Colored 2D maps for robot navigation with 3D sensor data

Conference Paper · January 2004

DOI: 10.1109/IROS.2004.1389864 · Source: IEEE Xplore

Colored 2D Maps for Robot Navigation with 3D Sensor Data

Oliver Wulf, Christian Brenneke, Bernardo Wagner
Institute for Systems Engineering
University of Hannover
Hannover, Germany
{wulf, brenneke, wagner}@rts.uni-hannover.de

Abstract — **This paper presents a navigation system for mobile service robots working in urban environments. The system combines a 3D laser sensor with 2D algorithms for path planning and simultaneous localization and mapping (SLAM). In contrast to other map representations the Colored 2D Map, first presented in this paper, is able to hold information adapted to both localization and path planning. The functionality of our approach is demonstrated by an experiment for online navigation with 3D data.**

Keywords: 3D perception, navigation, mapping

I. INTRODUCTION

Navigation capabilities including localization, mapping and path planning are key features of mobile service robots. Up to now many contributions have been made to solve these tasks. However, the complexity of real world environments is still a challenging problem. Uneven outdoor environments, dynamic objects, cluttered scenes and the absence of simple features are known problems. Most navigation systems presented up to now are using 2D sensors like laser or sonar in combination with a 2D map for localization and path planning.

In recent years more and more researchers are trying to solve the navigation task by use of 3D sensors. The opportunity to solve navigation problems with the rich information content of 3D data sounds promising. But on the other hand the huge amount of data that needs to be processed leads to new problems. These problems are especially present if the data is not only used to build static models, but also to close the control loop and navigate online.

Current implementations using 3D data for robot navigation can be divided into three groups:

One group of systems used for outdoor navigation build full 3D models of the environment [1][2][3]. These complex and computationally expensive methods are able to model completely unstructured environments as they can be found in rough terrain or on planetary surfaces. Common map representations in this kind of systems are digital elevation maps (DEM), 3D point clouds and 3D evidence grids.

Another approach is based on extraction of features like walls and tree trunks from the 3D data [4][5]. These features could thereafter be used for navigation with feature maps. The advantage of this approach is the high data abstraction and therefore compact map representation. But on the other hand feature extraction from 3D data is computational very expensive. For this reason no SLAM algorithm that uses 3D features in real-time is published yet.

The third approach combines 3D perception and a 2D map representation by use of virtual 2D scans. These virtual 2D scans, which are generated from 3D raw data, can be used as input data for already existing 2D navigation algorithms. This approach has been published for outdoor environments [6] and in combination with a line-based SLAM algorithm for indoor environments [7].

Figure 1. Mobile robot within urban environment

This paper describes an approach belonging to the third group, which continues the work presented in [6]. As the algorithms presented in [6] and [7] are limited to localization and mapping, the fundamental extension, which is contributed with this paper, is the ability to do 2D path planning with 3D constraints. In general different navigation tasks like localization and path planning need different information of the 3D environment stored in the map. The localization system for example needs information about natural landmarks and the path planning algorithm needs information about all non traversable obstacles. The bicycles in Fig. 1 for example are obstacles that need to be taken into account for collision free path planning. On the other hand the wall behind these bikes is preferably used as a natural landmark. The principal idea of our approach is to extract two virtual 2D scans from the 3D point cloud. One scan contains landmarks (used for SLAM) and the other scan contains obstacles (used for path planning). As regular 2D maps are not able to differ between obstacles and landmarks we introduce the *Colored 2D Map* that is able to carry both types of information in a consistent way.

The algorithm able to extract object points with attributes, in form of virtual landmark scans and virtual obstacle scans, from 3D raw data is shown in section II. Section III gives a description of the map representation and the applied SLAM techniques. Finally, section IV shows the results of the first online experiments including path planning and simultaneous localization and mapping in urban environments.

II. 3D PERCEPTION

This section describes the 3D perception system that generates virtual 2D data for the subsequent navigation algorithms. Subsection II.A gives a brief overview about the 3D raw data acquisition. In subsections II.B and II.C the algorithms for data preprocessing including segmentation and object-ground classification are described. Subsections II.D and II.E describe the methods used to generate *virtual landmark scans* and *virtual obstacle scans* from preprocessed 3D point-clouds.

