

M. Grzegorzek, I. Scholz, M. Reinhold, and H. Niemann. Fast training for object recognition with structure-from-motion. *Pattern Recognition and Image Analysis: Nauka Interperiodica*, 17(1):87–92, January 2007.

# Fast Training for Object Recognition with Structure-from-Motion<sup>¶</sup>

M. Grzegorzek, I. Scholz, M. Reinhold, and H. Niemann

*Chair for Pattern Recognition, University of Erlangen–Nürnberg, Martensstr. 3, 91058 Erlangen, Germany*

*e-mail: {grzegorz,scholz,reinhold,niemann}@informatik.uni-erlangen.de*

**Abstract**—In this paper we present a system for statistical object classification and localization that applies a simplified image acquisition process for the learning phase. Instead of using complex setups to take training images in known poses, which is very time-consuming and not possible for some objects, we use a handheld camera. The pose parameters of objects in all training frames that are necessary for creating the object models are determined using a structure-from-motion algorithm. The local feature vectors we use are derived from wavelet multiresolution analysis. We model the object area as a function of 3D transformations and introduce a background model. Experiments made on a real data set taken with a handheld camera with more than 2500 images show that it is possible to obtain good classification and localization rates using this fast image acquisition method.

**DOI:** 10.1134/S1054661807010105

## INTRODUCTION

For many tasks the localization and classification of objects in images is very useful, sometimes even necessary. Possible applications in this area are, for example, face recognition [3], localization of obstacles on the road with a camera mounted on a driving car, service robotics [15], and so on. The learning process in most object recognition systems begins with the image acquisition of all possible object classes in many known poses. In a laboratory environment, the images can be taken with a special setup such as a turntable with a camera arm (Fig. 1, left).

In real problems of object recognition in images, it is much easier to record the objects using a handheld camera (Fig. 1, right). For this reason we propose a new approach for object recognition, where the image acquisition is done in this way. The goal of our algorithm is to optimize the training process with respect to execution time and ease of image acquisition while still getting satisfying classification and localization rates. The poses of the objects in all training frames are computed using a structure-from-motion algorithm [5]. The whole learning process is therefore independent of environment assumptions, but we have to deal with an additional training inaccuracy.

Two main approaches exist to solving the problem of object recognition in images: the model- and the appearance-based methods. The model-based systems use a segmentation step to extract features of objects [6]. The appearance-based approaches compute the fea-

ture vectors directly from pixel intensities in the images [3, 11]. There are appearance-based systems that use one global feature vector for the whole image (e.g., the eigenspace approach [2]), and those that use more local feature vectors (e.g., neural networks [9]). In the present work, local feature vectors with two components are applied, which are computed with a wavelet multiresolution analysis [7] and statistically modeled by density functions.

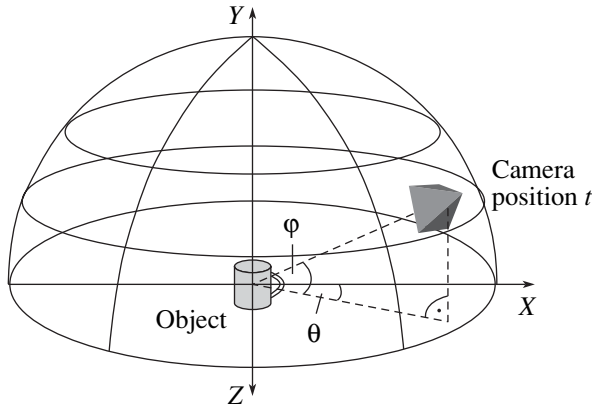
In the next section, we introduce the pose parameter reconstruction using a structure-from-motion algorithm, which yields the training pose parameters needed for object modeling. In the following two sections, the training of statistical object models and the algorithm for object localization and classification is presented. After that we describe experiments and discuss the results. We close our contribution with a conclusion.



**Fig. 1.** Left: turntable with camera arm. Right: handheld camera.

---

<sup>¶</sup> The text was submitted by the authors in English.



