Marcin Grzegorzek and Heinrich Niemann

Statistical Object Recognition Including Color Modeling

appeared in:
Proceedings of the 2nd International Conference
on Image Analysis and Recognition
Toronto, Canada
pp. 481-489

Statistical Object Recognition Including Color Modeling

Marcin Grzegorzek* and Heinrich Niemann

Chair for Pattern Recognition
University of Erlangen-Nuremberg
Martensstr. 3, 91058 Erlangen, Germany
{grzegorz,niemann}@informatik.uni-erlangen.de

Abstract. In this paper an appearance-based statistical approach for localization and classification of 3-D objects in 2-D color images with real heterogeneous backgrounds is presented. The object feature extraction is done separately for the red, green, and blue channel. We compute six dimensional local feature vectors directly from pixel values in the images using wavelet multiresolution analysis. The first and second component of the feature vectors depend on the pixel values in the red channel, the third and fourth in the green channel, and fifth and sixth in the blue channel. Then we define an object area as a function of 3-D transformations and represent the feature vectors as probability density functions. In the recognition phase we use an algorithm based on maximum likelihood estimation for object localization and classification. Experiments made on a real data set with 39600 images compare the recognition rates for the new algorithm, which uses the color information of objects, with the results in the case of gray level images.

1 Introduction

For many tasks the localization and classification of objects in images is very useful, sometimes even necessary. Algorithms for automatic computational object recognition can be applied for example: to face classification [11], to localization of obstacles on the road with a camera mounted on a driving car, to service robotics [13], to handwriting recognition, and so on. There exist two main approaches for 3-D object recognition: based on results of a segmentation process [5], or directly on the object appearance [4]. The comparison of them can be found in [7]. The appearance-based methods compute feature vectors from pixel values in images without a previous segmentation process [8]. Some of them use only one global feature vector for the whole image (e.g. eigenspace approach [3]), others describe objects with more local features (e.g. neural networks [9]). Many recognition systems do not make use of the color information of objects. For some applications objects are distinguishable very well in the gray level space, for others the recognition algorithm with color modeling takes too much

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

time compared to the improvement of the localization and classification rates. However, one can imagine situations, where two or more objects having totally different colors seem to look identical in gray level images. Their classification is very difficult, and it makes sense to use the color information of objects in this case. For some objects, which have different colors for different views, also the localization is easier in the color space.

In the present work we introduce the color modeling of objects, but in contrast to most approaches (e.g. [1]) we do not use histograms. Six dimensional local feature vectors are computed directly from pixel values (appearance-based approach) using wavelet multiresolution analysis [6] and modeled by density functions [10]. The first and second component of the feature vectors depend on the pixel values in the red channel, the third and fourth in the green channel, and fifth and sixth result from pixel values in the blue channel. The main advantage of the local feature vectors is that a local disturbance only affects the feature vectors in a small region around it. In contrast to this a global feature vector can change totally, if only one pixel in the image varies.

In Sect. 2 the training of statistical object models is presented. Beginning with the computation of the object density value, up to the algorithm for object localization and classification Sect. 3 describes the whole recognition phase. In Sect. 4 the recognition rates for the new algorithm with color modeling are compared with the results in the case of gray level images. Sect. 5 closes our contribution with conclusions.

2 Statistical Object Model

In order to learn a statistical object model $\mathcal{M}_\kappa$ for an object class Ω_κ we take training images of the object Ω_κ in known poses, compute feature vectors in these images (Sect. 2.1), define an object area (Sect. 2.2), and model the feature vectors by density functions (Sect. 2.3).

First we define a set of objects $\Omega = \{\Omega_1, \ldots, \Omega_\kappa, \ldots, \Omega_k\}$ and take training images of them on a dark background in known poses. The original training images are preprocessed by resizing them to RGB images sized $2^n \times 2^n$ pixels, where $n \in \{6, 7, 8, 9\}$. One image $\boldsymbol{f}_{\kappa,i}$ for each object class Ω_κ is used as a reference image. By pose of an object in the image $\boldsymbol{f}_{\kappa,j}$ we denote the 3-D transformation (translation and rotation) that maps the object in the reference image $\boldsymbol{f}_{\kappa,i}$ to the object in $\boldsymbol{f}_{\kappa,j}$. The 3-D transformation can be described by a translation $\boldsymbol{t} = (t_x, t_y, t_z)^{\mathrm{T}}$ and a rotation $\boldsymbol{\phi} = (\phi_x, \phi_y, \phi_z)^{\mathrm{T}}$. The x- and y-axes of the world coordinate system lie in the image plane, and the z axis is orthographic to the image plane (Fig. 2). A rotation about the x- and y-axes as well as a translation along the z-axis (scaling) changes the size and appearance of the object in the image. These are the so called external transformation parameters ($t_{ext} = t_z$ and $\boldsymbol{\phi}_{ext} = (\phi_x, \phi_y)^{\mathrm{T}}$). The remaining transformation parameters are called internal and do not change the object size and appearance. Up to the end of Sect. 2 the number of the object class κ is omitted, because the training of the statistical object model is identical for all object classes.

