Automatic Classification of Objects in 3D Laser Range Scans

Andreas Nüchter, Hartmut Surmann, Joachim Hertzberg
Fraunhofer Institute for Autonomous Intelligent Systems (AIS)
Schloss Birlinghoven
D-53754 Sankt Augustin, Germany
{nuechter,surmann,hertzberg}@ais.fraunhofer.de

Abstract. This paper presents a new method for object detection and classification in 3D laser range data that is acquired by an autonomous mobile robot. Off-screen rendered depth and reflectance images serve as an input for an Ada Boost learning procedure that constructs a cascade of classifiers. The performance of the classification is improved by combining both sensor modalities, which are independent from external light. The resulting approach for object classification is real-time capable and reliable. It combines recent results in computer vision with the emerging technology of 3D laser scanners.

1 Introduction

A fundamental problem in the design of autonomous mobile cognitive systems is the perception of the environment. A basic part of the perception is to learn, detect and recognize objects, which has to be done with the limited resources of a mobile robot. The performance of a mobile robot crucially depends on the accuracy, duration and reliability of its perceptions and the involved interpretation process. This paper proposes a new method for the learning, fast detection and classification of instances of 3D object classes. The approach uses 3D laser range and reflectance data on an autonomous mobile robot to perceive the 3D objects. The 3D range and reflectance data are transformed into images by off-screen rendering. A cascade of classifiers, i.e., a linear decision tree, is used to detect the objects. Based on the ideas of Viola and Jones, each classifier is composed of several simple classifiers, that in turn contain an edge, line or center surround feature [19]. They and others have presented and implemented a method for the effective computation of these features using an intermediate representation, namely, integral image [9, 10, 19]. For learning of the object classes, a boosting technique, namely, Ada Boost, is used [5]. The resulting approach for object classification is reliable and real-time capable and combines recent results in computer vision with the emerging technology of 3D laser scanners.

Other approaches use information of CCD-cameras that provide a view of the robot's environment. Nevertheless, cameras are difficult to use in natural environments with changing light conditions. Robot control architectures that include robot vision rely mainly on tracking of features, e.g., invariant features [15], light sources [8] or the ceilings [3]. Other camera-based approaches to robot vision, e.g., stereo cameras and structure from motion, have difficulties to provide navigation information for a mobile robot in real-time. So, many current successful robots are equipped with distance sensors, mainly 2D laser range finders [18]. 2D

scanners have difficulties to detect 3D obstacles with jutting out edges. Currently a general tendency exists to use 3D laser range finders and build 3D maps [1, 11, 16, 17].

Some groups have attempted to build 3D volumetric representations of environments with 2D laser range finders. For example, Thrun et al. [18], use two 2D laser range finders for acquiring 3D data. One laser scanner is mounted horizontally, the other vertically. The latter one grabs a vertical scan line that is transformed into 3D points using the current robot pose. A few other groups use 3D laser scanners [1, 16]. A 3D laser scanner generates consistent 3D data points within a single 3D scan. The RESOLV project aimed at modeling interiors for virtual reality and telepresence [16]. They used a RIEGL laser range finder on two mobile robots called EST and AEST (Autonomous Environmental Sensor for Telepresence). The AVENUE project develops a robot for modeling urban environments using a CYRAX laser scanner [1].

In the area of object recognition and classification in 3D range data, Johnson and Hebert use the well-known ICP algorithm [2] for registering 3D shapes in a common coordinate system [7]. The necessary starting guess of the ICP algorithm is done by localizing the object with spin images [7]. This approach was extended by Shapiro et al. [14]. In contrast to our proposed method, both approaches use local, memory consuming surface signatures based on prior created mesh representations of the objects.

The paper is organized as follows: The next section describes the autonomous mobile robot that is equipped with the AIS 3D laser range finder. Then we present the object learning and detection algorithm. Section 4 states the results and section 5 concludes.

