the a priori probabilities for the assignment to the background and to the object are equal. Therefore, no expensive training of the a priori probabilities is necessary. During the recognition process the assignment $\boldsymbol{\zeta}_\kappa$ for a certain object and pose is chosen so that the density $p(C_{O,\kappa}|\overline{\boldsymbol{B}}_\kappa,\boldsymbol{\phi},\boldsymbol{t})$ is maximized:

$$p(C_{O,\kappa}|\overline{\boldsymbol{B}}_\kappa,\boldsymbol{\phi},\boldsymbol{t}) = \max_{\boldsymbol{\zeta}_\kappa} p(C_{O,\kappa}|\boldsymbol{\zeta}_\kappa,\boldsymbol{B}_0,\boldsymbol{B}_\kappa,\boldsymbol{\phi},\boldsymbol{t}), \tag{19}$$

$$p(C_{O,\kappa}|\overline{\boldsymbol{B}}_\kappa,\boldsymbol{\phi},\boldsymbol{t}) \Rightarrow \widehat{\boldsymbol{\zeta}}_\kappa = \operatorname*{argmax}_{\boldsymbol{\zeta}_\kappa} p(C_{O,\kappa}|\boldsymbol{\zeta}_\kappa,\boldsymbol{B}_0,\boldsymbol{B}_\kappa,\boldsymbol{\phi},\boldsymbol{t}) \tag{20}$$

hereby $\widehat{\boldsymbol{\zeta}}_\kappa$ is called the *optimal assignment*.

The assumption that also neighbored assignments $\zeta_{\kappa,m}$ are independent leads to

$$\begin{aligned} p(C_{O,\kappa}|\overline{\boldsymbol{B}}_\kappa,\boldsymbol{\phi},\boldsymbol{t}) &= \prod_{\boldsymbol{x}_m\in O} \max_{\zeta_{\kappa,m}} p(C_{O,\kappa}|\zeta_{\kappa,m},\boldsymbol{B}_0,\boldsymbol{B}_\kappa,\boldsymbol{\phi},\boldsymbol{t}) \\ &= \prod_{\boldsymbol{x}_m\in O} \max\{p(\boldsymbol{c}_m|\zeta_{\kappa,m}=0,\boldsymbol{B}_0), \\ &\quad p(\boldsymbol{c}_m|\zeta_{\kappa,m}=1,\boldsymbol{\mu}_m,\boldsymbol{\sigma}_m,\boldsymbol{\phi},\boldsymbol{t})\}, \end{aligned} \tag{21}$$

$$\begin{aligned} &p(C_{O,\kappa}|\overline{\boldsymbol{B}}_\kappa,\boldsymbol{\phi},\boldsymbol{t}) \\ &\Rightarrow \widehat{\zeta}_{\kappa,m} = \operatorname*{argmax}_{\zeta_{\kappa,m}}\{p(\boldsymbol{c}_m|\zeta_{\kappa,m}=0,\boldsymbol{B}_0), \\ &\quad p(\boldsymbol{c}_m|\zeta_{\kappa,m}=1,\boldsymbol{\mu}_m,\boldsymbol{\sigma}_m,\boldsymbol{\phi},\boldsymbol{t})\}. \end{aligned} \tag{22}$$

This means, the decision, whether a local feature vector belongs to the background or to the object, is taken according to the higher density value.

