

Contour-based Object Detection in Range Images

Stefan Stiene, Kai Lingemann, Andreas Nüchter, Joachim Hertzberg
University of Osnabrück
Knowledge-Based Systems Research Group
Albrechtstraße 28 D-49069 Osnabrück, Germany
sstiene@uos.de

Abstract

This paper presents a novel object recognition approach based on range images. Due to its insensitivity to illumination, range data is well suited for reliable silhouette extraction. Silhouette or contour descriptions are good sources of information for object recognition. We propose a complete object recognition system, based on a 3D laser scanner, reliable contour extraction with floor interpretation, feature extraction using a new, fast Eigen-CSS method, and a supervised learning algorithm. The recognition system was successfully tested on range images acquired with a mobile robot, and the results are compared to standard techniques, i.e., Geometric features, Hu and Zernike moments, the Border Signature method and the Angular Radial Transformation. An evaluation using the receiver operating characteristic analysis completes this paper. The Eigen-CSS method has proved to be comparable in detection performance to the top competitors, yet faster than the best one by an order of magnitude in feature extraction time.

1. Introduction

A basic part of perception is to learn, detect and recognize objects, which has to be done with limited resources. Especially in mobile robotics there is a need for online capable object detection and environment interpretation. Meaning becomes inevitable if the robot has to interact with its environment. The robot is then able to reason about the objects; its knowledge becomes inspectable and communicable.

A wide variety of techniques for object detection have been developed. They can be classified into template matching, feature matching and appearance based methods. Contour-based algorithms have been of particular interest since the middle of the last century, resulting in a multiplicity of available methods, e.g., moment, ART and CSS based methods, as well as Fourier descriptors. However,

contour extraction has to cope with inherent problems originating mainly from changing lighting conditions and environment texturing when applied to real images. Using depth images avoids these problems but introduces other challenges, e.g., the reliable generation of range images. The segmentation of objects in natural environments is simplified by using range images, since they are distinguishable iff the spatial distance between object and background is sufficiently high. The contour extraction of objects from range images is relatively simple because scan points belonging to the same object show smooth changes in their distance values. At object borders, discontinuities emerge that cause an edge in the range image that in turn is identified by segmentation algorithms. Nevertheless, all standing objects cannot easily be separated from floor. We overcome this problem by introducing a novel technique to identify the ground based on local gradients. Contour or silhouette based object recognition in range images is view dependent. Hence, view invariant recognition is reached by generating several object views from the original object and training different classifiers.

This paper presents a complete object recognition system for mobile robots. The system consists of four major parts: First, a reliable, cost-effective 3D laser range finder for generating range images. Second, contour extraction based on adaptive thresholding combined with floor detection and morphological opening. Third, feature extraction using a new Eigen-Curvature Scale Space (Eigen-CSS) method. Hereby we extend the method proposed by Drew, Lee and Rova [8] by generating several Eigen-spaces. Fourth, the classification using support vector machines (SVMs) with different kernels.

To provide some background for assessing the quality of the Eigen-CSS method, this paper evaluates contour based object recognition in range images with standard methods, namely, geometric features, Hu and Zernike moments, Angular Radial Transformations (ART) and the Border Signature method. In the following paragraph we give a sketch of the state of the art in object recognition in range images,

focusing on contour recognition. Section 2 describes the contour extraction, followed by the feature extraction and a brief description of SVMs. The experimental results and comparisons with the reference methods are given in section 5. Section 6 concludes.

1.1. State of the Art

Campbell and Flynn review object detection algorithms in range images and classify them into three approaches [6]: appearance-based recognition, recognition from 2D silhouettes, and free form object recognition:

Appearance-based recognition represents the object in a high dimensional space and uses principal component analysis on a set of training image data. This recognition approach has already been tested on range images (yielding the so-called eigenshapes) [5].

According to Campbell and Flynn and to our knowledge, recognition from 2D silhouettes extracted from range images has not been realized before. However, shape recognition in images is well researched for the past decades. A wide range of contour describing features does exist, common techniques being Fourier descriptors [7], moments [10], geometric features, and contour functions. Beside these features, the Angular Radial Transformation (ART) [3] as a region based, and the Curvature Scale Space (CSS) [13] as a contour based feature extraction method are often used; due to their good performance, they are standardized in the MPEG-7 multimedia content description language [3].