2 The Autonomous Mobile Robot

2.1 The Kurt3D Robot Platform

Kurt3D (Figure 1, top left) is a mobile robot platform with a size of 45 cm (length) $\times$ 33 cm (width) $\times$ 26 cm (height) and a weight of 15.6 kg. Equipped with the 3D laser range finder the height increases to 47 cm and the weight to 22.6 kg.[1] Kurt3D's maximum velocity is 5.2 m/s (autonomously controlled 4.0 m/s). Two 90W motors are used to power the 6 wheels, where the front and rear wheels have no tread pattern to enhance rotating. Kurt3D operates for about 4 hours with one battery (28 NiMH cells, capacity: 4500 mAh) charge. The core of the robot is a Pentium-III-600 MHz with 384 MB RAM. An embedded 16-Bit CMOS microcontroller is used to control the motor.

2.2 The AIS 3D Laser Range Finder

The AIS 3D laser range finder (Figure 1, top middle) [11, 17] is built on the basis of a 2D range finder by extension with a mount and a standard servo motor. The 2D laser range finder is attached in the center of rotation to the mount for achieving a controlled pitch motion. The servo is connected on the left side (Figure 1, top middle). The 3D laser scanner operates up to 5h (Scanner: 17 W, 20 NiMH cells with a capacity of 4500 mAh, Servo: 0.85 W, 4.5 V with batteries of 4500 mAh) on one battery pack.

The area of $180°$(h) $\times$ $120°$(v) is scanned with different horizontal (181, 361, 721 pts.) and vertical (210, 420 pts.) resolutions. A plane with 181 data points is scanned in 13 ms by

[1] Videos of the exploration with the autonomous mobile robot can be found at `http://www.ais.fhg.de/ARC/kurt3D/index.html` and `http://www.ais.fhg.de/ARC/3D/scanner/cdvideos.html`