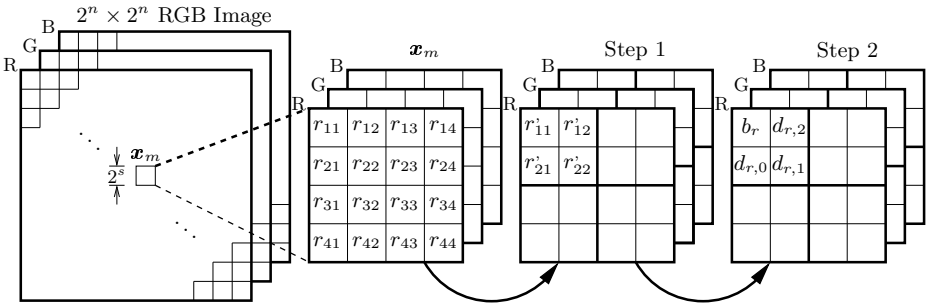

Fig. 1. Computation of a feature vector on a grid point $\boldsymbol{x}_m$ for the scale $s = 2$. r'_{ij} are calculated by horizontal and vertical low pass filtering of r_{ij} and resolution reduction by factor 0.5. The final coefficients result from r'_{ij} as follows: b_r - low pass horizontal and low pass vertical, $d_{r,0}$ - low pass horizontal and high pass vertical, $d_{r,1}$ - high pass horizontal and high pass vertical, $d_{r,2}$ - high pass horizontal and low pass vertical.

2.1 Feature Vectors

For the feature extraction we divide each preprocessed image $\boldsymbol{f}$ into squares of size $2^s \times 2^s$ ($s \leq n$) pixels, and set in their centers grid points $\boldsymbol{x}_m$. On all of these $2^{n-s} \times 2^{n-s}$ grid points six dimensional local feature vectors with the wavelet multiresolution analysis [6] are computed:

$$\boldsymbol{c}_m = \boldsymbol{c}(\boldsymbol{x}_m) = (c_{m,r,1}, c_{m,r,2}, c_{m,g,1}, c_{m,g,2}, c_{m,b,1}, c_{m,b,2})^{\mathrm{T}} \quad . \tag{1}$$

The choice of the wavelet transformation follows from the experimental results. The components $c_{m,r,1}$ and $c_{m,r,2}$ depend on the pixel values in the red channel, $c_{m,g,1}$ and $c_{m,g,2}$ in the green channel, and $c_{m,b,1}$ and $c_{m,b,2}$ in the blue channel. We explain their computation in detail only for the red channel $(c_{m,r,1}, c_{m,r,2})^{\mathrm{T}}$, because for the other channels as well as for gray level images it is done in the same way. We perform s-times the wavelet multiresolution analysis for the red channel values in the local neighborhood of $\boldsymbol{x}_m$ (neighborhood size: $2^s \times 2^s$ pixels) using Johnston 8-TAB wavelets [2]. The component $c_{m,r,1}$ of the feature vector $\boldsymbol{c}_m$ is given by:

$$c_{m,r,1} = \ln |b_{r,s,m}| \quad , \tag{2}$$

and $c_{m,r,2}$ can be calculated with the equation:

$$c_{m,r,2} = \ln(|d_{r,0,s,m}| + |d_{r,1,s,m}| + |d_{r,2,s,m}|) \quad . \tag{3}$$

$b_{r,s,m}$ is the low pass coefficient and $d_{r,0\ldots2,s,m}$ result from combinations of low pass and high pass filtering. An illustration of the feature vector computation for $s = 2$ can be seen in Fig. 1 (indices m and s are omitted). In Sect. 4.2 we compare the results for color and gray level images. In the case of gray level images two dimensional feature vectors $\boldsymbol{c}_m = (c_{m,1}, c_{m,2})^{\mathrm{T}}$ computed according to (2) and (3) are used [10].