Most of the work has been done in the area of free form object recognition and classification in 3D range data. Johnson and Hebert use the well-known ICP algorithm [2] for registering 3D shapes into a common coordinate system [11]. The necessary initial guess of the ICP algorithm is done by detecting the object with spin images [11]. Besides spin images, several surface representation schemes are in use for computing an initial alignment. Stein and Medioni presented the notion of “splash” to represent the normals along a geodesic circle of a center point, which is the local Gauss map for 3D object recognition with a database [14]. Ashrock et al. have proposed a pairwise geometric histogram to find corresponding facets between two surfaces that are represented by triangle meshes [1]. Harmonic maps and their use in surface matching have been used by Zhang and Hebert [19]. Recently, Sun and colleagues have suggested so-called “point fingerprints”: They compute a set of 2D curves that are projections of geodesic circles onto the tangent plane and compute similarities between them [15]. All these approaches take the local geometry of the surfaces into account, i.e., meshes.

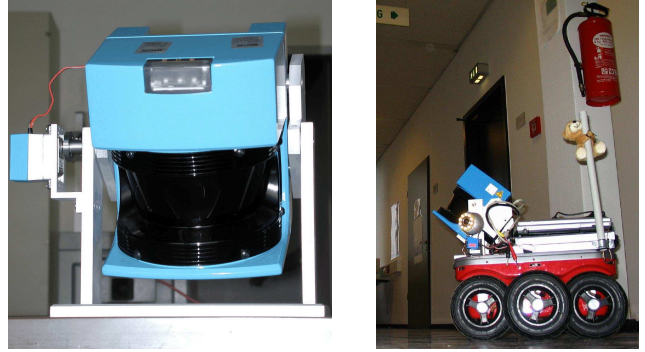


Figure 1. Left: The 3D laser scanner is built of 2D laser scanner and a servo motor step-rotating the scanner. Right: Mounted on the robot Kurt3D.

1.2 Range Image Generation

The data acquisition in our experiments was performed with the AIS 3D laser range finder (Fig. 1) [16] mounted on the autonomous mobile robot Kurt3D. It is built on the basis of a 2D range finder by extension with a mount and a small servomotor step-rotating the scanner around a horizontal axis. The area of $180^\circ(h) \times 120^\circ(v)$ is scanned with different horizontal (181, 361, 721 pts) and vertical (250, 500 pts) resolutions. The depth information of the 3D data is visualized as a gray-scale image: each scan point is assigned with a gray value, according to its distance to the scanner position,

Afterwards a range image is rendered under a specific view. Since the range image’s resolution is in general greater than the scanner’s, the program interpolates between the scan points’ gray values to assign a gray value to each pixel. A resulting range image is shown in Fig. 4 (left).

2 Contour Extraction

When segmenting the resulting range image, a problem arises with objects standing on the floor. For example, the feet of a human have the same gray value, i.e., distance values, as the floor at the point he is standing on. The feet and the floor form only a crease edge, no jump edge. This problem is solved by segmenting the floor in the range data prior to generating the range image:

Based on the idea of Wulf et al. [18] we designed an algorithm for labelling floor points in 3D scans. This is done by computing the gradient between a point $\mathbf{p}_{i,j} = (\phi_i, r_{i,j}, z_{i,j})$, given in a cylindrical coordinate system, and its nearest neighbor within the vertical sweep plane, i.e., a search region around ϕ_i , according to the following equa-

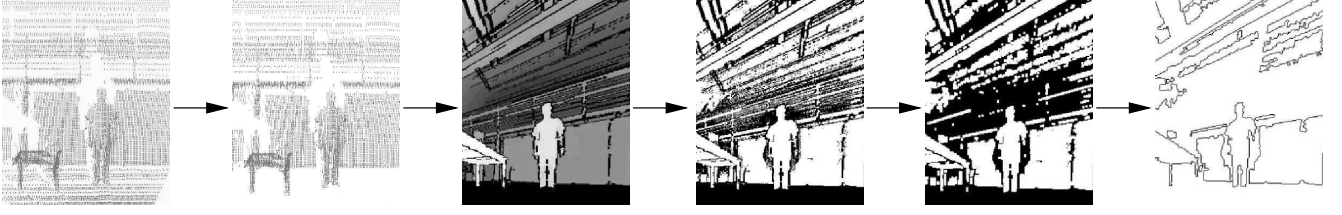


Figure 2. Cascade for contour extraction. From left to right: (1) Scanned scene as point cloud. (2) Point cloud with removed floor points. (3) Generated range image without interpolation at jump edges. (4) Binarized image using adaptive thresholding. (5) Morphological opening of the image. (6) Final contour representation.

