Grid based next best view planning for an autonomous robot in indoor environments

Marcus Strand and Rüdiger Dillmann

Abstract— **Autonomous navigation in 3D environments becomes more and more popular. Especially the topic of autonomous exploration is important for hazardous environments. Besides the reliable collection of 3D information and the corresponding integration of the data into a model, the planning of the next viewpoint is crucial for the safety and the effectivity of the exploration robot.**

This article proposes a reduction of the known environment onto a 2D navigation and planning grid. It describes that this highly condensed representation of the environment is sufficient for classification of navigation relevant features (surface, door, obstacle etc.) and furthermore for planning the next best position of the robot.

I. Introduction

Consistent environment models are a keystone to achieve real autonomy in mobile robotics. Since these models are not always available in any working scenario, the robot has to build these models at least partially by itself. In recent years 3D-models have been introduced to make reliable robot navigation possible. While 2D-models still lack in expressiveness for robust navigation, 3D-models cover the full workspace of the robot and hence deliver the complete geometrical information of an environment.

The 3D-model is built by capturing successive depth images. The origin of these depth images can be estimated by using the robot's odometry and a registration algorithm. For the acquisition of 3D geometrical data different variations of moving 2D scanners have been already been proposed. In [7], [17], [9], [4], [3] or [18] single scan lines are taken with a horizontal movement. This approach can generate a good model of the environment, but cannot avoid collisions. Depth image based approaches can be found in [19],[11], [1]. Here first steps towards autonomy or semi-autonomy are taken. The taken depth images show a high resolution in (for our approach) uninteresting areas like the ceiling [19] or the area laterally [11] of the robot. For concurrent collision avoidance and model building with a single sensor device it is important to have the highest resolution in the heading direction of the robot. The current commercial available 3D scanners have a very small viewing cone, which is not optimal for registration and do not support foveal vision. To overcome these problems we developed a 3D sensor called RoSi (Rotating Sick) which is a common Sick LMS200 scanner rotating continuously around its optical axis. Details can be found in [10].

For the registration of depth images Paul Besl and Neil McKay introduced in [2] the widely used Iterative Closest Point (ICP) algorithm. In this algorithm a transformation of a range image D into the frame of a range image M is determined starting with an initial transformation. For each point in D a corresponding point in M is determined by the distance of the points. Minimizing the sum of all distances of corresponding points is the goal of the ICP algorithm. Different variants of the ICP algorithm have been already proposed. In [14] a classification is done. Surmann et al. use a comparable 3D laser scanner and present their ICP variant in [11] and [16]. In the matching stage they use the improved approximated kD-tree and a SVD to minimize the distance between the corresponding point pairs. Furthermore they present two algorithms to minimize the global error of all range images.

The arrangement of different data sets can be solved using graph based approaches in 2D [12] or 3D (elastic view graphs [8]).

In our case, according to the measuring method of the RoSi scanner the center of range images has the highest point resolution which decreases at the borders. This raises difficulties for the registration of two range images, since the influences of the center are enormous, due to the high density of points. If the centers of the range images are far away from each others and the overlapping area is low, it occurs that the points in the center pull the registration results in the wrong way. Therefore one first requirement on selection and matching strategies has to be to thin out the centers to gain constant point resolution for the registration process.

Furthermore it is advantageous to register only the silhouettes of the range images in the first iterations of the ICP algorithm and then refill the silhouettes with more details over following iterations in order to have an even vague estimation at any time. Hence the ICP algorithm first overlays the silhouettes and then improves its registration results with more and more details. This makes the ICP algorithm more robust against bad odometry data and accelerates it in the first iterations. See [6] for details on the registration process.