A. 3D laser range scans

The 3D sensor we use for this work is a laser range scanner that works on the time-of-flight measurement principle. The sensor consists of a vertically aligned 2D scanner that is turned by a ScanDrive servo drive especially designed for fast 360° scans (yawing scan). For more details on the 3D sensor and methods for fast scanning see [8].

The acquired 3D raw data can be processed in different data representations and coordinate frames. One possibility is the range image representation that is commonly used in 3D computer vision. A range image can be defined as a 2D array of points where D_{ij} is the distance of the point P_{ij} to the sensor, with i being the index of the vertical raw scan (column) and j the index within the raw scan (row) counting bottom up. Thus a range image is similar to a grayscale image with the gray values representing the distance to the sensor.

Another data representation is the 3D point-cloud. In this format the 3D data is represented as a list of points that are given in Polar coordinates $P_{ij} = (r_{ij}, \varphi_{ij}, z_{ij})^T$ or in Cartesian coordinates $P_{ij} = (x_{ij}, y_{ij}, z_{ij})^T$. In contrast to range images 3D point-clouds are undistorted. In our application it is even possible to measure undistorted 3D point-clouds with a moving robot. Both representations, range image and 3D point-cloud, are used in our approach. The range image for segmentation and the 3D point-cloud for all subsequent algorithms.

B. Range image segmentation

The first processing step is the segmentation of the 3D data. The goal of this algorithm is to group points that belong to the same surface. The same time the algorithm classifies points that belong to objects with a rough surface and thus cannot be grouped. Taking a low-resolution 3D sensor, objects with rough surfaces are e.g. bushes, treetops and small objects like bikes.

In this work we use an edge-based segmentation algorithm to segment the 3D data given as a range image. Within our approach we define a vertical edge as a point D_{ij} where the gradient of the distance is not continuous. Inversely the discrete gradient is defined to be continuous if

$$\begin{gathered}\frac{1}{D_{rel,\max}} < \frac{D_{ij+1} - D_{ij}}{D_{ij} - D_{ij-1}} < D_{rel,\max} \text{ or if} \\ D_{ij} - D_{ij-1} < D_{\min} \text{ and} \\ D_{ij+1} - D_{ij} < D_{\min}\,, \end{gathered} \qquad (1)$$

with $D_{rel,\max}$ being a relative threshold and $D_{\min}$ being a threshold that eliminates the sensor noise. This 2D edge detection is calculated twice, first on each vertical raw scan and afterwards line-by-line with a similar definition for horizontal edges. Fig. 2 shows a range image of an example scene. Within this figure vertical edge points are colored black, horizontal edge points are colored red and points that are both horizontal and vertical edges in the same time are colored green. All other points are non-edge points (gray) or infinite readings (white).

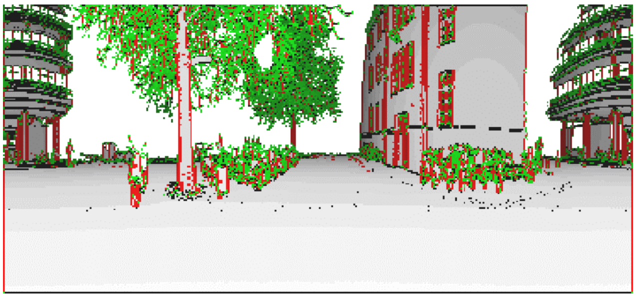

Figure 2. Range image with edge detection

Subsequent to edge detection, surface segments can be labeled by use of a flood fill algorithm that groups neighboring non-edge points. A final dilatation step removes edge points on the boarder of a segment and most of the noise. Fig. 3 shows the output of this segmentation algorithm applied to the same example scene. All data points that remain gray cannot be connected to a smooth surface and are therefore classified as rough surface points.