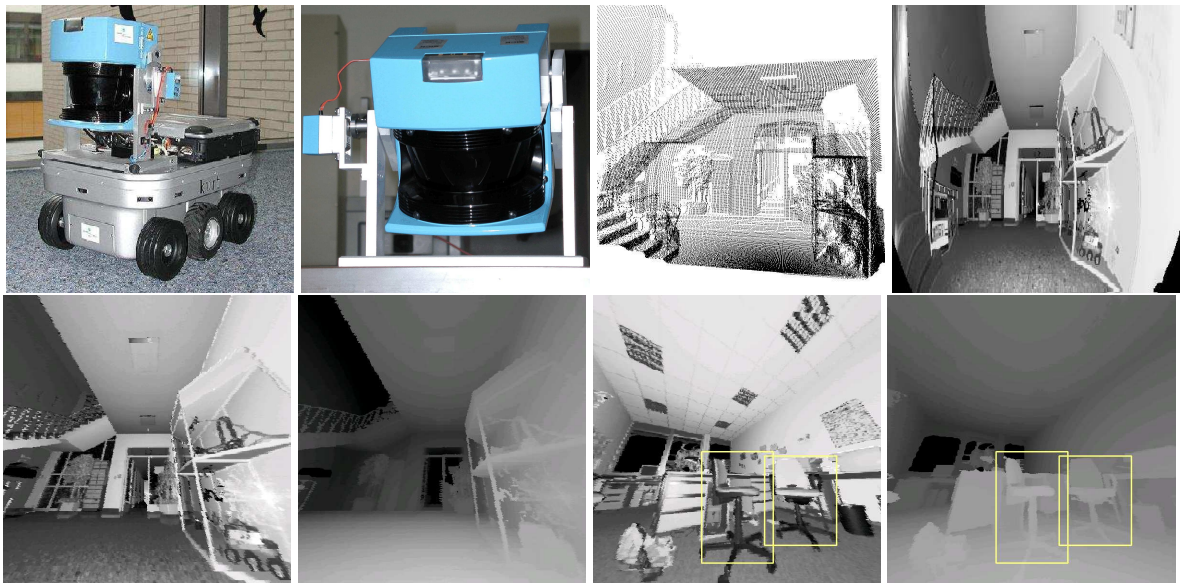

Figure 1: Top row: The autonomous mobile robot Kurt3D equipped with the 3D scanner. The AIS 3D laser range finder. Its technical basis is a SICK 2D laser range finder (LMS-200). Top Right: Scanned scene as point cloud (viewing pose 1 meter behind scan pose). Reflectance values (distorted view: One scan line of the figure corresponds to a slice of the 2D scanner [17]). Bottom row: Input images used for object learning (negative and positive examples for learing the object "office chair"). Undistorted rendered view as reflectance and depth image (range values are encoded by grey values).

the 2D laser range finder (rotating mirror device). Planes with more data points, e.g., 361, 721, duplicate or quadruplicate this time. Thus, a scan with 181 $\times$ 210 data points needs 2.8 seconds. In addition to the distance measurement, the 3D laser range finder is capable of quantifying the amount of light returning to the scanner. Figure 1 (top right) shows a scanned scene as point cloud with a viewing pose one meter behind the scan pose, and the reflectance image (bottom left) of this scene. After scanning the 3D data points are projected by an off-screen `OpenGL`-based rendering module onto an image plane to create a 2D images. The camera for this projection is located in the laser source, thus all points are uniformly distributed and enlarged to remove gaps between them on the image plane. Figure 1 (bottom row) shows a reflectance images and rendered depth images, with distances encoded with grey values.

3 Object Classification

Object detection and classification has intensely been researched in computer vision [4, 12, 13, 19]. Common approaches use for example neural networks or support vector machines (SVM) to detect and classify objects. Rowley et al. detect faces using a small set of simple features and neural networks [13] and Papageorgiou et al. recognize pedestrians with simple vertical, horizontal and diagonal features and SVMs [12]. Recently Viola and Jones have proposed a boosted cascade of simple classifiers for fast face detection [19]. Inspired by these ideas, we detect objects, e.g., office chairs [20], in 3D range and reflectance data using a cascade of classifiers composed of several simple classifiers, which in turn contain an edge, line or center surround feature.

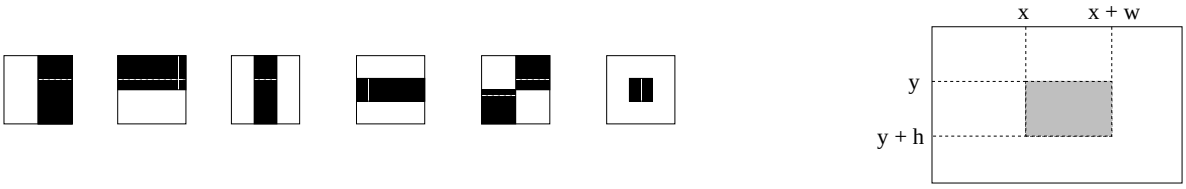

Figure 2: Left: Edge, line, diagonal and center surround features are used for classification. Right: Computation of feature values in the shaded region is based on the four upper rectangles.

3.1 Feature Detection using Integral Images

There are many motivations for using features rather than pixels directly. For mobile robots, a critical motivation is that feature based systems operate much faster than pixel based systems [19]. The features used here have the same structure as the Haar basis functions, i.e., step functions introduced by Alfred Haar to define wavelets [6]. They are also used in [9, 10, 12, 19]. Figure 2 (left) shows the six basis features, i.e., edge, line, and center surround features. The base resolution of the object detector is 20×40 pixels, thus the set of possible features in this area is very large (361760 features). In contrast to the Haar basis function, the set of rectangle features is not minimal. A single feature is effectively computed on input images using integral images [19], also known as summed area tables [9, 10]. An integral image I is an intermediate representation for the image and contains the sum of gray scale pixel values of image N with height y and width x, i.e.,