The determination of the next optimal position for data capture is known as next-best-view problem. In [5] this problem has been addressed initially for small objects. Later these approaches have been extended to larger environments. Due to their simplicity and their robustness occupancy grids have widely been used for next-best-view determination. Here, a discretization into cells of the environment is carried out. Each cell contains information about the occupancy of the

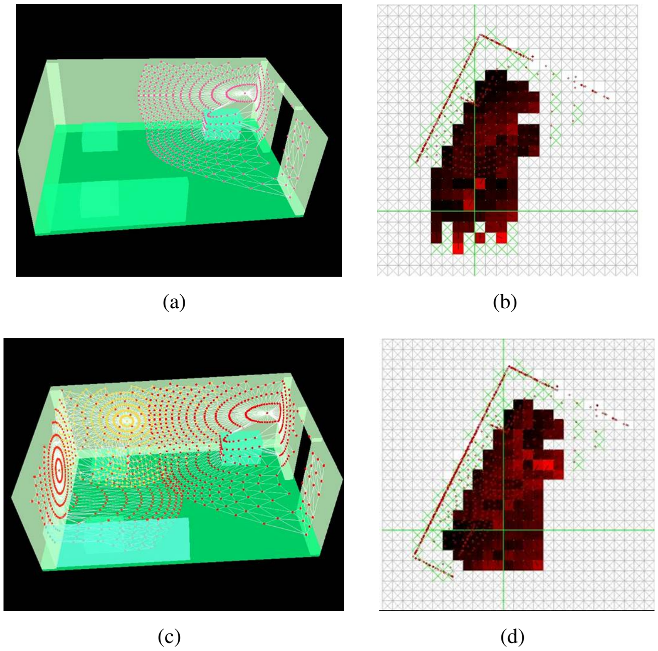
M. Strand is with the Division Intelligent Systems and Production Engineering (ISPE), Research Center for Information Technology - FZI, 76131 Karlsruhe, Germany strand@fzi.de

R. Dillmann is with the Institute for Anthropomatics, Universtity of Karlsruhe, Karlsruhe Institute of Technology - KIT, 76131 Karlsruhe, Germany dillmann@ira.uka.de

covered area. The extension from binary occupation states to a occupation probability has also been already proposed. Here the new cell value is determined using the Bayes' Rule [15]. This technique is useful if the cell value changes frequently (with noisy sensors or in dynamic environments). In contradiction to that, this article proposes a grid-styled structure as navigation map, containing navigation relevant 3D information. From this map navigation relevant features like ground surface, doors, obstacles can easily be extracted. The article is organized as follows. Section 2 covers the attributed 2D navigation map. Section 3 covers the planning of the next desired robot position, whereas in section 4 the exploration strategy is described. Two remaining sections show experimental results resp. give a conclusion.

II. The attributed 2D navigation grid

For the determination of the next best view the sensed data has to be segmented into different areas (like unknown, occupied, passable). To do this efficiently the available large amount of data is reduced to a planning model. With the planning model, next views can be calculated involving different quality criteria (like length of travelling distance, amount of overlap). The planning model is a 2D-grid which conserves the available navigation-specific information of the 3D world in cell attributes through projection onto the grid. By means of that the advantages of a 3D representation (geometrical information in every height) and 2D representation (easy to process, high data reduction) can be combined. Figure 1 shows the 3D model and the corresponding 2D grid in a simulation environment.


(a)

(b)

(c)

(d)

Fig. 1. Point clouds in a simulation environment (a),(c) and the corresponding 2D planning model (b),(d). In the planning model the quality of the cell is given in the brightness of the color. The method of determination of the quality of each cell is explained in the next section.

The cell attributes can be divided into direct attributes and indirect attributes. Direct attributes determine their value directly from the depth images. This group contains

- **isExplored:** is $true$ if the cell contains already a scan point
- **min,max:** define the minimal an maximal height values of a cell
- **hasLowObstacle:** is $true$ if the cell contains scan points which leads to a collision with the exploration platform
- **hasHighObstacle:** is $true$ if the cell contains scan points which are higher than the exploration platform

The group of indirect attributes contains

- **isWall:** is $true$ if the cell contains a high and a low obstacle
- **isGround:** is $true$ if the cell contains no low obstacles
- **isNearObstacle:** is $true$ if the cell contains no obstacles but is close to cells with low obstacles

In order to detect negative obstacles like stairs or holes all cells with attribute $isGround$ set are compared to their neighbours. Hereby the difference of the min values is evaluated. If this difference is over a threshold, it is assumed that there is a step and hence this area is not suited for driving.