**Fig. 2.** Calculating  $\theta_i$  and  $\phi_i$  using the camera pose. The camera is depicted as a pyramid with its tip being the optical center and its base being the image plane.

### POSE PARAMETER RECONSTRUCTION

Suppose an image sequence is given which was taken by moving a handheld camera around an object and showing it from different directions (Fig. 1, right). In order to train the object recognition system, it is necessary to estimate the internal and external object pose parameters for all frames. The internal pose parameters denote two translations and a rotation inside the image plane. The external pose parameters are two rotations outside the image plane and a translation along the optical axis of the camera. Only four of these six pose parameters—internal translations  $\mathbf{u} = (u_1, u_2)^T$  and external rotations  $\Phi = (\theta, \phi)^T$ —are used in our experiments; therefore, only the computation of these parameters will be explained in the following.

The first step is to compute a 3D reconstruction of the camera motion and scene structure using a structure-from-motion algorithm [5]. This requires the knowledge of the point correspondences in the images, which are retrieved by a feature detection and tracking algorithm based on the gradient tracking technique introduced by Tomasi and Kanade [12]. The extensions of the original algorithm, including affine distortion handling and robustness against illumination changes, are explained in detail in [14].

By applying a factorization method, in this case the paraperspective factorization introduced by Poelman and Kanade [10], the camera motion parameters and 3D point positions corresponding to the tracked 2D features are reconstructed for a relatively short initial subsequence. The subsequence is chosen as the longest one with a certain number of features visible in all its images, since this is a prerequisite of the factorization algorithm. The results are refined by a nonlinear optimization as proposed in [4], which minimizes the back-projection error of the reconstructed 3D points. In order to obtain a Euclidean reconstruction, the intrinsic camera parameters, are set to approximations of the true values, i.e., the principal point is set to the center of the

image, the focal length is roughly estimated and the image skew is assumed to be zero. If the intrinsic parameters are not exactly the correct ones, the reconstruction will be slightly skewed projectively. However, this effect is not sizable, as examined in [8]. The remaining camera and point positions are estimated by a similar optimization image by image. For this, previously unused features' correspondences are triangulated after each estimation of a new camera pose to increase the number of available 3D points. As an initialization for each new image the projection matrix of a neighboring image in the sequence is used. The method is explained in detail in [5].

At this point the cameras parameters for each image are given as projection matrices  $\mathbf{P}_i = \mathbf{K}(\mathbf{R}_i^T | -\mathbf{R}_i^T \mathbf{t}_i)$ , where  $\mathbf{K} \in 3 \times 3$  contains the camera intrinsic parameters and  $\mathbf{R}_i \in 3 \times 3$  and  $\mathbf{t}_i \in 3 \times 1$  denote the rotation and translation of the camera. The object recognition system, on the other hand, requires an entirely different parameter representation. Therefore, the parameters are transformed as follows. First, the origin of the coordinate system is translated into the center of mass of the object  $\bar{\mathbf{p}}$ . Since the object was placed on a black background, the feature-tracking algorithm is only able to track features on the object itself, and all 3D points are assumed to be on the surface of the object. Thus, the centroid of the reconstructed 3D points is used as an approximation to the center of mass of the object. The calculated translation is applied to all camera and 3D point positions.

The external rotations in polar coordinates for the training image  $\mathbf{f}_i$  can now be calculated easily, as depicted in Fig. 2. For a given translation  $\mathbf{t}_i = (t_{i,x}, t_{i,y}, t_{i,z})^T$  of the camera in world coordinates, the angle  $\theta_i$  computes as