$$I(x,y) = \sum_{x'=0}^{x} \sum_{y'=0}^{y} N(x',y').$$

The integral image is computed recursively, by the formulas: $I(x,y) = I(x,y-1) + I(x-1,y) + N(x,y) - I(x-1,y-1)$ with $I(-1,y) = I(x,-1) = 0$, therefore requiring only one scan over the input data. This intermediate representation $I(x,y)$ allows the computation of a rectangle feature value at (x,y) with height and width (h,w) using four references (see Figure 2 (right)):

$$F(x,y,h,w) \;\; = \;\; I(x,y) + I(x+w,y+h) - I(x,y+h) - I(x+w,y).$$

Since the features are a composition of rectangles, they are computed with several lookups and subtractions weighted with the area of the black and white rectangles. To detect a feature, a threshold is required. This threshold is automatically determined during a fitting process, such that a minimum number of examples are misclassified. The examples are given in a set of images that are classified as positive or negative samples. The set is also used in the learning phase that is briefly described next.

3.2 Learning Classification Functions

The Gentle Ada Boost Algorithm is a variant of the powerful boosting leaning technique [5]. It is used to select a set of simple features to achieve a given detection and error rate. In the following, a detection is referred as hit and an error as a false alarm. The various Ada Boost algorithms differ in the update scheme of the weights. According to Lienhart et al. the Gentle Ada Boost Algorithm is the most successful learning procedure tested for face detection applications [9].

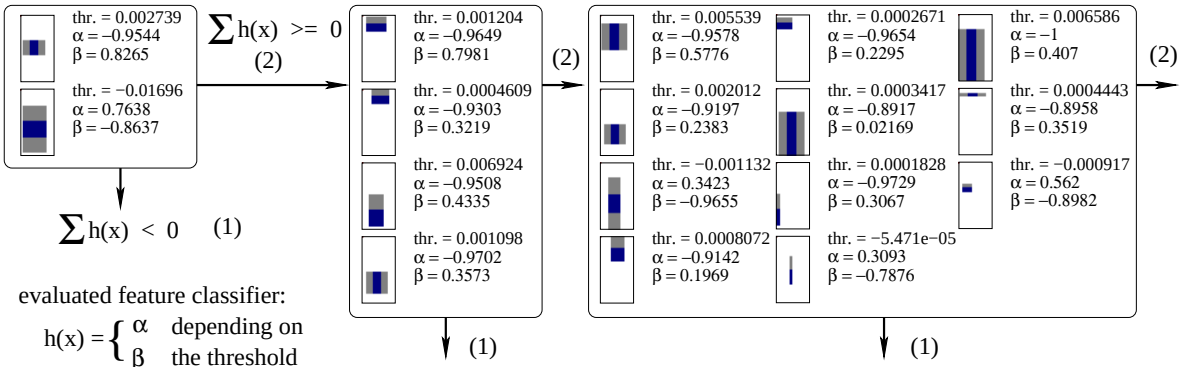

Figure 3: The first three stages of a cascade of classifiers to detect an office chair in depth data. Every stage contains several simple classifiers that use Haar-like features.

The learning is based on N weighted training examples $(x_1, y_1), \ldots, (x_N, y_N)$, where x_i are the images and $y_i \in \{-1, 1\}, i \in \{1, \ldots, N\}$ the classified output. At the beginning of the learning phase the weights w_i are initialized with $w_i = 1/N$. The following three steps are repeated to select simple features until a given detection rate d is reached:

1. Every simple classifier, i.e., a single feature, is fit to the data. Hereby the error e is calculated with respect to the weights w_i.
2. The best feature classifier h_t is chosen for the classification function. The counter t is increased.
3. The weights are updated with $w_i := w_i \cdot e^{-y_i h_t(x_i)}$ and renormalized.

The final output of the classifier is sign$(\sum_{t=1}^{T} h_t(x))$, with $h(x) = \alpha$, if $x \geq$ thr. and $h(x) = \beta$ otherwise. α and β are the output of the fitted simple feature classifiers, that depend on the assigned weights, the expected error and the classifier size. Next, a cascade based on these classifiers is built.