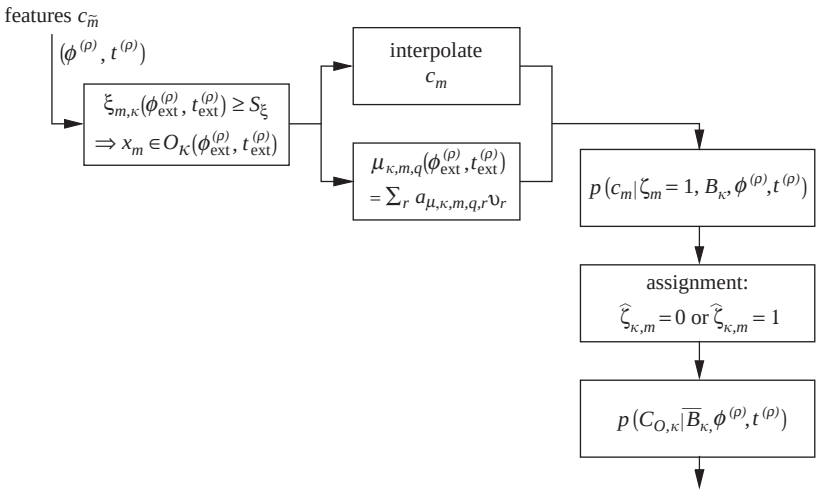

Fig. 7. Evaluation of the density function for one pose hypotheses $(\boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)})$ with background model.

3.2. Localization and classification

Using the background model, now, the pose and class of an object can be estimated in spite of heterogeneous background and occlusion. The evaluation of the density function $p(C_{O,\kappa}|\overline{\boldsymbol{B}}_\kappa, \boldsymbol{\phi}^{(\rho)}, \boldsymbol{t}^{(\rho)})$ for a pose hypotheses $(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$ is illustrated in Fig. 7. As one can see, it is similar to the simple object model in the last section Fig. 4. The only difference is the additional estimation of the assignment $\hat{\zeta}_{\kappa,m}$ for each feature vector $\boldsymbol{c}_m$ in the respective bounding region $O_\kappa(\boldsymbol{\phi}_{\text{ext}}^{(\rho)}, t_{\text{ext}}^{(\rho)})$.

Therefore for the localization and the classification nearly the same equations as in the last section can be used, only the estimation of the assignment is added. Thus, the equation for the localization is

$$(\hat{\boldsymbol{\phi}}_\kappa, \hat{\boldsymbol{t}}_\kappa, \hat{\boldsymbol{\zeta}}_\kappa) = \underset{(\boldsymbol{\phi},\boldsymbol{t},\boldsymbol{\zeta}_\kappa)}{\text{argmax}} \sqrt[N_{O,\kappa}]{p(C_{O,\kappa}|\boldsymbol{\zeta}_\kappa, \boldsymbol{B}_0, \boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})}. \tag{23}$$

The equation for the classification is

$$(\kappa, \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{t}}, \hat{\boldsymbol{\zeta}}_\kappa) = \underset{\kappa}{\text{argmax}} \left\{ \underset{(\boldsymbol{\phi},\boldsymbol{t},\boldsymbol{\zeta}_\kappa)}{\text{argmax}} \sqrt[N_{O,\kappa}]{p(C_{O,\kappa}|\boldsymbol{\zeta}_\kappa, \boldsymbol{B}_0, \boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})} \right\}. \tag{24}$$

Also, the same search algorithm as for the simple object model is applied.

4. Experiments and results

We verified our approach presented in the last two sections on three data sets: the DIROKOL database (13 objects under two external transformations), the 3D-REAL-ENV database (10 objects under two external transformations) and the 3D3 database (two objects under three external transformations). These are difficult test sets: the appearance and the size of the objects vary much, and partially the objects are very small in the image. The data sets contain images with different illuminations, heterogeneous background and partial occlusion.

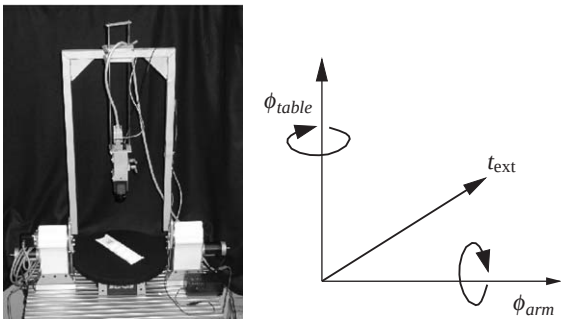

Fig. 8. Left: turntable and camera arm; right: the three external transformations: scaling t_{ext}, turntable rotation ϕ_{table} and tilt angle of the camera ϕ_{arm}.