tion (cf. Fig. 3):

$$\alpha_{i,j} = \arctan \left(\frac{z_{i,j} - z_{i,j-k}}{r_{i,j} - r_{i,j-k}} \right)$$

with

$$-\frac{1}{2}\pi \leq \alpha_{i,j} < \frac{3}{2}\pi.$$

In comparison with a fixed threshold τ (here: $\tau = 20^\circ$) each 3D point is assigned to one of these three groups:

1. $\alpha_{i,j} < \tau$: $\mathbf{p}_{i,j}$ is a ground point
2. $\tau \leq \alpha_{i,j} \leq \pi - \tau$: $\mathbf{p}_{i,j}$ is an object point
3. $\pi - \tau < \alpha_{i,j}$: $\mathbf{p}_{i,j}$ is a ceiling point

This labelling proved to be robust against uneven and non-horizontal ground and against data jitter.

In order to enhance segmentation of objects against their background, we skip the interpolation as described in Sec. 1.2 if the range difference between two neighboring points is above a fixed threshold. This method yields a range image in which each object that has a sufficient distance to its background is enclosed by a black contour. Fig. 4 (right) shows the result.

The actual contour extraction is done after applying a binarization filter with an adaptive threshold [12]. Each image pixel is set to 1 or 0 in comparison with a local threshold computed with respect to the local neighborhood. In areas with homogeneously distributed gray values, this threshold is close to the pixel's grey value, leading to a random assignment of 1 or 0. Therefore, the local, adaptive threshold is combined with a global threshold that is subtracted from the center pixel before comparison. The resulting binary image typically contains many small structures, leading to a large number of contours and a slow object detection system. We overcome this problem by convolving the image with the nonlinear filter “morphological opening”, resulting

in an image without small structures. As an additional benefit, structures get separated that are connected only by a few pixels. Finally, contours are extracted from the binarized image using a contour following algorithm. Fig. 2 shows the overall image contour extraction process.

3 Feature Extraction

Here we present the Eigen-CSS feature extraction method to describe a contour. This method improves the original CSS method by Mokhtarian [8,13]. The CSS representation interprets the contour as a curve depending on the parameter u . The curve is repeatedly smoothed by convolution with a gauss function ($\otimes g(u, \sigma)$) of increasing standard deviation σ . In each iteration step the curvature, depending on σ , is computed according to the following equation:

$$\kappa(u, \sigma) = \frac{X_u(u, \sigma)Y_{uu}(u, \sigma) - X_{uu}(u, \sigma)Y_u(u, \sigma)}{(X_u(u, \sigma)^2 + Y_u(u, \sigma)^2)^{1.5}}$$

with

$$X_u(u, \sigma) = \frac{\partial}{\partial u}(x(u) \otimes g(u, \sigma)) = x(u) \otimes g_u(u, \sigma)$$

$$X_{uu}(u, \sigma) = \frac{\partial^2}{\partial u^2}(x(u) \otimes g(u, \sigma)) = x(u) \otimes g_{uu}(u, \sigma).$$

$Y_u(u, \sigma)$ and $Y_{uu}(u, \sigma)$ are defined analogously. The CSS representation is obtained by plotting the solution of $\kappa(u, \sigma) = 0$, using the curve parameter u as ordinate and σ as abscissa, as shown in Fig. 5 (top).

The main problem with feature extraction from the CSS representation is that a rotated contour causes a horizontally shifted representation. To solve this problem Drew, Lee and Rova proposed in 2005 the Eigen-CSS feature extraction method [8], consisting of three simple techniques: marginal-sums, phase correlation and singular value decomposition. The first two are used to solve the shift problem, the latter one is used to map the rotation invariant feature vector into its eigenspace.