3.3 The Cascade of Classifiers

The performance of one classifier is not suitable for object classification, since it produces a high hit rate, e.g., 0.999, and error rate, e.g., 0.5. Nevertheless the hit rate is much higher than the error rate. To construct an overall good classifier, several classifiers are arranged in a cascade, i.e., a degenerated decision tree. In every stage of the cascade a decision is made whether the image contains the object or not. This computation reduces both rates. Since the hit rate is close to one, their multiplication results also in a value close to one, while the multiplication of the smaller error rates approaches zero. Furthermore the whole classification process speeds up. Figure 3 shows an example cascade of classifiers for detecting an “office chair” in depth images.

An overall effective cascade is learned by a simple iterative method. For every stage the classification function $h(x)$ is learned, until the required hit rate is reached. The process continues with the next stage using only the currently misclassified negative examples. The number of features used in each classifiers increases with additional stages (Figure 3).

4 Application of the Cascade and Results

Several experiments have been done to evaluate the performance of the proposed approach with two different kinds of images, namely, reflectance and depth images. Both types are ac-

quired by the AIS 3D laser range finder and are light invariant. Figure 1 shows two examples of the training data set. Around 200 representation of an "office chair" were taken in addition to a wide variety of negative examples without any chair, e.g., the scene given in Figure 1. The detection starts with a classifier of size 20×40 pixels. The image is searched from top left to bottom right by applications of the cascade. To detect objects on larger scales, the detector is rescaled. An advantage of the Haar-like features is that they are easily scalable. Each feature requires only a fixed number of look-ups in the integral image, independent of the scale. Time-consuming picture scales are not necessary to achieve scale invariance.

Table 1 summarizes the results of the object detection algorithm with a test data set of 30 scans that are not used for learning. Some examples of the detection of an "office chair" in 3D scans are given in Figure 4. Hits as well as missed and false alarms are documented. In addition, the figure presents the scaling feature of the detector, since the last two images of the third row were rendered with a wide apex angle of the virtual projection camera. In addition some results of the proposed object detection with partial occlusions are shown (bottom row). The cascade in Figure 3 presents the first three stage classifiers for the object "office chair" using depth values. One main feature is the horizontal bar (first stage).

The experiments inspired us to combine the cascades of the depth and reflectance images. Figure 5 shows two variants of the combination: Either the two cascades run interleaved (left) or serial (right) and represent a logical "and". The joint cascade decreases the false detection rate close to zero. To avoid the reduction of the hit rate, 6 different off-screen rendered images are used, where the virtual camera is rotated, i.e., the rotation by the Euler angles $(\theta_x, \theta_y) \in \{(0,0), (-20,-20), (-20,20), (20,-20), (20,20)\}$ is applied. The 6th image is generated with a wide apex angle of 150 deg.

Table 1: Number of stages versus hit rate and false alarms. The last row shows the result of the combined classifier for reflectance and depth images. A detection including searching in the image using the combined cascade with 15 + 15 stages needs 376ms (Pentium-IV-2400).

number of stages	hit rate		false alarms	
	reflect. img.	depth img.	reflect. img.	depth img.
15	0.9	0.866	0.067	0.067
30	0.867	0.767	0.067	0.033
(15 + 15) applied to 6 img.	0.967		0.0	

5 Conclusions

This paper has presented a new method for object classification in 3D scans. The scans are automatically acquired by an autonomous mobile robot equipped with the AIS 3D laser range finder. The scanner provides external, light independent, bimodal data that is converted to depth and reflectance images. It is shown that both image types, and especially their combination, are a good choice for object detection and classification. The object is classified with complex classifiers that are arranged in a cascade. The classifiers use Haar-like features and an internal object representation is learned with the Ada Boost techniques. Typical vision problems, e.g, shadows or posters on the wall showing distracting objects [20], are avoided by the use of range images.