$$\theta_i = \arcsin(t_{i,x} / \sqrt{t_{i,x}^2 + t_{i,z}^2}) \quad (1)$$

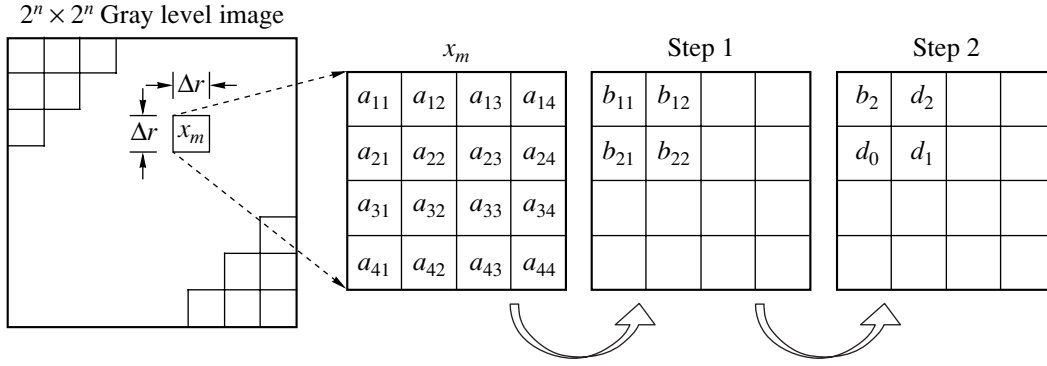
and the angle  $\phi_i$  as

$$\phi_i = -\arcsin(t_{i,y} / \sqrt{t_{i,x}^2 + t_{i,y}^2 + t_{i,z}^2}). \quad (2)$$

The internal translation is estimated by back-projecting the center of mass of the object into image coordinates, i.e.,  $\mathbf{u}_i' = \mathbf{P}_i \bar{\mathbf{p}}'$ , where  $\mathbf{u}_i'$  and  $\bar{\mathbf{p}}'$  denote  $\mathbf{u}_i$  and  $\bar{\mathbf{p}}$  in homogeneous coordinates.

### TRAINING OF STATISTICAL OBJECT MODELS

In order to learn a statistical object model  $M_K$  for an object class  $\Omega_K$ , we take an image sequence of the object, determine the pose parameters in each frame using the structure-from-motion algorithm, and preprocess and compute feature vectors in the training images. An object area is then defined, and the feature vectors modeled as density functions.



**Fig. 3.** Computation of a feature vector at a grid point  $\mathbf{x}_m$  for the scale  $s = 2$ .  $b_{ij}$  are calculated by horizontal and vertical low-pass filtering of  $a_{ij}$  and resolution reduction by factor 0.5. The final coefficients result from  $b_{ij}$  as follow:  $b_2$ —low-pass horizontal and low-pass vertical,  $d_0$ —low-pass horizontal and high-pass vertical,  $d_1$ —high-pass horizontal and high-pass vertical, and  $d_2$ —high-pass horizontal and low-pass vertical.  $\Delta r = 2^s = 4$  in this case.

The original training images are preprocessed by resizing them to square gray level images with a size of  $2^n \times 2^n$  pixels, where  $n \in \{7, 8, 9\}$ . One image of each object class is used as a reference image. With a pose of an object in the image  $\mathbf{f}_i$ , we denote the 3D transformation (translation and rotation) that maps the object in the reference image to the object in  $\mathbf{f}_i$ . Up to the end of the current section, the number of object class  $\kappa$  is omitted, because the training of the statistical object model is identical for all object classes.

For the feature extraction we divide each preprocessed training image into squares of size  $2^s \times 2^s$  ( $s \leq n$ ) pixels, and set in their centers grid points  $\mathbf{x}_m$ . 2D feature vectors  $\mathbf{c}_m = \mathbf{c}(\mathbf{x}_m)$  are computed on all of these  $2^{n-s} \times 2^{n-s}$  grid points. For this purpose we perform  $s$  times the wavelet multiresolution analysis [7] using Haar Wavelet. The components of the feature vectors are given by

$$\mathbf{c}_m = \begin{pmatrix} \ln(2^s |b_{s,m}|) \\ \ln(2^s (|d_{0,s,m}| + |d_{1,s,m}| + |d_{2,s,m}|)) \end{pmatrix}, \quad (3)$$

where  $b_{s,m}$  is a low-pass coefficient and  $d_{0, \dots, 2, s, m}$  result from combination of high-pass and low-pass filtering. An illustration for the feature vector computation for  $s = 2$  can be seen in Fig. 3. The indices  $s$  and  $m$  are omitted.