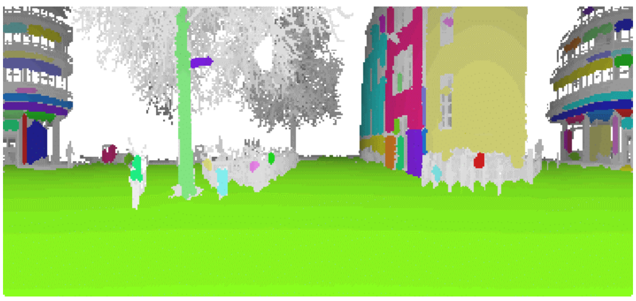

Figure 3. Segmented range image

C. *Object-ground classification*

The second preprocessing step is the classification of object- and ground-points within the 3D point-cloud. As a similar algorithm has already been presented in [9], it will only be discussed briefly.

The algorithm is based on the definition that a point P_{ij} is an object-point if there is at least one point P_{ik} within the same vertical scan that is "below" P_{ij}. With "below" defined as:

$$
\begin{gathered}
P_{ij,z} - P_{ik,z} > H_{\min}, \\
\left| P_{ij,r} - P_{ik,r} \right| < R_{\max}, \\
\left| \frac{P_{ij,r} - P_{ik,r}}{P_{ij,z} - P_{ik,z}} \right| < \tan(\alpha), \\
0 \leq k < j,
\end{gathered}
\tag{2}
$$

where $H_{\min}$ is the minimum height of an obstacle, $R_{\max}$.is the maximum search radius and α is the maximum angle misalignment (see Fig. 4).

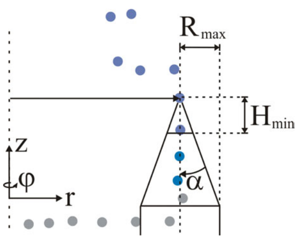

Figure 4. Definition of object points

D. *Extraction of landmark scans*

The aim of virtual 2D scans is to reduce the amount of data that is contained in 3D scan without loosing the information essential for navigation in case of the virtual landmark scan and for obstacle avoidance in case of the virtual obstacle scan. The problem is to find a suitable heuristic that is able to do the necessary projection. A first heuristic for virtual landmark scans was already presented in [5]. The work presented in this paper contributes fundamental extensions of this method. Thus it is able to deal with a broader variety of environment situations.

This heuristic that selects one 3D point per vertical raw scan divides into 4 steps:

The first criterion says that a landmark point needs to be a point that is classified as both object-point and flat-surface point. By this means all points that are not suitable for 2D matching are filtered out (e. g. vegetation- or ground-points, see Fig. 5).

Corresponding to [6] the foot point of each flat object segment is projected to the virtual landmark scan. These foot points are preferably used for 2D matching as they can be clearly identified in successive 3D scans.

In case of two or more objects that can be seen in one vertical raw scan the most distant object will be chosen to be represented in the scan as it is most likely the best landmark. For example, in case of a small object standing in front of a wall, the wall is more likely to be seen in the following 3D scan than the small object.

Finally the virtual 2D scan is filtered to remove separated and very dense points. This filter does not affect the general shape of the scan but it enhances the performance of the following scan matching.

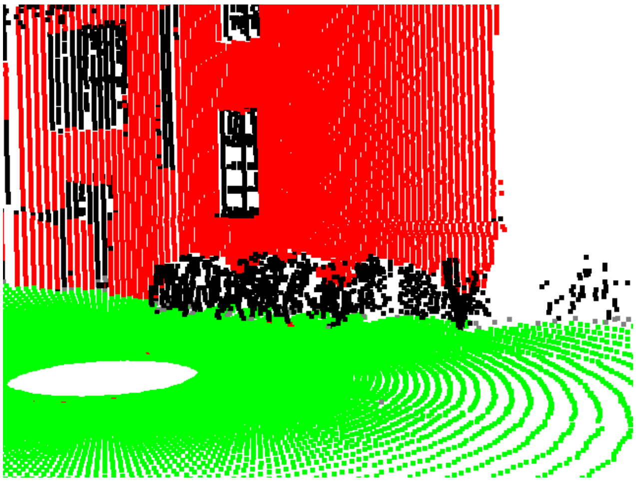

Figure 5. 3D point-cloud with flat object points (red), raw object points (black) and ground points (green)