The images of the size 256 × 256 pixels were taken with the setup illustrated in Fig. 8. The objects were put on the turntable, with $0° \leqslant \phi_{\text{table}} \leqslant 360°$, and the robot arm with the camera was moved from horizontal to vertical, i.e. $0° \leqslant \phi_{\text{arm}} \leqslant 90°$. So, we have two external rotations that form a hemisphere. Additionally for the 3D3 database, we varied the camera distance with a scale factor 1.5. Thus, we got three external transformations. The illumination changes are generated by switching lamps on and off so that the

Fig. 9. The DIROKOL database: on the one hand, office tools like staplers, hole punches, cans, and on the other hand, hospital objects like NaCl-bottle, pillbox, cups and cutlery.

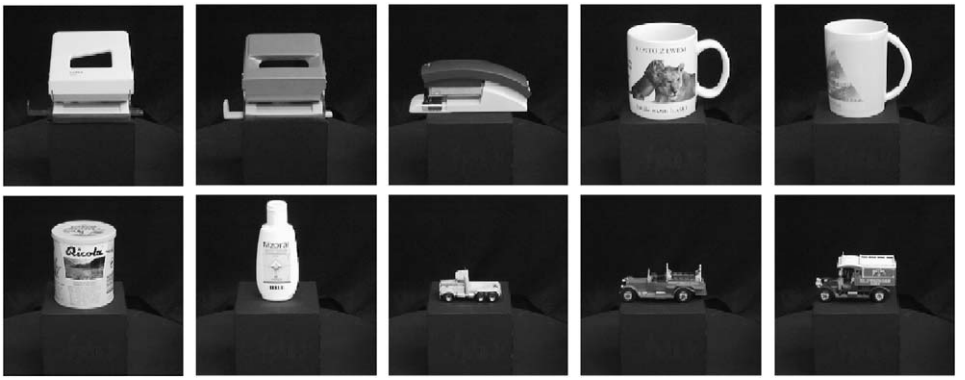

Fig. 10. The 10 objects of the 3D-REAL-ENV database.

brightness as well as the direction of lighting vary in the images.

For the DIROKOL database (see Fig. 9) from each object 3720 images were taken. Three different lighting conditions were applied so that the illumination of adjacent viewpoints is different. The training set comprises half of the data set, i.e. 1860 images for each object, so the angle between two adjacent training viewpoints is 4.2°. For the tests we took the other half of the data set, i.e. the 1860 images not used for the training.

For the training of the 3D-REAL-ENV database (see Fig. 10) we applied 1680 viewpoints, i.e. the angle between two adjacent viewpoints is 4.5°. Two different illuminations were used. So, we got 3320 training images of each object. For the tests, 3×288 additional images of each object were taken on positions and with an illumination different from the training. On each of these positions one image with homogeneous and two with real heterogeneous background are taken. Besides for each object four real scenes are arranged. That are altogether 40 scenes for this database.

For the 3D3 database (see Fig. 11) we additionally used six different camera distances t_{ext}, $20\,\text{cm} \leqslant t_{\text{ext}} \leqslant 30\,\text{cm}$ with $\Delta t_{\text{ext}} = 2\,\text{cm}$. For each camera distance t_{ext} we applied 960 viewpoints. For each viewpoint two different illuminations were utilized. These are altogether 11520

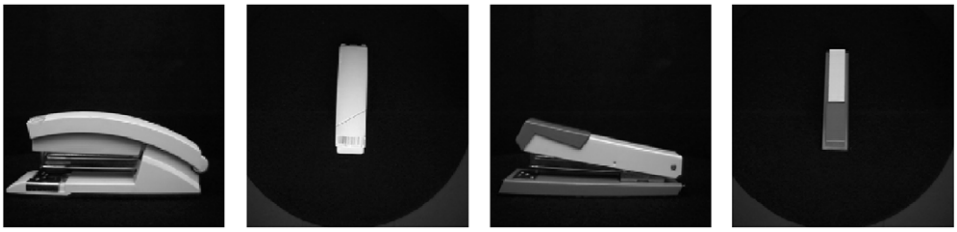

Fig. 11. 3D3 database. The white and the white–green stapler, each from the side with $t_{ext} = 20$ cm and from above with $t_{ext} = 30$ cm.