2.2 Object Area

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

$$\widehat{\xi}_m(\boldsymbol{\phi}_{ext,t}, t_{ext,t}) = \begin{cases} 1, & \text{if } \; c_{m,\{r \vee g \vee b\},1}(\boldsymbol{\phi}_{ext,t}, t_{ext,t}) \geq S_t \\ 0, & \text{otherwise} \end{cases} \quad . \tag{4}$$

S_t is chosen manually. If for all color channels the first feature vector coefficient ($c_{m,r,1}$, $c_{m,g,1}$, $c_{m,b,1}$) computed according to (2) is less than S_t, $\boldsymbol{c}_m$ does not belong to the object. 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 $\xi_m(\boldsymbol{\phi}_{ext}, t_{ext})$ using values of $\widehat{\xi}_m(\boldsymbol{\phi}_{ext,t}, t_{ext,t})$ by interpolation with trigonometric functions. The set of feature vectors belonging to the object for the given external pose $(\boldsymbol{\phi}_{ext}, t_{ext})$ (called object area $O(\boldsymbol{\phi}_{ext}, t_{ext})$) can now be determined with the following rule:

$$\xi_m(\boldsymbol{\phi}_{ext}, t_{ext}) \geq S_O \Longrightarrow \boldsymbol{c}_m(\boldsymbol{\phi}_{ext}, t_{ext}) \in O(\boldsymbol{\phi}_{ext}, t_{ext}) \quad . \tag{5}$$

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

2.3 Density Functions of the Feature Vectors

All feature vectors computed in the training phase according to (1), (2), and (3) are interpreted as random variables. The object feature vectors are modeled with the normal distribution [10]. For each object feature vector we compute a mean value vector $\boldsymbol{\mu}_m$ and standard deviation vector $\boldsymbol{\sigma}_m$. The density of the object feature vector can be written as:

$$p(\boldsymbol{c}_m) = p(\boldsymbol{c}_m | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m, \boldsymbol{\phi}, \boldsymbol{t}) = \prod_{i \in \{r,g,b\}} \prod_{j=0}^{2} p(c_{m,i,j} | \mu_{m,i,j}, \sigma_{m,i,j}, \boldsymbol{\phi}, \boldsymbol{t}) \quad . \tag{6}$$

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

3 Localization and Classification

After a corresponding object model $\mathcal{M}_\kappa$ was created for each object class Ω_κ, we can localize and classify objects in test images. At the beginning each test image is preprocessed and feature vectors are computed according to (1), (2), and (3) with the same method as in the training phase (Sect. 2.1). Then we start our localization and classification algorithm based on the maximum likelihood estimation (Sect. 3.2), which maximizes the object density value (Sect. 3.1).

3.1 Object Density Value

In order to compute the object density value for the class Ω_κ in pose $(\boldsymbol{\phi}, \boldsymbol{t})$ for the given test image $\boldsymbol{f}$ we determine the set of feature vectors that belong to the object $C = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_M\}$ (object area $O_\kappa(\boldsymbol{\phi}, \boldsymbol{t})$, Sect. 2.2) according to (5) and compute their values using equations (1), (2), and (3). Then we compare the calculated object feature vectors with the corresponding density functions (6) stored in the object model $\mathcal{M}_\kappa$ and determine density values for these vectors $(p(\boldsymbol{c}_1), p(\boldsymbol{c}_2), \ldots, p(\boldsymbol{c}_M))$. The density value of object Ω_κ in pose $(\boldsymbol{\phi}, \boldsymbol{t})$ for the given test image $\boldsymbol{f}$ is given by:

$$p(C|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t}) = \prod_{i=0}^{M} \max\{p(\boldsymbol{c}_i), p_b\} \quad . \tag{7}$$

$\boldsymbol{B}_\kappa$ comprehends the trained mean value vectors and standard deviation vectors from $\mathcal{M}_\kappa$ and p_b is the background density value (Sect. 2.3).

3.2 Recognition Algorithm

The localization and classification algorithm is realized with maximum likelihood estimation [12] and can be described with the following equation:

$$(\widehat{\kappa}, \widehat{\boldsymbol{\phi}}, \widehat{\boldsymbol{t}}) = \underset{\kappa}{\operatorname{argmax}}\{\underset{(\boldsymbol{\phi}, \boldsymbol{t})}{\operatorname{argmax}}\, G(p(C|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t}))\} \quad . \tag{8}$$

$\widehat{\kappa}$ is the classification result and $(\widehat{\boldsymbol{\phi}}, \widehat{\boldsymbol{t}})$ is the localization result. First the object density (normalized by G) is maximized according to the pose parameters $(\boldsymbol{\phi}, \boldsymbol{t})$ and then to the class κ. The norm function G is defined by:

$$G(p(C|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})) = \sqrt[M]{p(C|\boldsymbol{B}_\kappa, \boldsymbol{\phi}, \boldsymbol{t})} \quad . \tag{9}$$

M is the number of feature vectors belonging to the object area $O_\kappa(\boldsymbol{\phi}, \boldsymbol{t})$ (Sect. 3.1). This norm function reduces the dependency between the maximization result and the object area size.