Needless to say, some work remains to be done. The detected object 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. Furthermore additional rotated features of 45° as proposed by Lienhart and Maydt will be used to improve

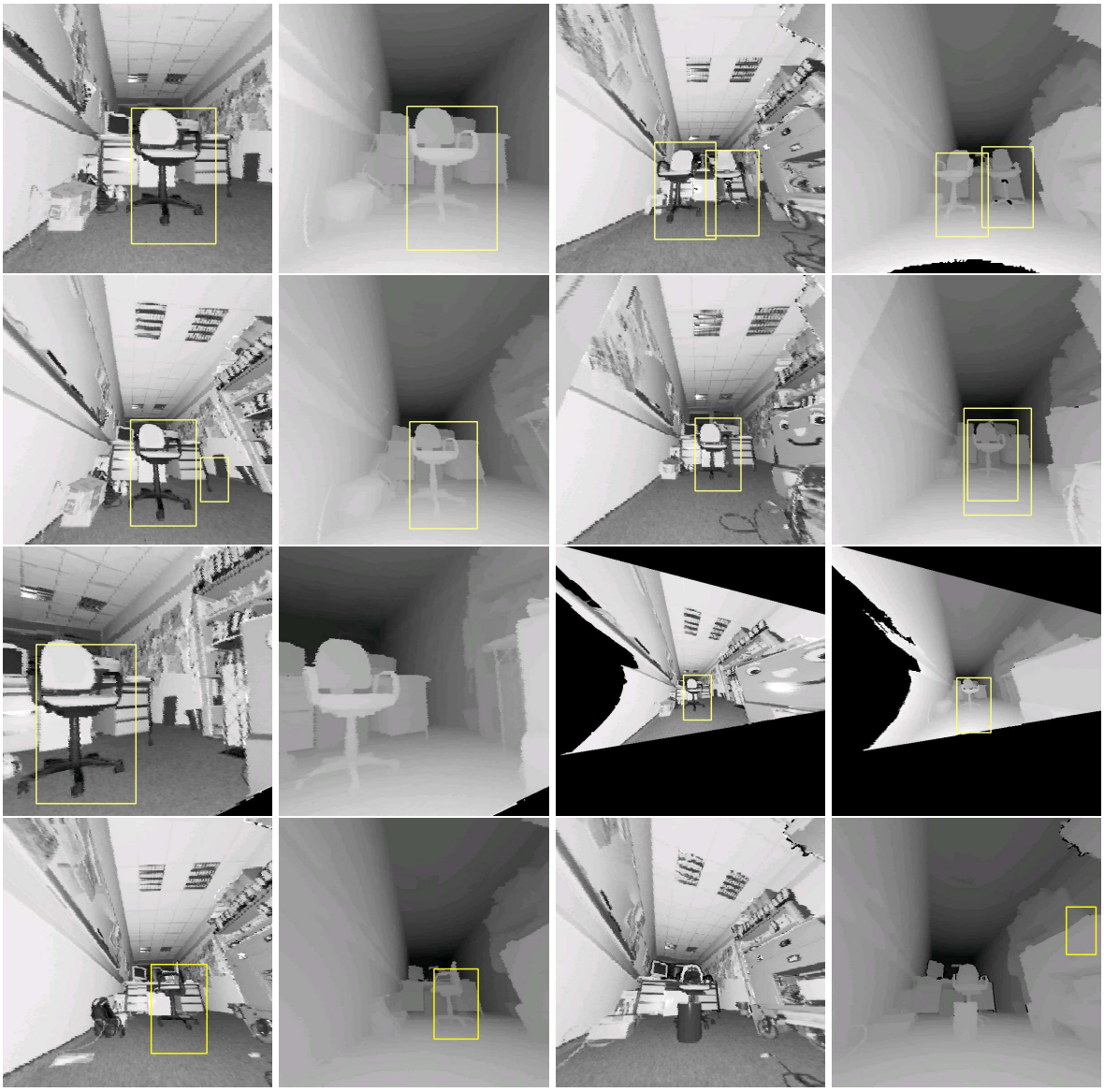

Figure 4: Detection results using the classifier with 15 stages. The classified object is marked by a rectangle. Top row: Detection in reflectance and depth images. Second row: A false classification in a reflectance image is not present in the depth image (left). An object might be detected with different detector scales (right). Third row: Rotated images (left) and wide angle projections (right). Bottom row: Detection results under presence of partial occlusions. Small changes of the viewpoint are tolerated, e.g., a view from the side (left). If the main features are occluded the object detection fails (right).

the classification [10]. 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.