For the object model, we consider only those feature vectors that belong to the object and not to the background. For each feature vector  $\mathbf{c}_m$  in each external training pose  $(\Phi_{\text{ext}, t}, u_{\text{ext}, t})$  (for each training image), a discrete assignment function is defined:

$$\hat{\xi}_m(\Phi_{\text{ext}, t}, u_{\text{ext}, t}) = \begin{cases} 1, & \text{if } c_{m,1}(\Phi_{\text{ext}, t}, u_{\text{ext}, t}) \geq S_t \\ 0, & \text{if } c_{m,1}(\Phi_{\text{ext}, t}, u_{\text{ext}, t}) < S_t \end{cases}. \quad (4)$$

The threshold value  $S_t$  is chosen manually. In the test images, objects appear not only in the training poses, but also between them. In order to localize such objects, we construct a continuous assignment function  $\xi_m(\Phi_{\text{ext}}, u_{\text{ext}})$  using values of  $\hat{\xi}_m(\Phi_{\text{ext}, t}, u_{\text{ext}, t})$  by interpolation with trigonometric functions. The set of feature vectors belonging to the object for the given external pose  $(\Phi_{\text{ext}}, u_{\text{ext}})$  can be now determined with the following rule:

$$\begin{aligned} \xi_m(\Phi_{\text{ext}}, u_{\text{ext}}) &\geq S_0 \\ \Rightarrow \mathbf{c}_m(\Phi_{\text{ext}}, u_{\text{ext}}) &\in O(\Phi_{\text{ext}}, u_{\text{ext}}). \end{aligned} \quad (5)$$

The threshold  $S_0$  is also chosen manually. In the case of internal transformations, the object area does not change size and can be translated and rotated with these transformations. Thus, we can write the object area as a function of all transformation parameters:  $O(\Phi, \mathbf{u})$ .

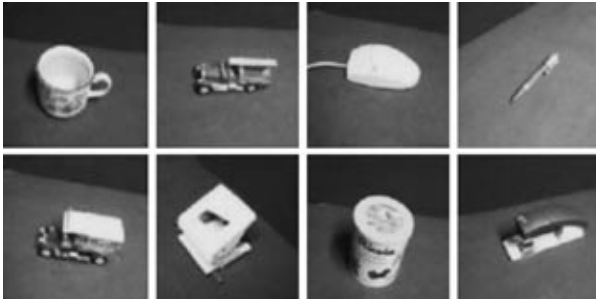
All feature vectors computed in the training phase (3) are interpreted as random variables. The object feature vectors are modeled with the normal distribution [11]. For each object feature vector  $\mathbf{c}_m \in O$ , we compute a mean value vector  $\boldsymbol{\mu}_m$  and a standard deviation vector  $\boldsymbol{\sigma}_m$ . The density of the object feature vector can be written as

$$p(\mathbf{c}_m) = p(\mathbf{c}_m | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m, \Phi, \mathbf{u}). \quad (6)$$

The feature vectors, which belong to the background are modeled with the uniform distribution, and their density functions are constant:  $p(\mathbf{c}_m) = p_b$ .

## LOCALIZATION AND CLASSIFICATION

After a corresponding object model  $M_\kappa$  is created for each object class  $\Omega_\kappa$ , we can localize and classify objects in test images. At the beginning, each test image is preprocessed and feature vectors are computed with the same method as in the training phase. Then, we start



**Fig. 4.** Used object classes. In the first row from left: cup, toy fire engine, mouse, and pen. In the second row from left: toy passenger car, hole puncher, candy box, and stapler.

our localization and classification algorithm based on the maximum likelihood estimation [13], which maximizes the object density value.