Fig. 12. Upper row left: two examples of background images. In the *two images right* an object is pasted in the background: gray can and spoon. Lower row left: two examples of objects of the 3D-REAL-ENV database in "real" heterogeneous background, right: two examples of "real scenes".

images for each object. As for the DIROKOL database the training set comprises half of the data set, i.e. the angle between two adjacent training viewpoints is 8.5°. The tests were performed on the 5760 images not used for the training.

For the experiments with heterogeneous background, we took 313 images of office scenes and pasted the objects inside these images, examples can be seen in Fig. 12. We used this method, because it is very time-consuming to produce a sufficient number of representative scenes for each object of the databases. For the same reason we generated the occlusion artificially. Exemplarily, for each object of the 3D-REAL-ENV database 576 images with real heterogenous background (not pasted) and four real scenes were taken (see Fig. 12).

We performed for the DIROKOL and the 3D3 database the following four different test scenarios: homogeneous background, heterogeneous background, homogeneous background with 20% occlusion and heterogeneous background with 20% occlusion. For the homogeneous background we tested with and without background modelling, whereas for all the other experiments we tested the object recognition system only with background modelling. For the 3D-REAL-ENV database the following four different test scenarios are used: homogeneous background ("artificial"), heterogeneous background ("real"), heterogeneous background and real scenes. Here, we always applied the background model.

To model the external rotations, we employed a sine–cosine-decomposition with 13 basis functions for the turntable rotation ϕ_{table} and a cosine-decomposition with 4 basis functions for the camera arm rotation ϕ_{arm}; that are altogether $13 \times 4 = 52$ basis functions for the DIROKOL and the 3D-REAL-ENV database. For the 3D3 database, we additionally employed a cosine-decomposition with 3 basis functions for the scaling. So, totally we get $13 \times 4 \times 3 = 156$ basis functions for this database.

Table 1
Recognition rates DIROKOL and 3D3 database

	DIROKOL				3D3	
	Localization		Classification		Local.	Classif.
	1–10	1–13	1–10	1–13		
Homog. without backm.	98.4%	95.7%	99.9%	99.7%	98.8%	100%
Homog. with backm.	97.4%	94.7%	99.8%	99.3%	99.0%	100%
Heterog.	82.3%	64.9%	88.5%	69.1%	76.9%	95.4%
Homog. +20% occl.	94.1%	88.4%	93.2%	91.5%	64.6%	99.3%
Heterog. +20% occl.	69.6%	54.7%	67.2%	54.2%	50.9%	87.4%
Time	1.7 s	1.7 s	16.9 s	22.0 s	8.0 s	16.1 s

DIROKOL: 1–10 means without cutlery, 1–13 means all 13 objects. For the experiments with occlusion only 120 test images (DIROKOL) and 720 test images (3D3) of each object were used.

Table 2
Recognition rates 3D-REAL-ENV database

3D-REAL-ENV	Localization		Classification
	All	Only right classf.	
Homog.	99.1%	99.1%	100%
Artificial heterog. background	79.7%	87.8%	82.2%
Real heterog. background	79.7%	84.9%	86.1%
Real scenes	77.5%	84.4%	80.0%
Time	1.7 s	1.7 s	17.0 s

All means localization evaluated independently of the classification result, *only right classf.* means localization only evaluated for the right classified objects. All experiments are performed with background modelling.

In addition to the external transformations, we considered the internal translations t_x and t_y, i.e. we searched the whole image for the object. So, the transformation space had four dimensions for the DIROKOL and the 3D-REAL-ENV database ($t_x, t_y, \phi_{\text{arm}}, \phi_{\text{table}}$) and five for the 3D3 database ($t_x, t_y, \phi_{\text{arm}}, \phi_{\text{table}}, t_{\text{ext}}$). The coarse resolution (see Section 2.6) was $r_{s_c} = 2^3 = 8$ pixels, the finer resolution was $r_{s_f} = 2^2 = 4$ pixels.