Acknowledgment: Special thanks to Simone Frintrop, Kai Lingemann, and Kai Pervölz for supporting our work. Many thanks to the unkown reviewers for hints and corrections.

References

[1] P. Allen, I. Stamos, A. Gueorguiev, E. Gold, and P. Blaer. AVENUE: Automated Site Modeling in Urban Environments. In *Proc. of the 3rd Int. Conf. on 3D Digital Imaging and Modeling (3DIM '01)*, Canada, 2001.

Figure 5: Proposed combined cascades. Left: Interleaved cascade. Right: Serial cascade.

[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, 1992.

[3] F. Dellaert, W. Burgard, D. Fox, and S. Thrun. Using the Condensation Algorithm for Robust, Vision-based Mobile Robot Localization. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '99)*, USA, 1999.

[4] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2002.

[5] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Machine Learning: Proc. of the 13th Int. Conf.*, pages 148 – 156, 1996.

[6] A. Haar. Zur Theorie der orthogonalen Funktionensysteme. *Math. Ann.*, (69):331 – 371, 1910

[7] A. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on PAMI*, 21(5):433 – 449, 1999.

[8] F. Launay, A. Ohya, and S. Yuta. Autonomous Indoor Mobile Robot Navigation by detecting Fluorescent Tubes. In *Proc. of the 10th Int. Conf. on Advanced Robotics (ICAR '01)*, Hungary, 2001.

[9] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. In *Proc. of the German 25th Pattern Recognition Symposium (DAGM '03)*, Germany, 2003.

[10] R. Lienhart and J. Maydt. An Extended Set of Haar-like Features for Rapid Object Detection. In *Proc. of the IEEE Conf. on Image Processing (ICIP '02)*, pages 155 – 162, USA, 2002.

[11] A. Nüchter, H. Surmann, and J. Hertzberg. Planning Robot Motion for 3D Digitalization of Indoor Environments. In *Proc. of the 11th Int. Conf. on Advanced Robotics (ICAR '03)*, pages 222 – 227, Portugal, 2003.

[12] C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *Proc. of the 6th Int. Conf. on Computer Vision (ICCV '98)*, India, 1998.

[13] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on PAMI*, 20(1):23 – 38, 1998.

[14] S. Ruiz-Correa, L. G. Shapiro, and M. Meila. A New Paradigm for Recognizing 3-D Object Shapes from Range Data. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '03)*, USA, 2003.

[15] S. Se, D. Lowe, and J. Little. Local and Global Localization for Mobile Robots using Visual Landmarks. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS '01)*, USA, 2001.

[16] V. Sequeira, K. Ng, E. Wolfart, J. Goncalves, and D. Hogg. Automated 3D reconstruction of interiors with multiple scan–views. In *Proc. of SPIE, Electronic Imaging '99, The Society for Imaging Science and Technology /SPIE's 11th Annual Symp.*, USA, 1999.

[17] 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 Int. Symp. on Robotics (ISR '01)*, pages 153 – 158, Korea, 2001.

[18] S. Thrun, D. Fox, and W. Burgard. A real-time algorithm for mobile robot mapping with application to multi robot and 3D mapping. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA '00)*, USA, 2000.

[19] P. Viola and M. Jones. Robust Real-time Object Detection. In *Proc. of the 2nd Int. Workshop on Statistical and Computational Theories of Vision – Modeling, Learning, Computing and Sampling*, Canada, 2001.

[20] Max Planck Institute for biological Cybernetics. `http://www.kyb.tuebingen.mpg.de/de/bu/projects.html?ra=10`