III. Next-best-view Planning

A. Rating of cells

For every cell with the $isGround$ attribute set and which is reachable from the current position, horizontal scans with variable step width are simulated. For the determination of the rating G of a simulated scan, a simulated laser beam is sent, and the attribute of every cell it traverses is considered. The beam is followed until is reaches a cell with attribute $isWall$ or $!isExplored$. Both attributes lead to a termination of the process.

If there is a minimal count of unexplored and wall cells, the amount for both termination types are weighed and added. Additionally the values are weighed with a distance dependent piecewise linear function (see figure2). As one can see in the figure, close cells or cells far off are treated with minor quality. This prevents the system of choosing cells too close and too far from walls and unexplored areas as next best view.

Depending on the desired weight of overlapping resp.

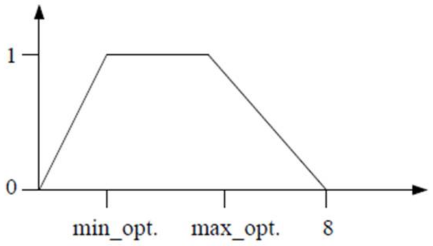
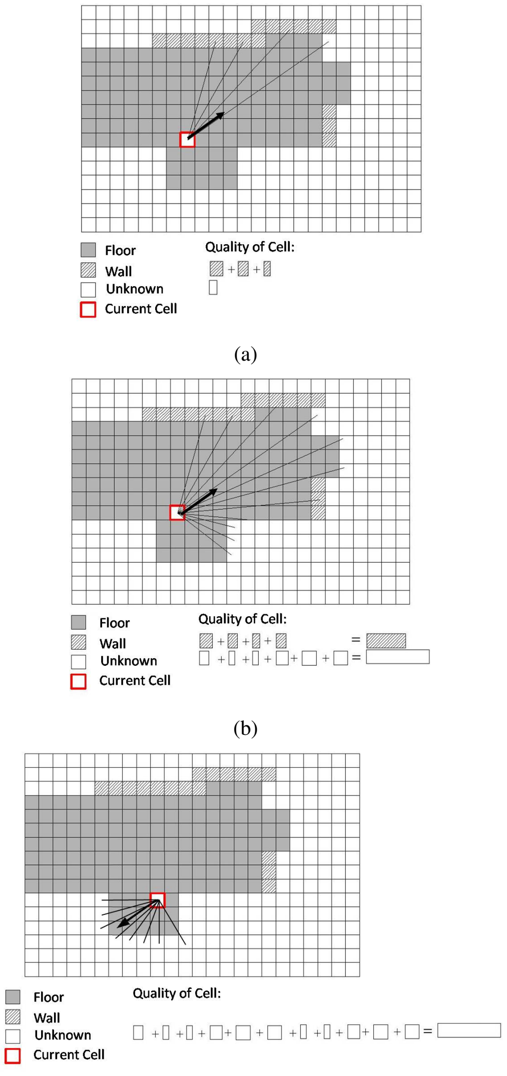


Fig. 2. Distance depended weighting

unexplored areas a quality of each cell and direction is found. The course of the qualities of all cells we call the quality function. Figures 3a,b show two stages of the evaluation of the quality of a cell. The quality of the cell and direction shown is high because the amount of known and unknown cells is balanced and the distances to the next wall and unexplored cells are reasonable.

Figure 3c shows the same cell in another direction. The quality is low, because no known cells are expected to be seen from this position and the unexplored cells are close to the position. If the room-type is determined as *floor* also the travelling distance to the next position is considered in order to avoid oscillations on the floor.



(a)

(b)

(c)

Fig. 3. Rating of cells depending on the amount and distance to unexplored and explored cells

B. Path Planning

In order to find a path towards the next best view cell, the neighbours of the respective cells are examined, and the neighbouring cell with the shortest distance is chosen. Afterwards the path is segmented into straight lines using direct obstacle free lines along the path. Therefore are direct test line from the current start to the next cells along the path is examined for collisions with obstacles. The last cell without collision is treated as segment.