The marginal-sums are used to transfer the CSS representation into a rotation invariant column-sum vector \mathbf{c} and a rotation sensitive row-sum vector \mathbf{r} . The rotation invariance of \mathbf{r} is obtained by using phase correlation, i.e., converting the vector to the frequency domain, calculating the magnitude as a function of frequency, and transforming the results back to the spatial domain, according to

$$\tilde{\mathbf{r}} = |F^{-1}(|F(\mathbf{r})|)|.$$

The rotation invariant row- and column-sum vectors are combined into a contour describing feature vector \mathbf{x} (cf. Fig. 5) by $\mathbf{x} = [\tilde{\mathbf{r}}\mathbf{c}]^T$. The feature vector \mathbf{x} is mapped in its eigenspace using the following procedure:

1. Determine the feature vector \mathbf{x} for a fixed number n of examples (from one object class).
2. Subtract from each vector \mathbf{x} its mean value $\bar{\mathbf{x}}$.
3. Construct a matrix \mathbf{X} as

$$\mathbf{X} = [\mathbf{x}_1 - \bar{\mathbf{x}}_1, \mathbf{x}_2 - \bar{\mathbf{x}}_2, \dots, \mathbf{x}_n - \bar{\mathbf{x}}_n].$$

Each vector \mathbf{x} normalized by its mean corresponds to a column in the matrix \mathbf{X} .

4. Execute Singular Value Decomposition (SVD) on \mathbf{X} , i.e.,

$$\mathbf{X} = \mathbf{U}\mathbf{W}\mathbf{V}^T$$

5. Reduce the matrix \mathbf{U} to the columns j , whose corresponding singular values w_j are unequal to zero.
6. Form the eigenspace from the column vectors of the reduced matrix \mathbf{U} .
7. Project \mathbf{x} in its eigenspace through a multiplication with the transposed matrix \mathbf{U} , i.e.,

$$\mathbf{u} = \mathbf{U}^T \mathbf{x}.$$

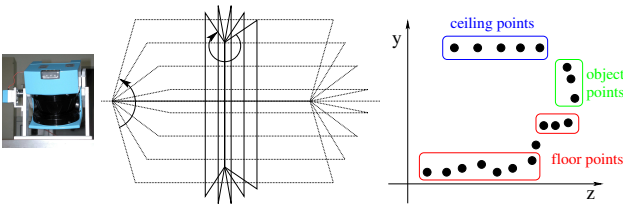


Figure 3. Left: 3D scan planes due to the rotation of the 2D laser range finder (tilt around horizontal axis) vs. 3D sweep planes (turn around vertical axis). Right: Interpretation example, based on scan points of one vertical sweep plane.

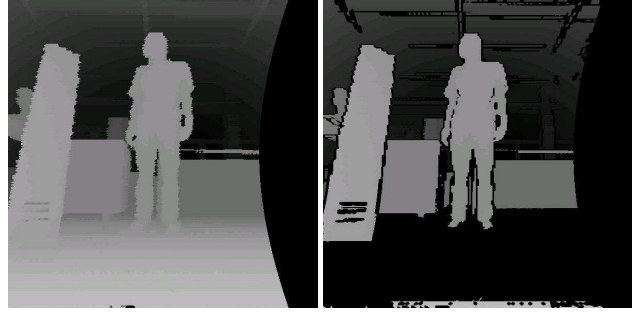


Figure 4. Left: Range image generated with all 3D scan points. Right: Range image with removed ground points and without gray value interpolation at range jumps.

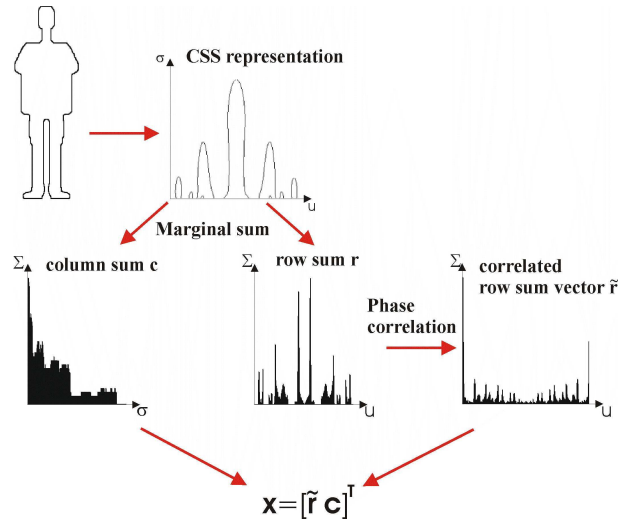


Figure 5. Construction of the Eigen-CSS feature vector.

In contrast to the method proposed in [8] our algorithm does not group phase correlated feature vectors from different object classes in one matrix \mathbf{X} . Thus, we use a matrix that consists only of feature vectors from one object class, and every object has its own eigenspace and matrix \mathbf{U} .