E. *Extraction of obstacle scans*

The virtual obstacle scan is supposed to contain all information that is necessary for obstacle avoidance. The straightforward method to generate this obstacle scan is to project the obstacle points with the shortest distance to the robot into the virtual scan. As for all virtual scans only one point per vertical raw scan is included in the 2D scan. In this context an obstacle point is an object point with a height over ground that is smaller than the robot height. In general the calculation of the height over ground is a computational expensive task, as for every object point a pair of ground points needs to be found that lies directly under it. But we found out that a linear approximation of the ground level between the robot foot point and the most distant ground point gives good results within urban environments.

As it can be seen in Fig. 6 the bikes in front of the building are contained as obstacles (top of the scan) whereas the treetops are not contained within the virtual obstacle scan (center of the scan).

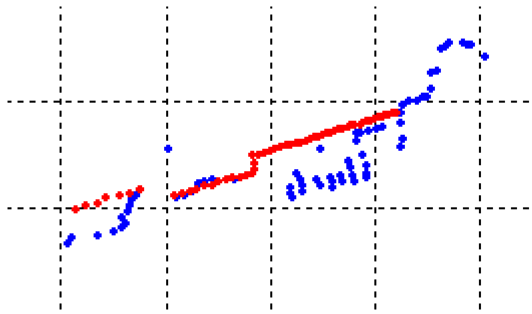

Figure 6. Virtual landmark scan (red), virtual obstacle scan (blue)

III. ROBOT NAVIGATION

This section describes an approach to use the generated virtual scans described above to give a holistic solution for the localization and mapping problem (subsection III.A) as well as path planning (subsection III.B) in complex urban environments.

The basic idea is to generate a universal *Colored 2D map* of the robots environment, which includes measuring points with different attributes (colors). These attributes were determined by 3D sensor data measurements. The implementation described in this paper is currently using two different attributes: landmark points and obstacle points.

A. Localization and mapping with landmark points

The scan based SLAM (simultaneous localization and mapping) algorithm is used to generate the *Colored 2D map*. A summary of this algorithm was first presented in [10] for autonomous robot navigation in poorly structured outdoor environments based on 2D laser range scans. The complete approach is described detailed in [11].

However, instead of using scans of a 2D laser range scanner we now use the virtual 2D landmark scan generated from the 3D sensor data. This allows a robust landmark detection as well as the possibility to extract only good landmark points for a successful scan matching.

For environment representation the scan based map is used. Thus, instead of extracting individual geometric features from a laser range scan, it is interpreted as a partial local image from the whole environment. By overlapping all single scans, a geometrically correct representation of natural landmarks in different structured environments is possible. In this case no restrictions concerning the geometrical contour of the landmarks are needed. All objects are faithfully reproduced, and the accuracy of the map is primarily restricted to the maximum resolution of the laser scanner.

To build a scan based map, full sets of sensor data are collected at different environment positions. Every set integrates measurements from the 3D laser sensor to generate a virtual landmark and obstacle scan as well as data from odometry and if available also from GPS. These sets of sensor data are representing network nodes. The distances between two individual nodes are denoted as network edges. The value of the distances are observed from the different sensors.

An extended IDC algorithm similar to [12] is used to observe the traveled distance between two data collecting positions by matching overlapped virtual landmark scans. The IDC algorithm is based on a point-to-point correspondence between scans that are compared. Two heuristics build pairs of points that are used to compute the rotation and shift between the two scans.

With odometry only relative distances between robot positions can be observed. However, the calculation of network edges by odometry is depending on the orientation of the network node. Thus, to get only linear dependencies, odometry can only be used for successive robot positions.

By solving the set of linear equations which describes the constraints of the sensor observation network an optimal solution for the position of the network nodes exists [10]. So the points of sensor data fusion in the robot environment are known. A map representation arises by overlaying all used 2D laser range scans into an absolute reference frame.