IV. Exploration strategy

The indoor environment, which is explored is supposed to consist of rooms and floors. Figure 4 shows a possible indoor environment. Since the maximal scanner range is at 8 m the explored rooms should have an extent below 8 m. If there is a high amount of unreflected laser beams (and hence a high amount of points at the 8 m range) the system will detect a floor. If most of the laser beams are reflected the system detects a room. Besides rooms and floors a collision-free navigation through doors has to be managed separately. To take advantage of the loop closing operation without handling complex registration graphs the exploration strategy is split into two stages. In the first stage a loop with several overlapping depth images is created and closed through the loop closing algorithm. Missing areas of a room or floor (e.g. through occlusion) are sensed in the second stage an registered against the loop-closed depth images. In a last step open doors are searched to explore a new room or floor. It is assumed that doors have the $hasHighObstacle$ attribute set and lack the $hasLowObstacle$ attribute. Furthermore doors cells border to cells with $isWall$ attribute set. If the width of the so found door is below a threshold and over the robot width a door is found. Figure 5 depicts the strategy.

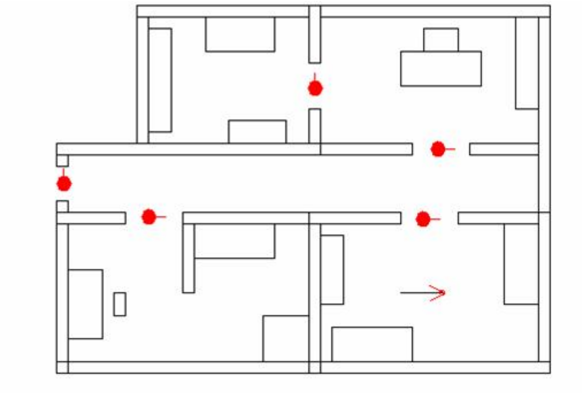


Fig. 4. Sample indoor environment

To apply the different stages several modi have been implemented:

- **MakeFirstScan:** This is the initial state of the system. One depth image is taken from the current position.
- **FastRoomScan:** This is the first stage. n depth images are taken with an angular distance of $\frac{360}{n}$ at one position. After that the loop is closed and the decision between room and floor is made.
- **ExploreRoom:** With help of the room quality function and the 2D-grid the rest of the room is explored. The corresponding depth image for registration is determined on the 2D-grid.

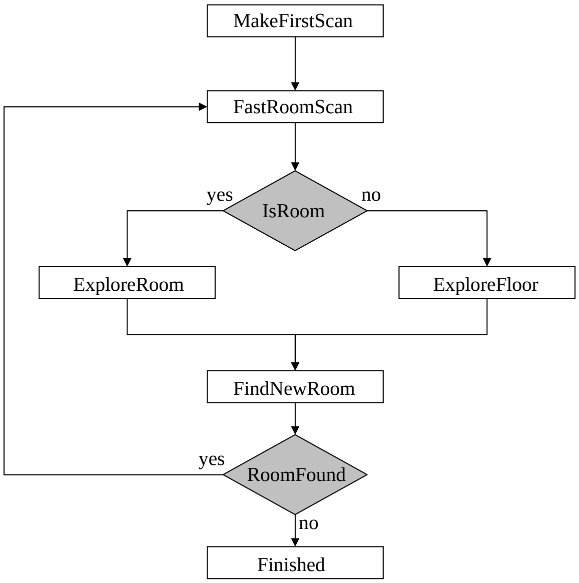


Fig. 5. Diagram of the exploration strategy

- **ExploreFloor:** With help of the floor quality function and the 2D-grid the rest of the floor is explored. The corresponding depth image for registration is determined on the 2D-grid.
- **FindNewRoom:** A new door is found on the 2D-grid and a new room is approached.
- **Finished:** If no open door is found the exploration is finished.