Following the results in [8], we start the Eigen-CSS procedure with a smoothed version of the CSS object representation, using a standard deviation $\sigma = 5$. This eliminates small peaks in the CSS representation. To ensure that all feature vectors have a constant length for the contours, the procedure also starts with a normalized contour, i.e., a fixed parameter length is used ($u = 100$). This normalization affects the ordinate. The abscissa has to be normalized, too, since different contours yield different curvatures resulting in variable heights of the representations. For this normalization the abscissa in each CSS representation is padded to 100, before computing the marginal sum vectors.

4 Object Learning and Classification

Object learning and classification using the constructed feature vectors is done with Support Vector Machines (SVMs). SVMs are supervised learning methods used for classification and regression. When used for classification, the SVM algorithm creates a hyperplane that separates the data into two classes with the maximum margin. Given training examples labelled either “yes” or “no”, a maximum margin hyperplane is identified which splits the two regions such that the distance between the hyperplane and the support vectors (the margin) is maximized. The parameters of the hyperplane are derived by solving a quadratic programming optimization problem.

The original hyperplane algorithm, restricted to linear classification problems only, was augmented by Boser et al. to allow non-linear classification by applying the kernel trick [4]: Replacing every dot product by a non-linear kernel function allows the algorithm to fit the maximum margin hyperplane in the transformed feature space and leads to a non-linear separation of the data in the original input space. Our algorithm uses three different kernel functions, namely, a radial kernel function (here: $\sigma = 1$):

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|}{2\sigma}\right), \quad (1)$$

a polynomial kernel (here: $d = 2$)

$$k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}')^d \quad (2)$$

and a linear kernel

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'. \quad (3)$$

5 Experimental Results

Results of our proposed classification system are shown in Fig. 6. Our method is evaluated against the following five standard methods: geometric features, Hu [10] and Zernike moments [17], the Angular Radial Transformation [3] and the Border Signature algorithm.

Geometric features with Hu moments: The 13 dimensional vector consists of 6 geometric features: area-perimeter ratio, aspect ratio, rectangularity, eccentricity, orientation and radii ratio. These features are combined with 7 Hu moments to a feature vector.

Zernike moments: Zernike moments map the contour onto the unit circle using orthogonal Zernike polynomials. We compute the first 42 invariant Zernike moments.

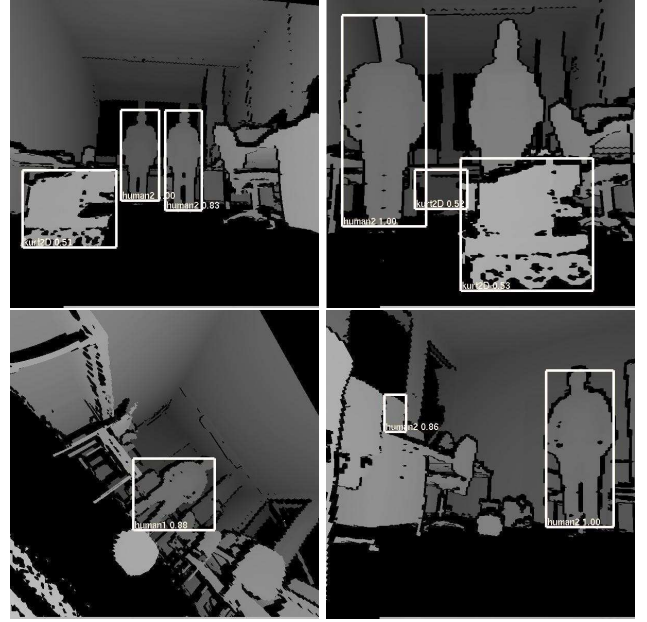


Figure 6. Example scenes with detected objects (human & robot Kurt3D). The 4th picture shows a false detection of a wrongly recognized human above the table. See the text for explanations.

Angular Radial Transformation: The ART is a region orientated feature. Due to its performance it is standardized in the MPEG-7 language. Like Zernike moments, it maps the contour onto the unit circle, but uses simpler basis functions. Like proposed in the MPEG-7 standard, we compute 36 coefficients using the first to normalize the others.

As an alternative to the ART as defined in MPEG-7, orthogonal ARTs that feature orthogonal radial components are evaluated in this paper, too.

Border Signature: The Border Signature method divides the area enclosed by the contour into radial segments, whose common origin is the contour’s center of gravity. The feature is the contour points average distance in each segment. These distances are normalized over the whole contour area. We use 32 segments which lead to the same number of contour describing features.