In order to compute the object density value for the class  $\Omega_K$  in the pose  $(\Phi, \mathbf{u})$  for the given test image  $\mathbf{f}$  we determine the set of feature vectors that belong to the object  $C = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_M\}$  according to (5) and compute their values with Eq. (3). Then, we compare the calculated feature vectors with the object densities stored in the object model  $M_K$  and determine the values of these vectors  $(p(\mathbf{c}_1), p(\mathbf{c}_2), \dots, p(\mathbf{c}_M))$ . The density value of the object  $\Omega_K$  in the pose  $(\Phi, \mathbf{u})$  for the given test image  $\mathbf{f}$  is given by

$$p(C|\mathbf{B}_K, \Phi, \mathbf{u}) = \prod_{i=0}^M \max\{p(\mathbf{c}_i), p_b\}, \quad (7)$$

where  $\mathbf{B}_K$  comprehends the trained mean value vectors and standard deviation vectors from the model  $M_K$  and  $p_b$  is the background density value.

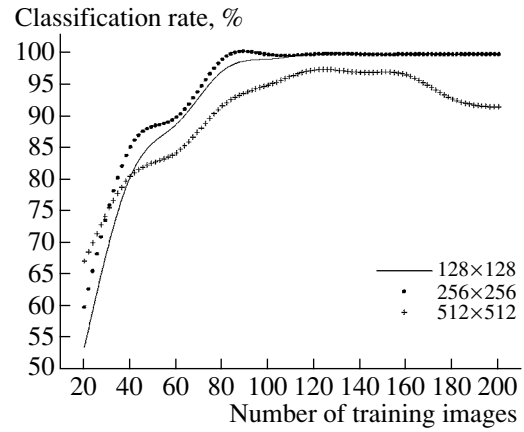
The localization and classification algorithm is realized with a maximum likelihood estimation and can be described with the following equation:

$$(\hat{\kappa}, \hat{\Phi}_\kappa, \hat{\mathbf{u}}_\kappa) = \underset{\kappa}{\operatorname{argmax}} \left\{ \underset{(\Phi, \mathbf{u})}{\operatorname{argmax}} G(p(C|\mathbf{B}_\kappa, \Phi, \mathbf{u})) \right\}, \quad (8)$$

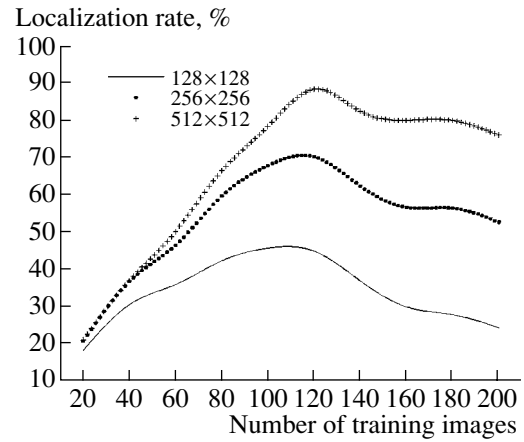
where  $\hat{\kappa}$  is the classification result and  $(\hat{\Phi}, \hat{\mathbf{u}})$  is the localization result. First, the object density (normalized by  $G$ ) is maximized according to the pose parameters  $(\Phi, \mathbf{u})$  and then to the object class  $\kappa$ . The norm function  $G$  is defined by

$$G(p(C|\mathbf{B}_\kappa, \Phi, \mathbf{u})) = \sqrt[M]{G(p(C|\mathbf{B}_\kappa, \Phi, \mathbf{u}))}, \quad (9)$$

where  $M$  is the number of feature vectors belonging to the object area  $O_\kappa(\Phi, \mathbf{u})$ . This norm function reduces the dependency between the maximization result and the object area size.



**Fig. 5.** Classification rate depending on the number of training images sized  $128 \times 128$ ,  $256 \times 256$ , and  $512 \times 512$  pixels.



**Fig. 6.** Localization rate depending on the number of training images sized  $128 \times 128$ ,  $256 \times 256$ , and  $512 \times 512$  pixels.