V. Experimental Results

This section contains the description and the results of the experiments carried out. First of all, a method of comparing the results against ground truth is proposed and the used hardware is introduced. Afterwards three experiments carried out in two different environments are described.

A. Comparability of 3D models

In order to compare the resulting data sets against ground truth the sets are merged offline to one reduced set with equal point resolution. With a marching intersections approach similar to [13] the scene is recalculated.

In order to get ground truth information the robot is additionally equipped with a horizontal 2D-laserscanner. This scanner is used for scanmatching against a previously manually modelled environment so that it can be assumed the position of the robot is known. The data sets collected in this way are also merged to one set. If the initial position of the ground truth model and the explored model are equal the models can be compared, since there is no offset. In order to compare the model M with N points to a model K assumed as ground truth a method based on the differences between the points and the surfaces, resp. triangles is used. For that, all triangles d of model K and all points p of model M are determined. After that the minimal distance mindist_i of a point $\vec{p}_i$ to all the triangles d is determined and interpreted as error value, so that

$$\text{mindist} = \min(\text{dist}(p_i, \forall d))$$

with *dist(p,d)* as signed Euclidian distance between point $\vec{p}$ and triangle d.

The sums μ and δ with

$$\mu = \frac{\sum_{i=1}^{N} \text{mindist}_i}{N}$$

and

$$\delta^2 = \frac{\sum_{i=1}^{N} \text{mindist}_i^2}{N}$$

are calculated.

The sum μ is the mean difference of the points to the next triangle. Through

$$\sigma = \sqrt{\delta^2 - \mu}$$

the standard deviation σ can be determined. This value together with the mean value μ reflect the quality of the explored model, since the mean value μ itself can be compensated, and hence can be small even in erroneous models. Together with the standard deviation σ one can analyse the distribution of false model points and still extract some information on the direction of the average offset of the model.

Table I gives an example for the values μ and σ for the same scene with different offsets. It is obvious that even with no offset the models are not equal due to the noise in the scanner data.

TABLE I
Evaluation results of scenes with offset

Offset	Mean [cm]	Deviation [cm]
0cm	0,17	19,8
3cm	-11,7	33,5
9cm	-28,6	59,8
15cm	-48,4	90,6
21cm	-70,4	135,5

B. Hardware Setup

Different experiments were carried out in the laboratory environment and outside the laboratory environment in an completely unknown environment of a former pediatric clinic. The experiments were carried out with the $Rosete$-platform, which is shown in figure 6. As main computer an embedded PC with 1GHz clock rate is used. The mobile platform uses a pentium III processor with 1,4GHz. Communication between the two system is done via LAN. Each depth image consists of 64800 single points which leads to a angular resolution of $0.5°$. The maximal velocity of the mobile platform is $v = 1.5\frac{m}{s}$.

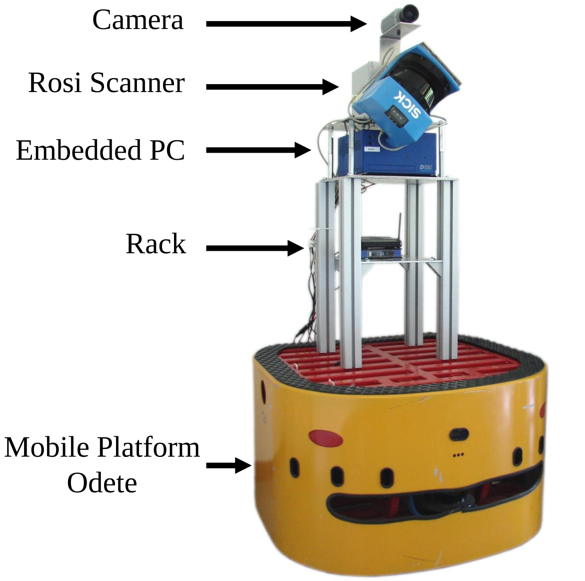


Fig. 6. Exploration system *Rosete*

C. Experiments

Several experiments in different environments were carried out. One experiment with 3 rooms and one floor was carried out in the laboratory environment. The explored area consists of about 170 sqm. Figure 7 depicts several stages of the autonomous exploration. It lasted about 33 minutes.