Eigen-CSS: The feature vector is the Eigen-CSS vector as described above. We do not truncate the matrix U , therefore gaining a 200 dimensional feature vector.

The different feature vectors are classified using an SVM. To compare the classifier’s performance we use the receiver operating characteristics (ROC) Analysis [9].

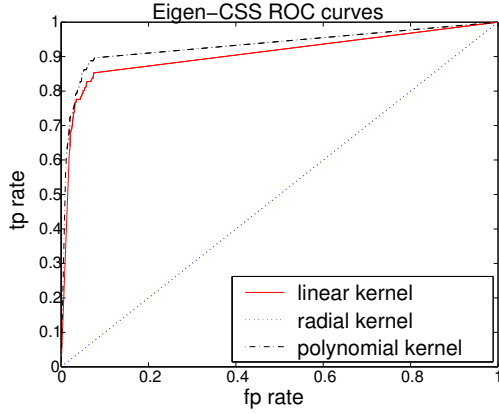


Figure 7. ROC curves for three classifiers trained with linear, radial and polynomial kernel.

In the training phase, test scans are taken from each object. We generate three range images from each scan under different views. Afterwards every range image is segmented, then stored as positive example in case of correct segmentation. Each classifier was trained using 200 positive and 700 negative examples.

For performance evaluation, we take test scans for each object with 100 positive and about 1800 negative examples. Only one range image is generated from one scan. We mark the positive contours in the range image, then apply the corresponding classifiers and count the true positive (tp), false positive (fp), true negative (tn) and false negative (fn) classifications. The ROC metrics TP and FP rate are calculated according to

$$\text{TP rate} = \frac{tp}{tp + fn} \quad \text{FP rate} = \frac{fp}{fp + tn}.$$

Afterwards we produce a ROC curve for each classifier to determine the best kernel for the respective feature extraction method. The performance is measured by the AUC metric, i.e., the area below the ROC curve (cf. Fig. 7).

Table 1 shows the results for a human standing frontal to the scanner with the legs apart. For this setup, orthogonal ART provides the best classification results, being followed closely by the methods c2 – c5, while c1, using geometric features, shows significant drawbacks. Note the extensive differences in computation time. Combining the classifiers c1 – c6, as done for classifier c7, proved to be not profitable, due to the dominating high performance of the ART method. The combination is done like a classifier voting as follows:

1. Apply each classifier to the current contour. The result is a number that encodes the degree of membership for the object class.

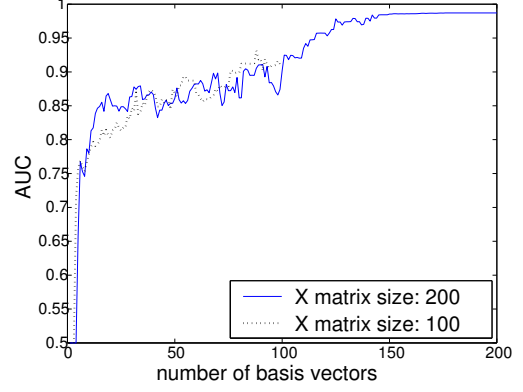


Figure 8. AUC value depending on the number of basis vectors. Each point in this graph belongs to a different classifier. Solid line: Using 200 positive and 700 negative examples for training the SVMs and all 200 positive have been used to create the eigenspace. Dotted line: 200 positive and 700 negative examples for training the SVMs and 100 positive have been used to create the eigenspace.

2. In case of positive classification add the result of the SVM for the contours. In case of negative subtract the result.
3. After all classifiers were applied to the contour, assign the contour to the membership class with the highest overall value.

The computing time for this combination is the sum of the computing times for each classifier. However, adding further objects only increases the computational time slightly, since the feature vectors have already been extracted. The computational times of the non-combined methods as well as their classification results are shown in table 2.

In addition to the classifier's performance, the Eigen-CSS algorithm has been evaluated in terms of the number of eigenvectors. 200 classifiers, i.e., SVMs, using again 200 positive and 700 negative examples were learned, each of these classifiers uses a different number of basis vectors. Thus, the number of basis vectors for matrix \mathbf{U} was truncated to values between 1 and 200 (cf. step 5 on page 4). We have computed the ROC curve for each classifier and have determined the according AUC value. Fig. 8 shows the the resulting AUC curve. The performance increases very fast between 5 and 20 basis vectors. From 20 to 100 basis vectors there is only a slight improvement. Between 100 and 150 basis vectors there is another increase of performance from 0.85 to 0.98, followed by a constant performance.