For map optimization the virtual 2D landmark scans generated from the 3D sensor data were used. Due to the fact that the point of view for the 3D scan and the virtual 2D landmark scan are identical, also the position of the 3D scan is known. This allows us to integrate further virtual 2D scans, e.g. obstacle scans, into the environment map. To identify the characteristics of the integrated measuring points different attributes are implemented. They are represented as different colors in the generated environment map.

In addition to general SLAM problems, e.g. convergence, consistency, accuracy and boundedness of the map error, the algorithm gives a high precision environment map independent of the type and the structure of available natural landmarks. That allows use of the algorithm in different structured environments without the explicit identification of single objects.

B. Path planning with obstacle points

The self generated *Colored 2D map* is now used for collision free path planning considering the 3D shape of the robot. However, here only the obstacle points of the *Colored 2D map* are used. These obstacle points enclose all measurements of objects in the environment, which would lead into a collision with the robot.

To use this information for collision free path planning a map representation for efficient path finding algorithms is needed. For this purpose a 2D grid map is generated. With the obstacle points of the *Colored 2D map* the state of the grids is being identified. We differ between obstacle grids, drivable grids and unknown grids. Out of the amount of drivable grids those are been extracted by a standard A*-algorithm [12] which are used for a collision free path to a pre-set goal position. Afterwards, the centers of these grids are used as base points for spline generation.

IV. EXPERIMENTAL RESULTS

The experiment presented here was made in a real world environment on the campus of the University of Hannover. The test scene includes objects which are typical for a urban environment, e.g.buildings, parking places, vegetation, bikes and so on (compare Fig. 7).

In this environment the robot was driven manually by a remote joystick. The driven speed of the robot was about walking speed. It was equipped with a gyroscope to observe the 3D orientation and wheel encoders to measure traveled distances.

For 3D perception a laser range scanner was used [8]. The horizontal opening angle was 360° with an angle resolution of 1.0° and the vertical opening angle was 90° with a vertical angle resolution of 0.5°. Taking a SICK

Figure 7. Campus scene which was used for navigation experiments with Colored 2D maps

LMS 291-S15 these scan parameters result in a 3D scan with 65000 range measurements taken in 4.8s. In general faster 3D scans are possible but they would limit the perceptibility of small vertical objects. With the taken parameters it is possible to detect palls and trees up to 25m. The 3D data perception was done at equidistant points on the way while the robot is driving. For relative distant measurements the odometry was used.

All algorithms that are needed for 3D Perception and SLAM are calculated in real-time on a Pentium III 700Mhz embedded PC.

A. Colored 2D map representation

The described scan based SLAM algorithm was used to autonomously generate the *Colored 2D map* while the robot was driving a course around the bike rack. The generated 2D environment map in Fig. 8 has a size of 60m x 50m and shows the campus scene of Fig. 7 from a top view.

The red points are representing the measurements of the extracted virtual 2D landmark scans. Only these points are used to generate the consistent environment map. As seen in the picture only measured points of flat-surfaces are used as landmarks. Objects like buildings, container and tree trunks as well as thick branches of the tree tops are very useful as natural landmarks. Also the cross bar of the steel monument, as seen in the center background of Fig. 7, is used as a natural landmark. On the other hand detected points of very cluttered objects, e.g. tree tops, bushes or bikes, are filtered out. These points would degrade the scan matching result which could lead into an inconsistent environment map.

The blue points are representing the measurements of the extracted virtual 2D obstacle scans. All objects between the ground level and the top level of the robot are identified as obstacles. These points are needed for a collision free path planning in this urban environment. In the *Colored 2D map* of Fig. 8 various obstacles can be seen. In the lower left side a parking space with different cars on it is visible. On the right hand side we see a lot of bikes in front of a building. The facade of the building is basically used as a landmark for SLAM, but in the area of the front door no bikes are parked, so it is also identified as an obstacle. In the middle of the map a large bike rack between two trees can be seen. The tree trunks are mapped twice, as landmark as well as obstacle point. However, the bikes in the bike racks are only seen as obstacles because of their cluttered surfaces.