In every new room a **FastRoomScan** is carried out and the type *room* resp. *floor* is determined. Figure 7d shows the scene after the first room is explored. Since the center of the room is occupied with an obstacle, no more positions are found to complete the room and the next door is determined (depicted in blue).

After the door is traversed another **FastRoomScan** is carried out (figure 7e). Due to the high amount of non-reflected laser beams, the room-type is detected as *floor*. The exploration continues with the depicted next position. Figure 7f shows the scene after completion of the first half of the floor. Since this area is completed the second half of the floor is explored. Since the travelling distance is high, one intermediate scan is taken for relocalization. Figure 7j shows the scene after completion of the floor. The next door is determined and traversed.

The next room is due to its compactness already after **FastRoomScan** completed. Figure 7k shows the scene after exploration of that room.

The next door to a non explored room is determined and the last room is approached. Again this room is completed with **FastRoomScan** and two additional scans. Figure 7l shows the 2D model after completion of the third room.

Since no more doors are found, the system terminates. The mean deviation of the points compared to the ground truth model is $\mu = 46,9cm$ with a standard deviation of $\sigma = 267,0cm$.

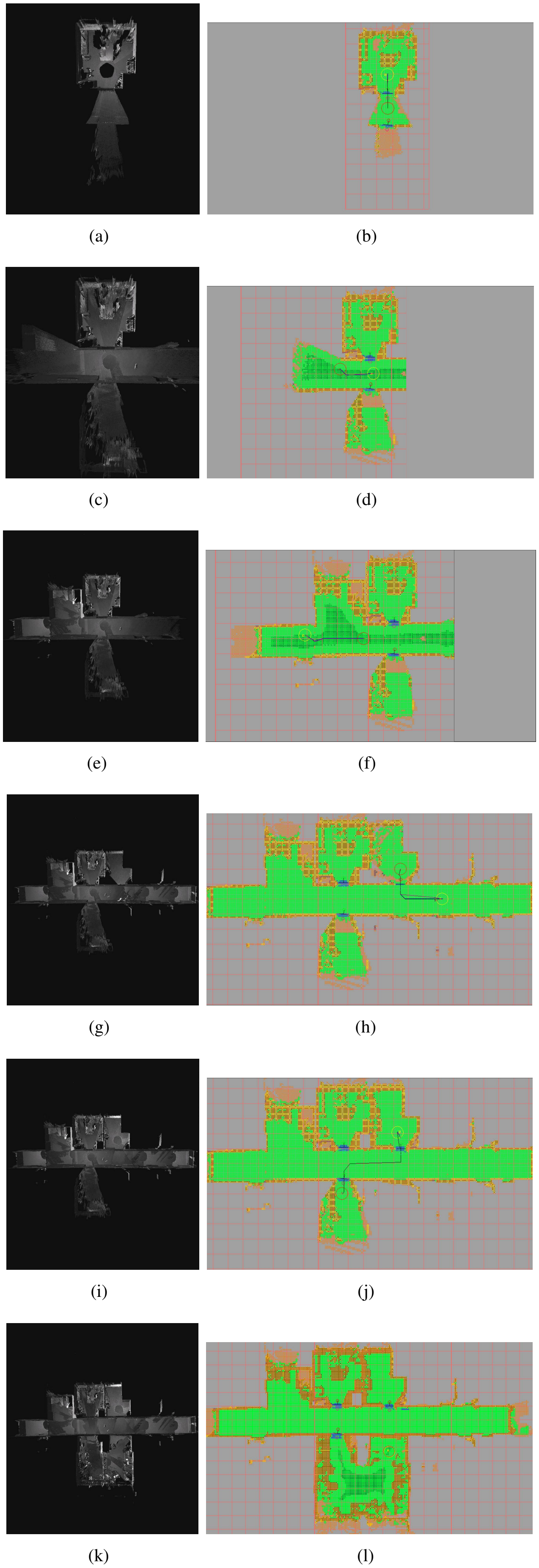


Fig. 7. Incremental Exploration of a floor and three rooms

VI. Conclusion

The article presented a system for autonomous 3D-modelling with laser range data and monocular vision. The developed 3D-sensor and the processing of the data was explained. The registration of depth images was done via a modified ICP-algorithm with octree-based matching and additional usage of color and remission values. Next best views are planned by means of a reduced 2D-grid which conserves navigation-important 3D information. In three experiments the application and the functionality of the system has been shown. The behaviour of the system in environments with slopes or stairs has not been examined yet. In flat environments it turned out that the trajectories may not be optimal, but the system can cope with rather complex environments. Furthermore the loop closing procedure may be optimized, since nested loops cannot be handled so far.

Further work will also focus on the integration of inclinometers for better registration results on slopes with known initial values for the roll- and pitch angles. Furthermore the planning strategy will be extended to more than two different types of rooms so that the system is also able to explore more different environments.

References