The dotted line in Fig. 8 represents classifiers for which the matrix \mathbf{X} is the concatenation of a reduced number of

Table 1. Classification performance of the classifiers using different contour describing features; c7 is the classifier combined from c1 – c6. Each classifier was trained with 200 positive and 700 negative examples, the best SVM has been selected. The stated time is the duration needed to classify one contour on a standard Pentium-IV-3000.

extracted feature	c1	c2	c3	c4	c5	c6	c7
geometric features ¹	×	–	–	–	–	–	×
Hu moments	7	–	–	–	–	–	7
Zernike moments	–	42	–	–	–	–	42
CSS	–	–	×	–	–	–	×
U matrix examples	–	–	200	–	–	–	200
number of basis vectors	–	–	200	–	–	–	200
Border signature	–	–	–	32	–	–	32
ART	–	–	–	–	35	–	35
ortho. ART	–	–	–	–	–	35	35
time [ms]	7	187	4	0.4	32	32	Σ^2
optimal kernel	poly.	poly.	lin.	rad.	poly.	rad.	(comb.)
optimal threshold	0.98	0.15	0.76	0.97	0.77	0.78	0.40
accuracy	0.964	0.982	0.989	0.963	0.990	0.995	0.996
AUC	0.685	0.980	0.986	0.990	0.991	0.999	0.999

Table 2. Classification results of 30 example images, containing one positive example each. The mean computation time per image is composed of the rendering time (286 ms), contour extraction (193 ms), feature extraction (see table) and classification (12.6 ms).

	<i>tp</i>	<i>fp</i>	feature extraction time [ms]
geometric features	22	1	146.4
Eigen-CSS	27	11	87.4
Zernike moments	23	2	3876.4
ART	28	3	658.4
ortho. ART	29	0	658.4
Border Signature	26	1	5.4

positive examples (cf. step 3, page 4). Therefore, classifiers are trained with positive examples that have not been used for creating the eigenspace, similar to the online classification phase. However, it turned out that the mean AUC value is not higher than the one of the solid line.

Although the classifiers have not been specially designed to deal with occlusions, the experiment showed that the Eigen-CSS method performs best with partly occluded objects. Fig. 9 presents the performance of all tested features. The lines mark the maximal cutting level above or below which the rest of the object can be classified.

6. Summary and Conclusions

This paper has presented a novel object recognition approach in range images from a 3D laser scanner. The algorithm utilizes a contour-based technique applied to depth information, resulting in a new, reliable and fast detection approach. Various real world experiments showed that the system is capable of stable object detection, applicable for environment cognition of autonomous mobile robots. Our approach benefits from a reliable 3D laser scanner that is the emerging sensing technology in robotics.

The key innovation presented in this paper is the combination of reliable contour extraction with a new Eigen-CSS method for feature extraction. The computed features are of high quality, resulting in an object detection that achieves a classification performance comparable to the MPEG-7 standard method ART. However, our method is nearly one order of magnitude faster. The only method being faster than Eigen-CSS, i.e., Border Signature, shows difficulties with occlusions (cf. Fig. 9).

Future work in our robotics context will concentrate on three aspects:

1. The detected objects will be used as an index to a database of 3D models. The model and the position of the detected object can be used as a start position for an ICP based matching in the range data.

¹The following geometric features have been used: area-perimeter ratio, aspect ratio, rectangularity, eccentricity, orientation, and radii ratio.

²In practice, the actual time of the combined classifiers lie below the sum of the single times due to hashing.

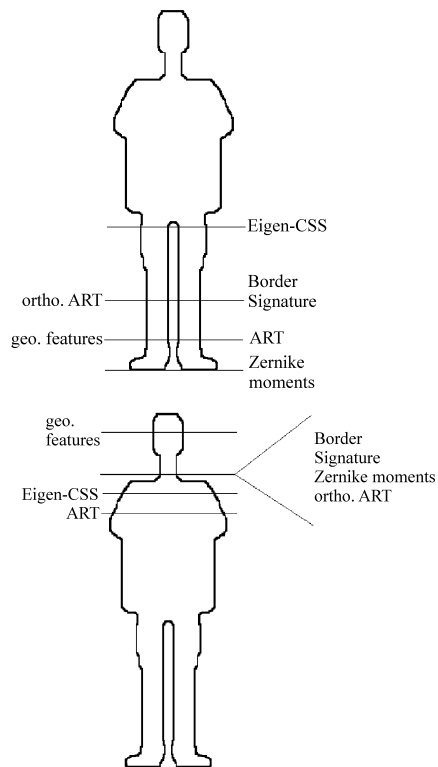


Figure 9. Comparison of the classifiers' robustness against occlusions. Top: maximal occlusion from below; down: maximal occlusion from above.