## EXPERIMENTS AND RESULTS

We count a localization result as correct if the error for the external rotations is not larger than  $15^\circ$  and the error for the internal translations is not larger than 10 pixels.

We tested our approach on a data set that consisted of eight objects, which are illustrated in Fig. 4.

In the training phase, sequences with more than 200 frames of each object class were taken with a handheld camera (Fig. 1, right), which accelerates the image acquisition process compared to the common methods. The recording of 200 training images of objects located on a turntable (Fig. 1, left) takes about 20 minutes. Using the handheld camera, we obtained a video with 200 frames in about 5 seconds. Next, we preprocessed the original images by converting the  $512 \times 512$ -pixel color images to gray level images with sizes of  $128 \times 128$ ,  $256 \times 256$ , and  $512 \times 512$  pixels and created the

Recognition rates for 120 training frames

| training sequences<br>120 frames | $128 \times 128$ | $256 \times 256$ | $512 \times 512$ |
|----------------------------------|------------------|------------------|------------------|
| classification                   | 100%             | 100%             | 97.5%            |
| localization                     | 45.0%            | 70.3%            | 88.5%            |

object models. The preprocessing of 100 training frames and creation of one object model takes 27 s for an image with a size of  $128 \times 128$  pixels, 36 s for  $256 \times 256$  pixels, and 44 s for  $512 \times 512$  pixels on a Pentium 4 (2.66 GHz).

For the recognition phase, we took eight image sequences with about 120 frames on a homogeneous background. The recognition time in 100 test images amounts to 72 s for the  $128 \times 128$ -pixel images, 114 s in the case of images with sizes of  $256 \times 256$  pixels, and 158 s for  $512 \times 512$  pixels.

The classification rates as a function of the number of training images are presented in Fig. 5.

A very good classification result (98.8%) with a relatively short execution time (training of one object class 38 s and recognition in 100 test images 72 s on a Pentium 4, 2.66 GHz) was obtained using 140 training images with a size of  $128 \times 128$  pixels. The results show that using larger images does not bring about an improvement in the classification rates.

In the case of localization, the results are much better for images with a size of  $512 \times 512$  pixels than for resolutions of  $256 \times 256$  and  $128 \times 128$  pixels (Fig. 6).

A comparison of the classification and localization rates for the case of 120 training images can be seen in the table.

## CONCLUSIONS

In this paper we presented an approach for statistical object classification and localization of 3D objects in which image data acquisition was performed using a handheld camera. This innovation accelerated, simplified, and universalized the learning process compared to most other object recognition systems. The pose parameters of the training frames needed for creating the object models were calculated using a structure-from-motion algorithm. To insure robustness of the system, we applied a statistical framework that included both object and background models.

In the experiments we showed that it is possible to obtain excellent recognition rates in a relatively short execution time.

In the future we will work on the algorithm for pose parameter reconstruction and the system for statistical object recognition in order to improve the localization rates.