The bottom line of the tree tops as well as the cross bar of the steel monuments are higher than the size of the robot. Therefore navigation underneath these objects is possible. This is the reason why measured points of these objects are not included in the virtual obstacle scan.

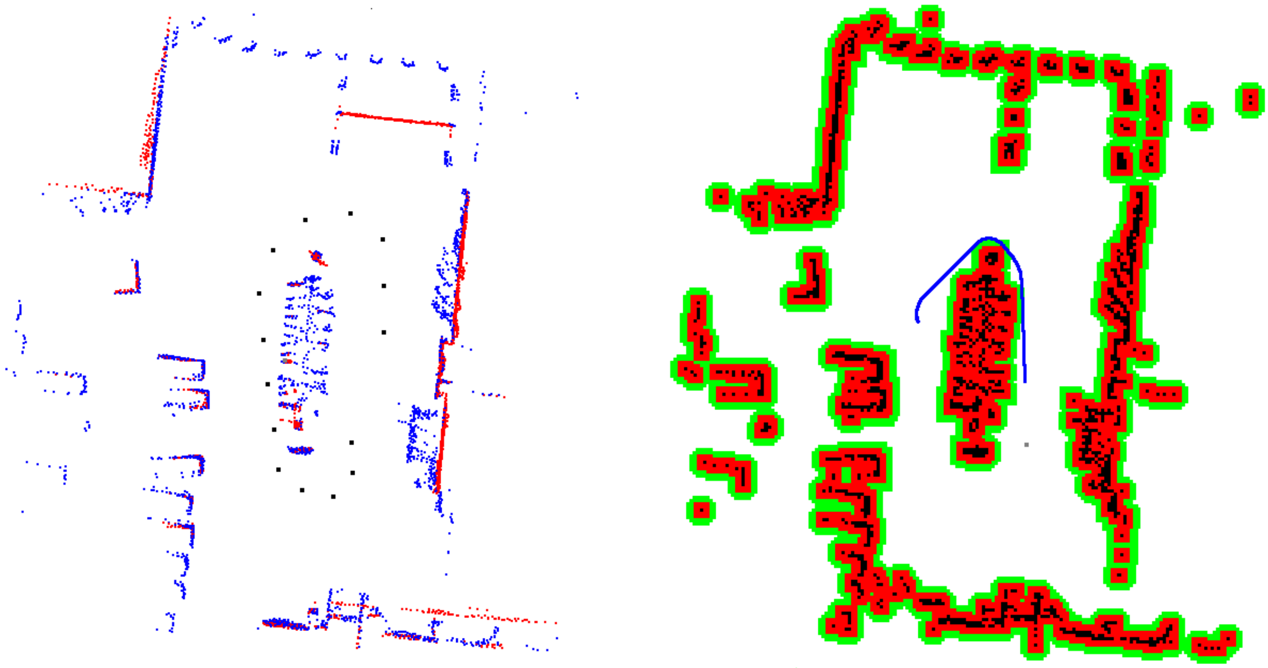

Figure 8. Left: A Colored 2D map of the robots environment including landmark points (red), obstacle points (blue) and positions of data acquisition (black). Right: Grid map generated from the obstacle points of the scan based map which was used for robot path planning (blue line).

The black points represent the equidistant points of sensor fusion respectively the optimized position of the network nodes of the scan based SLAM algorithm.

B. *Collision free path planning in 3D environments*

The obstacle points of the *Colored 2D map* are used to generate an obstacle grid map (Fig. 8 right). The black grids are representing the identified obstacles in the environment. The red grids are needed as a safezone to model the robot as a moving point. The size of the safezone is depending on the size of the robot. The additional green grids are representing avoid obstacle zones to bar the A*-algorithm from too closely passing the objects. Based on this grid map the robot has planned a collision free path for autonomous navigation (blue line).