The results of the experiments for the DIROKOL database and the 3D3 database are presented in Table 1, the results for the 3D-REAL-ENV database in Table 2. A localization is counted as wrong, if the error for the internal translations t_x or t_y is bigger than 10 pixels or the error for the external rotations ϕ_{table} or ϕ_{arm} is bigger than 15° or the error for the scaling t_{ext} is bigger than 2 cm. That corresponds to the accuracy of a human observer and it is also sufficient for many technical applications. In Table 1 the localization results are evaluated independently of the classification results. In Table 2 the localization results are evaluated independently as well as dependently on the classification results.

The recognition rates for the objects in front of a homogeneous background are very high, mostly 96–100%, with and without background modelling. As one can see in Fig. 13, the trained bounding region O encloses the object very tightly. In contrast to this, the fixed region of Fig. 1, here plotted dashedly, is too big. By the use of the variable bounding region, for heterogeneous background recognition rates around 80% could be reached. The results for the "artificial" heterogeneous background, the "real" heterogeneous background as well as for the scenes are comparable. Only the cutlery in the DIROKOL database was often not found in the heterogeneous background. But also for a human observer, it is difficult to detect the cutlery in the heterogeneous background as one can see in the right image in Fig. 12.

For homogeneous background and 20% occlusion the recognition rates of the DIROKOL database are mostly greater than 90%. Even for the difficult task that the objects are located in heterogeneous background and 20% of them are occluded, the recognition rates for the DIROKOL database nearly reach considerable 70%. Also for the 3D3 database, the localization rate amount 50%, although there are two internal and three external transformations. Fig. 14 illustrates that the background model and the assignment function ζ works reliably. In spite of heterogeneous background and occlusion, the hole punch is localized well. Most of the feature vectors at the border and of the occluded part of the object are assigned to the background, whereas the others are principally assigned to the object. The average computation time for one localization with known object class is 1.7 s on a Pentium IV with 2.4 GHz for the DIROKOL and for the 3D-REAL-ENV database and 8.0 s for the 3D3 database, because here the transformation space comprises one dimension more.

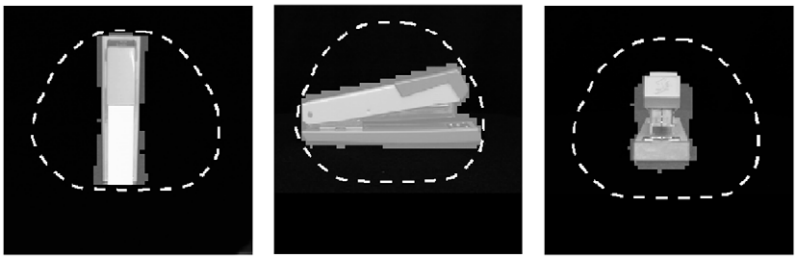

Fig. 13. The same viewpoints for the stapler as in Fig. 1. The trained bounding region O is plotted in gray. For comparison the fixed region of Fig. 1 is plotted dashedly.

Fig. 14. From left: hole punch partially occluded in heterogeneous background; pose estimated by the object recognition system; respective trained bounding region O; these feature vectors marked gray (inside the bounding region) are assigned to the object; the others are assigned to the background.

5. Conclusions and outlook

5.1. Conclusions

In this article we presented a powerful statistical, appearance-based approach for classification and localization of 3-D objects in complex scenes. We modelled the region of the object in the image, i.e. the bounding region O, as a function of the external transformations. Also, the local object features were modelled as functions of the external transformations. We formulated the dependency on the external transformations by sums of continuous basis functions, i.e. sine–cosine- and the cosine-decomposition. For robustness, we applied a statistical framework that also includes a background model and an assignment function.