2. Integration of camera information for detecting objects that are difficult to process by a laser scanner, e.g., due to their small size.
3. Using bimodal laser data, i.e., combining range and reflectance data for an even more robust object segmentation.

The overall goal is to use an autonomous mobile robot to build 3D semantic maps that contain temporal and spatial 3D information with descriptions and labels about the environment.

Acknowledgments

We thank Kai Pervözl and Hartmut Surmann for preceding joint research.

References

- [1] A. P. Ashrock, R. B. Fisher, C. Robertson, and N. Werghi. Finding surface correspondences for object recognition and registration using pairwise histograms. In *Proceedings of*

- the European Conference on Computer Vision (ECCV '98)*, pages 185 – 201, Freiburg, Germany, June 1998.
- [2] P. Besl and N. McKay. A method for Registration of 3–D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239 – 256, February 1992.
- [3] M. Bober. MPEG-7 Visual Shape Descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):716 – 719, 2001.
- [4] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on COLT*, pages 144 – 152, Pittsburgh, PA, U.S.A., 1992.
- [5] R. J. Campbell and P. J. Flynn. Eigenshapes for 3D Object Recognition in Range Data. In *Proceedings of the IEEE Conference Computer Vision and Pattern Recognition (CVPR '99)*, Fort Collins, CO, U.S.A., 1999.
- [6] R. J. Campbell and P. J. Flynn. A Survey Of Free-Form Object Representation and Recognition Techniques. *Computer Vision and Image Understanding (CVIU)*, 81, 2001.
- [7] R. R. C.T. Zahn. Fourier descriptors for plane closed curves. *IEEE Trans. Comput.*, 21:269 – 281, 1972.
- [8] M. Drew, T. Lee, and A. Rova. Shape Retrieval with Eigen-CSS Search. Technical report, School of Computing Science, Simon Fraser University, Canada V5A 1S6, 2 2005.
- [9] T. Fawcett. Roc Graphs: Notes and Practical Considerations for Researchers. Technical report, HP Laboratories, MS 1143, 1501 Page Mill Road, CA 94304, 3 2004.
- [10] M. Hu. Visual Pattern Recognition by Moment Invariants. *IRE Trans. on Information Theory*, IT-8:179 – 187, 1962.
- [11] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433 – 449, May 1999.
- [12] N. Milstein. Image Segmentation by adaptive Thresholding. Technical report, Technion Israel Institute of Technology, The Faculty for Computer Sciences, 1998.
- [13] F. Mokhtarian. Silhouette-Based Isolated Object Recognition through Curvature Scale Space. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(5):539 – 544, 1995.
- [14] F. Stein and G. Medioni. Structural indexing: Efficient 3d object recognition. *IEEE Transaction on Pattern Analysis and machine Vision (PAMI)*, 14:125 – 145, February 1992.
- [15] Y. Sun, J. Paik, A. Koschan, D. Page, and M. Abidi. Point Fingerprint: An New 3D Object Representation Scheme. *IEEE trans. on Systems, Man, and Cybernetics – Part B: Cybernetics*, 33(4), 2003.
- [16] H. Surmann, K. Lingemann, A. Nüchter, and J. Hertzberg. A 3D laser range finder for autonomous mobile robots. In *Proc. of the of the 32nd International Symposium on Robotics (ISR '01)*, pages 153 – 158, Seoul, Korea, April 2001.
- [17] M. Teague. Image Analysis via the General Theory of Moments. *J. Optical Soc. Am.*, 70(8):920 – 930, 1980.
- [18] O. Wulf, K. O. Arras, H. T. Christensen, and B. Wagner. 2D Mapping of Cluttered Indoor Environments by Means of 3D Perception. In *ICRA*, New Orleans, USA, April 2004.
- [19] D. Zhang and M. Hebert. Harmonic maps and their application in surface matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '99)*, pages 2524 – 2530, Ft. Collins, CO, U.S.A., June 1999.