4 Experiments and Results

We verified our approach on a 3D-REAL-ENV image data base (Sect. 4.1). The color modeling of objects brings the most profit in very heterogeneous environments compared to the algorithm for gray level images (Sect. 4.2).

4.1 Image Data Base

3D-REAL-ENV (Image Data Base for 3-D Object Recognition in Real World Environment) consists of 10 objects depicted in Fig. 2. The experiments were done using images of size 256×256 pixels. The pose of an object is defined

Fig. 2. 10 object classes used for experiments. In the first row examples of test images with "more heterogeneous" backgrounds; from left: bank cup, toy fire engine, green puncher, siemens cup, nizoral bottle. In the second row examples of test images with "less heterogeneous" backgrounds; from left: toy passenger car, candy box, stapler, toy truck, white puncher. In the right upper corner the coordinate system for the object pose definition is shown.

with external rotations and internal translations $(\phi_x, \phi_y, t_x, t_y)^{\mathrm{T}}$ (Fig. 2). For the training we took 3360 images of each object with two different illuminations. The objects were put on a turntable ($0° \leq \phi_{table} < 360°$) and a robot arm with a camera was moved from horizontal to vertical ($0° \leq \phi_{arm} \leq 90°$). The angle between two adjacent training viewpoints amounts to 4.5°. For the tests 2000 images with homogeneous, 2000 images with "less heterogeneous", and 2000 with "more heterogeneous" backgrounds were taken. In the test images with "less heterogeneous" backgrounds the objects are easier to distinguish from the background than in the scenes with "more heterogeneous" backgrounds. The object poses and the illumination in the recognition phase are different from the training viewpoints and illuminations. For the test images with heterogeneous backgrounds we used more than 200 different backgrounds.

4.2 Localization and Classification Rates

We count a localization result as correct, if the error for the external rotations $(\phi_x, \phi_y)^{\mathrm{T}}$ is not larger than 15° and the error for the internal translations $(t_x, t_y)^{\mathrm{T}}$ is not larger than 10 pixels. The feature extraction for the experiments was made for the scale $s = 3$ of the wavelet multiresolution analysis (Sect. 2.1). Fig. 3 presents the recognition rates depending on the distance of the training views for test images with homogeneous, "less heterogeneous", and "more heterogeneous" backgrounds. Table 1 contains the recognition rates for 4.5° distance of training views. The color modeling brings the most improvement of the localization and classification rates for test images with "more heterogeneous" backgrounds. For scenes with homogeneous backgrounds the algorithm for gray level images works very well, and it is not necessary to use the color information of objects. Object localization and classification takes $3.6s$ in one gray level image, and $7s$ in one color image on Pentium 4, 2.66 MHz, 512 MB RAM.