## REFERENCES

1. C. Chui, *An Introduction to Wavelets* (Academic Press, San Diego, 1992).
2. C. Gräßl, F. Deinzer, and H. Niemann, "Continuous Parametrization of Normal Distribution for Improving the Discrete Statistical Eigenspace Approach for Object Recognition," in *Proceedings of Conference on Pattern Recognition and Information Processing 03, Minsk, May 2003*, pp. 73–77.
3. R. Gross, I. Matthews, and S. Baker, "Appearance-Based Face Recognition and Light Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26** (4), 449–465 (2004).
4. R. Hartley, "Euclidean Reconstruction from Uncalibrated Views," in *Applications of Invariance in Computer Vision*, Vol. 825 of *Lecture Notes in Computer Science* (Springer-Verlag, 1994), pp. 237–256.
5. B. Heigl, *Plenoptic Scene Modeling from Uncalibrated Image Sequences* (ibidem-Verlag, Stuttgart, 2004).
6. J. Kerr and P. Compton, "Toward Generic Model-Based Object Recognition by Knowledge Acquisition and Machine Learning," in *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, August 2003*, pp. 9–15.
7. S. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11** (7), 674–693 (1989).
8. H. Niemann and I. Scholz, "Evaluating the Quality of Light Fields Computed from Handheld Camera Images," *Pattern Recognition Letters* **26** (3), 239–249 (2005).
9. S. Park, J. Lee, and S. Kim, "Content-Based Image Classification Using a Neural Network," *Pattern Recognition Letters* **25** (3), 287–300 (2004).
10. C. Poelman and T. Kanade, "A Paraperspective Factorization Method for Shape and Motion Recovery," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19** (3), 206–218 (1997).
11. M. Reinhold, *Robuste, Probabilistische, Erscheinungs-basierte Objekterkennung* (Logos Verlag, Berlin, 2004).
12. C. Tomasi and T. Kanade, *Detection and Tracking of Point Features. Technical Report CMU-CS-91-132* (Carnegie Mellon University, 1991).
13. A. R. Webb, *Statistical Pattern Recognition* (John Wiley & Sons Ltd, Chichester, England, 2002).
14. T. Zinßer, C. Gräßl, and H. Niemann, "Efficient Feature Tracking for Long Video Sequences," in *Proceedings of 26th DAGM Symposium of Pattern Recognition, Springer-Verlag, August 2004* (to appear).
15. M. Zobel, J. Denzler, B. Heigl, E. Nöth, D. Paulus, J. Schmidt, and G. Stemmer, "MOBSY: Integration of Vision and Dialogue in Service Robots," *Machine Vision and Applications* **14** (1), 26–34 (2003).



**Marcin Grzegorzek**, born in 1977, obtained his Master's Degree in Engineering from the Silesian University of Technology Gliwice (Poland) in 2002. Since December 2002 he has been a PhD candidate and member of the research staff of the Chair for Pattern Recognition at the University of Erlangen–Nuremberg, Germany. His fields are 3D object recognition, statistical modeling, and computer vision. He is an author or coauthor of seven

publications.



**Ingo Scholz**, born in 1975, graduated in computer science at the University of Erlangen–Nuremberg, Germany, in 2000 with a degree in Engineering. Since 2001 he has been working as a research staff member at the Institute for Pattern Recognition of the University of Erlangen–Nuremberg. His main research focuses on the reconstruction of light field models, camera calibration techniques, and structure from motion. He is an author

or coauthor of ten publications and member of the German Gesellschaft für Informatik (GI).



**Michael Reinhold**, born in 1969, obtained his degree in Electrical Engineering from RWTH Aachen University, Germany, in 1998. Later, he received a Doctor of Engineering from the University of Erlangen–Nuremberg, Germany, in 2003. His research interests are statistical modeling, object recognition, and computer vision. He is currently a development engineer at Rohde & Schwarz in Munich, Germany, where he works in

the Center of Competence for Digital Signal Processing. He is an author or coauthor of 11 publications.



**Heinrich Niemann** obtained his Electrical Engineering degree and Doctor of Engineering degree from Hannover Technical University, Germany. He worked with the Fraunhofer Institut für Informationsverarbeitung in Technik und Biologie, Karlsruhe, and with the Fachhochschule Giessen in the Department of Electrical Engineering. Since 1975 he has been a professor of computer science at the University of Erlangen–Nuremberg,

where he was dean of the engineering faculty of the university from 1979 to 1981. From 1988 to 2000, he was head of the Knowledge Processing research group at the Bavarian Research Institute for Knowledge-Based Systems (FOR-WISS). Since 1998, he has been a spokesman for a “special research area” with the name of “Model-Based Analysis and Visualization of Complex Scenes and Sensor Data” funded by the German Research Foundation. His fields of research are speech and image understanding and the application of artificial intelligence techniques in these areas. He is on the editorial board of *Signal Processing*, *Pattern Recognition Letters*, *Pattern Recognition and Image Analysis*, and the *Journal of Computing and Information Technology*. He is an 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. He is a member of DAGM, ISCA, EURASIP, GI, IEEE, and VDE and an IAPR fellow.