In the experiments, we showed that the trained, variable bounding region O encloses the object for the external transformations very tightly. This is a great advantage over other approaches, e.g. Refs. [7,9,10,14,15,17], which use a fixed bounding region and so have problems to handle the varying size of the objects. By the normalization of the density function by the N_{O_κ}th root, i.e. the geometric mean of the densities of the single object feature vectors $\boldsymbol{c}_{O_\kappa,\overline{m}}$, also objects whose size differ much can be recognized. Besides by the use of the trigonometric basis functions, 52 basis functions are sufficient to model all viewpoints on a hemisphere. The background model and the assignment function $\zeta_{\kappa,m}$ work well: by heterogeneous background and occlusions, the single feature vectors inside the bounding region are reliably assigned to the object or to the background. In spite of non-uniform illumination changes, heterogeneous background and occlusions, we got good recognition rates on three data sets that comprises two and three external transformations. Our approach is even suitable for real scenes.

5.2. Discussion and outlook

The initial global search seems to be expensive. But also other appearance-based approaches, even the eigenspace-approaches of Murase and Nayar [7,17], Bischof et al. [14] and Leonardis and Bischof [15], starts with an exhaustive search in the whole image. Mostly, they shift the template only one pixel each time. Additionally, for the robust eigenspace-approach [15], one has to apply for each object class an own eigenspace, and one has to evaluate several hypotheses for each possible internal transformation. Whereas for our approach the "search grid" for the global search can be coarse: the bounding region can be shifted $\Delta x = \Delta y = 8$ pixels, and for the external transformations we only need to evaluate a limited number of hypotheses, for example 36 hypotheses for a 360° turn table rotation. In the future we will develop this algorithm further.

Here we presented results on single object recognition: The object with the highest density value according to Eqs. (17) and (24) is recognized. However, by the use of the tight bounding region O and the assignment function $\boldsymbol{\zeta}$, our approach is capable to classify in multiobject scenes. The main idea is to mask out the feature vectors assigned to already

recognized objects and to perform a second recognition on the same image, until no further object is detected. Some results on this approach can be found in Ref. [23].

Acknowledgements

This research work was funded by the German Research Foundation (DFG) Graduate Research Center "3-D Image Analysis and Synthesis".