By using the virtual obstacle points of the *Colored 2D map* the calculated path can consider 3D object geometries of the environment. The huge information from very complex 3D object surfaces is reduced into 2D and can be used for a collision free path planning online. Furthermore, the robot can decide online if an object can be passed underneath, or if there is a danger of collision.

V. CONCLUSION

In this paper we presented a navigation system for mobile robots working in urban environments. It was shown that the use of 3D sensors has advantages over pure 2D systems. Our 3D perception system is able to extract natural landmarks, even from partially occluded buildings and trees. These good landmarks are used with a scan-based SLAM algorithm able to build consistent environment maps. At the same time it is possible to extract obstacles on different heights from 3D data. Even overlapping object can be detected as obstacles if the headroom is smaller than the robot height.

To reduce the computational costs of 3D algorithms we use virtual 2D scans that allow an efficient combination of 3D perception and 2D localization, mapping and path planning. With this paper we contributed the ability to use 3D data in form of virtual 2D scans not only for SLAM but also for path planning and navigation. To achieve this the Colored 2D map was introduced. The benefit of this Colored 2D map is the ability to hold different information for localization and path planning in a consistent way.

Over all it was possible to demonstrate a mobile robot that is able to navigation with 3D data in real-time.

REFERENCES

[1] D.F. Huber, and M. Hebert, “A New Approach to 3-D Terrain Mapping”, International Conference on Intelligent Robots and Systems (IROS), Kyongju, Korea, October 1999.

[2] S. Lacroix, A. Mallet, D. Bonnafous, G. Bauzil; S. Fleury, M. Herrb, and R. Chatila, “Autonomous Rover Navigation on Unknown Terrains: Functions and Integration”, International Journal of Robotics Research, vol. 21, issue 10, October 2002.

[3] H. P. Moravec, “Robot Spatial Perception by Stereoscopic Vision and 3D Evidence Grids”, Tech Report CMU-RI-TR-96-34, Pittsburgh, USA, September 1996.

[4] P. Forsman, “Three-Dimensional Localisation and Mapping of Static Environments by Means of Mobile Perception”, Doctoral Thesis Helsinki University of Technology, Helsinki, Finland, November 2001.

[5] R. Unnikrishnan, and M. Hebert, “Robust Extraction of Multiple Structures from Non-uniformly Sampled Data”, International Conference on Intelligent Robots and Systems (IROS), Las Vegas, USA, October 2003.

[6] C. Brenneke, O. Wulf, and B. Wagner, “Using 3D Laser Range Data for SLAM in Outdoor Environments”, International Conference on Intelligent Robots and Systems (IROS), Las Vegas, USA, October 2003.

[7] O. Wulf, K. O. Arras, H. I. Christensen, and B. Wagner, “2D Mapping of cluttered Indoor Environments by Means of 3D Perception”, International Conference on Robotics and Automation, New Orleans (ICRA), USA, April 2004.

[8] O. Wulf, and B. Wagner, “Fast 3D Scanning Methods for Laser Measurement Systems”, International Conference on Control Systems and Computer Science (CSCS), vol 1, Bucharest, Romania, July 2003.

[9] A. Talukder, R. Manduchi, A. Rankin, and L. Matthies, “Fast and Relaiable Obstacle Detection and Segmentation for Cross-country Navigation”, Intelligent Vehicle Symposium, Versaille, France, June 2002.

[10] C. Brenneke, and B. Wagner, “A scan based navigation system for autonomous operation of mobile robots in man-made environments”, International Conference of Systems Engineering (ICSE), Coventry, Great Britain, September 2003.

[11] Brenneke, C.: "Ein scanbasierter Ansatz zur exploratorischen Navigation autonomer mobiler Systeme in unstrukturierten Outdoor-Umgebungen", dissertation.de - Verlag im Internet GmbH, Berlin, 2004, ISBN 89825-778-9.

[12] F. Lu, “Shape registration using optimization for mobile robot navigation”, Doctoral thesis at the University of Toronto, 1995.

[13] Nilsson, N.J.: “Principles of Artificial Intelligence”, Tioga Publishing Company, 1980.