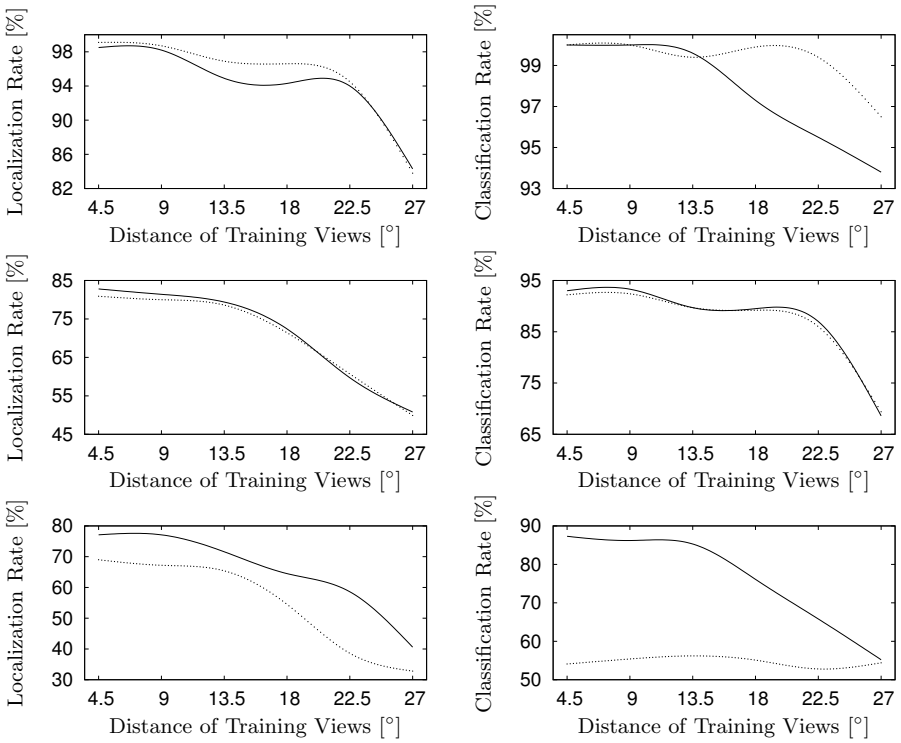

Fig. 3. Localization and classification rates depending on the distance of the training views for 2000 test images with homogeneous (first row), 2000 test images with "less heterogeneous" (second row), and 2000 test images with "more heterogeneous" backgrounds (third row). (— color images; $\cdots$ gray level images).

5 Conclusions

In this article a powerful statistical appearance-based approach for 3-D object recognition in 2-D images with real heterogeneous backgrounds is presented. After feature extraction, which is done separately for the red, green, and blue channel, we define an assignment function, which assigns the features to the object or to the background, and statistically model them by density functions. In the recognition phase we use an algorithm based on the maximum likelihood estimation for localization and classification of objects. Results show that the color modeling brings a great improvement of the recognition rates in heterogeneous environments. On the other side we proved that for scenes with homogeneous backgrounds the use of gray level images is sufficient.

In the future we will try to obtain better recognition rates by transformation of the RGB images into other color spaces. We will also consider the case of multi-object scenes with context dependencies.

Table 1. Recognition rates for 4.5° distance of training views for 2000 test images with homogeneous, 2000 with "less heterogeneous", and 2000 with "more heterogeneous" backgrounds.

Distance of Training Views 4.5°	**Localization**			**Classification**		
	Hom. Back.	**Less Het. Back.**	**More Het. Back.**	**Hom. Back.**	**Less Het. Back.**	**More Het. Back.**
Color Images	98.5%	82.2%	77.1%	100%	93.0%	87.3%
Gray Level Images	99.1%	80.9%	69.0%	100%	92.2%	54.1%

References

1. P. Chang and J. Krumm. Object recognition with color cooccurrence histograms. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 498–504, Fort Collins, USA, June 1999. IEEE Computer Society.
2. C. Chui. *An Introduction to Wavelets*. Academic Press, San Diego, USA, 1992.
3. Ch. Gräßl, F. Deinzer, and H. Niemann. Continuous parametrization of normal distribution for improving the discrete statistical eigenspace approach for object recognition. In V. Krasnoproshin, S. Ablameyko, and J. Soldek, editors, *Pattern Recognition and Information Processing 03*, pages 73–77, Minsk, Belarus, Mai 2003.
4. 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, April 2004.
5. 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*, pages 9–15, Acapulco, Mexico, August 2003.
6. S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.
7. J. Mundy, A. Liu, N. Pillow, A. Zisserman, S. Abdallah, S. Utcke, S. Nayer, and C. Rothwell. An experimental comparison of appearance and geometric model based recognition. In J. Ponce, A. Zisserman, and M. Hebert, editors, *Object Representation for Computer Vision II*, pages 247–269, Cambridge, UK, April 1996. Springer Verlag.
8. H. Murase and S. K. Nayar. Visual learning and recognition of 3-d objects from appearance. *International Journal of Computer Vision*, 14(1):5–24, January 1995.
9. S. Park, J. Lee, and S. Kim. Content-based image classification using a neural network. *Pattern Recognition Letters*, 25(3):287–300, February 2004.
10. M. Reinhold. *Robuste, probabilistische, erscheinungsbasierte Objekterkennung*. Logos Verlag, Berlin, Germany, 2004.
11. D. Terzopoulos, L. Yuencheng, and M. Vasilescu. Model-based and image-based methods for facial image synthesis, analysis and recognition. In *Automatic Face and Gesture Recognition 2004*, pages 3–8, Seoul, Korea, Mai 2004.
12. A. R. Webb. *Statistical Pattern Recognition*. John Wiley & Sons Ltd, Chichester, England, 2002.
13. B. You, M. Hwangbo, S. Lee, S. Oh, Y. Kwon, and S. Lim. Development of a home service robot issac. In *Intelligent Robots and Systems 2003*, pages 2630–2635, Las Vegas, USA, October 2003.