References

[1] I. Shimshoni, J. Ponce, Probabilistic 3-D object recognition, Int. J. Comput. Vision 36 (1) (2000) 51–70.
[2] A.R. Pope, D.G. Lowe, Probabilistic models of appearance for 3-D object recognition, Int. J. Comput. Vision 40 (2) (2000) 149–167.
[3] M.S. Costa, L.G. Shapiro, 3-D object recognition and pose with relational indexing, Comput. Vision Image Understand. 79 (2000) 364–407.
[4] A. Selinger, R.C. Nelson, Minimally supervised acquisition of 3D recognition models from cluttered images, in: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2001, pp. I:213–220.
[5] J. Hornegger, V. Welker, H. Niemann, Localization and classification based on projections, Pattern Recognition 35 (2002) 1225–1235.
[6] H. Chen, I. Shimshoni, P. Meer, Model based object recognition by robust information fusion, in: 17th International Conference on Pattern Recognition (ICPR), Cambridge, UK, August 2004.
[7] H. Murase, S.K. Nayar, Visual learning and recognition of 3-D objects from appearance, Int. J. Comput. Vision 14 (1995) 5–24.
[8] B. Schiele, J.L. Crowley, Recognition without correspondence using multidimensional receptive field histograms, Int. J. Comput. Vision 36 (1) (2000) 31–50.
[9] H. Borotschnig, L. Paletta, M. Prantl, A. Pinz, Appearance-based active object recognition, Image Vision Comput. 18 (9) (2000) 715–727.
[10] J. Dahmen, D. Keysers, H. Ney, M.O. Guld, Statistical image object recognition using mixture densities, J. Math. Imag. Vision 14 (3) (2001) 285–296.
[11] J. Pösl, H. Niemann, Erscheinungsbasierte statistische Objekterkennung, Inf.—Forsch. Entwicklung 17 (1) (2002) 21–40.
[12] Ch. Gräßl, F. Deinzer, H. Niemann, Continuous parametrization of normal distribution for improving the discrete statistical eigenspace approach for object recognition, in: V. Krasnoproshin, S. Ablameyko, J. Soldek (Eds.), Pattern Recognition and Information Processing 03, Minsk, Belarus, May 2003, pp. 73–77.
[13] C. Wallraven, B. Caputo, A. Graf, Recognition with local features: the kernel recipe, in: Ninth International Conference on Computer Vision (ICCV), Nice, France, October 2003, pp. 257–264.
[14] H. Bischof, H. Wildenauer, A. Leonardis, Illumination insensitive recognition using eigenspaces, Comput. Vision Image Understand. 95 (1) (2004) 86–104.
[15] A. Leonardis, H. Bischof, Robust recognition using eigenimages, Comput. Vision Image Understand. 78 (1) (2000) 99–118.
[16] J. Dahmen, D. Keysers, M. Motter, H. Ney, T. Lehmann, B. Wein, An automatic approach to invariant radiograph classification, in: H. Handels, A. Horsch, T. Lehmann, H.-P. Meinzer (Eds.), Bildverarbeitung für die Medizin 2001, Springer, Berlin, Lübeck, March 2001, pp. 337–341.
[17] H. Murase, S.K. Nayar, Detection of 3D objects in cluttered scenes using hierarchical eigenspace, Pattern Recognition Lett. 18 (5) (1997) 375–384.
[18] M. Reinhold, D. Paulus, H. Niemann, Improved appearance-based 3-D object recognition using wavelet features, in: T. Ertl, B. Girod, G. Greiner, H. Niemann, H.-P. Seidel (Eds.), Vision, Modeling, and Visualization 2001, AKA/IOS Press, Berlin, Amsterdam, Stuttgart, November 2001, pp. 473–480.
[19] S. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.
[20] M.J.D. Powell, Approximation Theory and Methods, Cambridge University Press, Cambridge, 1981.
[21] M. Ghanbari, Video Coding—An Introduction to Standard Codecs, The Institution of Electrical Engineers, London, UK, 1999.
[22] W.H. Press, B.P. Flannery, S.A. Teukolsky, W. Vetterling, Numerical Recipes in C++—The Art of Scientific Computation, Cambridge University Press, New York, 2002.
[23] M. Grzegorzek, K. Pasumarthy, M. Reinhold, H. Niemann, Statistical object recognition for multi-object scenes with heterogeneous background, in: B. Chanda, S. Chandran, L. Davis (Eds.), Fourth Indian Conference on Computer Vision, Graphics and Image Processing, Allied Publishers Private Limited, Kolkata, India, December 2004.

About the Author—MICHAEL P. REINHOLD studied Electrical Engineering and received the degree Diplom–Ingenieur at the RWTH Aachen, Germany. Afterwards he obtained the doctoral degree from the University Erlangen-Nuremberg, Germany. His research interests were statistical modelling, object recognition and computer vision. Now he is development engineer at Rohde&Schwarz in Munich, Germany. There, he works in the Center of Competence for Digital Signal Processing.

About the Author—MARCIN GRZEGORZEK studied Computer Science at the Silesian University of Technology Gliwice (Poland), and graduated with the degree "magister inzynier". His specialization was application and system programming. Since December 2002 he is Ph.D. candidate and member of the research staff of the Chair for Pattern Recognition at the University Erlangen-Nuremberg, Germany. His topics are 3-D object recognition, statistical modelling, and computer vision.

About the Author—JOACHIM DENZLER studied computer science at the University Erlangen-Nuremberg, Germany, from 1987 to 1992, and graduated with the degree 'Diplom-Informatiker'. He received his doctoral degree in computer science in 1997, and the 'Habilitation' in June 2003. Currently, he holds the position of a full professor at the department of mathematics and computer science at the University of Jena, Germany. His research activities concentrate on probabilistic methods in computer vision, object recognition and tracking as well as 3-D reconstruction. Joachim is member of IEEE, IEEE Computer Society and GI.

About the Author—HEINRICH NIEMANN has been Professor of Computer Science at the University of Erlangen-Nuremberg since 1975. 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 seven books and about 400 journal and conference contributions as well as editor or coeditor of 24 proceedings volumes and special issues.