[1] H. Andreasson, R. Triebel, and W. Burgard. Improving plane extraction from 3d data by fusing laser data and vision. In *In Proc. of the International Conference on Intelligent Robots and Systems (IROS)*, 2005.

[2] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE transactions on pattern analysis and machine intelligence*, 14, 1992.

[3] F. Blais, J.A. Beraldin, S.F. El-Hakim, P. Boulanger, and G. Roth. A mobile system for indoor 3-d mapping and positioning. In *In Proceedings of the International Symposium on Real-Time Imaging and Dynamic Analysis*, 1998.

[4] W. Burgard, S.Thrun, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2000.

[5] C. J. Connolly. The determination of next best views. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 432 – 435, 1985.

[6] Ruediger Dillmann Frank Erb, Marcus Strand. Range image registration using an octree based matching strategy. In *Proceedings of the IEEE International conference on Mechatronics and Automation (ICMA), Harbin*, 2007.

[7] Christian Früh. *Automated 3D Model Generation for Urban Environments*. PhD thesis, Universität Karlsruhe, 2002.

[8] Peter Kohlhepp, Marcus Strand, Georg Bretthauer, and Ruediger Dillmann. The Elastic View Graph framework for autonomous, surface-based 3D-SLAM. *at-Automatisierungstechnik*, pages 190–199, 2007.

[9] S. Kristensen and P. Jensfelt. Active global localisation for a moble robot using multiple hypothesis tracking. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1999.

[10] Ruediger Dillmann Marcus Strand, Peter Steinhaus. A foveal 3d laser scanner integrating texture into range data. In *Proceedings of the 9th international conference on Intelligent Autonomous Systems (IAS9), Tokyo*, 2006.

[11] A. Nuechter, K. Lingemann, J. Hertzberg, and H. Surmann. Heuristic-based laser scan matching for outdoor 6d slam. In *Proceedings of the 28th Annual German Conference on AI*, 2005.

[12] Edwin Olson, John Leonard, and Seth Teller. Fast iterative optimization of pose graphs with poor initial estimates. pages 2262–2269, 2006.

[13] Claudio Rocchini and et al. The marching intersections algorithm for merging range images. *The Visual Computer*, 20:149–164, 2004.

[14] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling (3DIM)*, 2001.

[15] Cyrill Stachniss. *Exploration and Mapping with mobile Robots*. PhD thesis, Universität Freiburg, 2006.

[16] H. Surmann, A. Nuechter, K. Lingemann, and J.Hertzberg. 28th annual german conference on ai. In *Proceedings of the 28th Annual German Conference on Artificial Intelligence*, 2005.

[17] S. Thrun, D. Hähnel, and W. Brugard. Learning compact 3d models of indoor and outdoor environments with a mobile robot. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2001.

[18] A. Walthelm and A. M. Momlouk. Multisensoric active spatial exploration and modeling. In *Dynamische Perzeption: Workshop der GI-Fachgruppe 1.0.4*, 2000.

[19] Oliver Wulf, Bernardo Wagner, and Mohamed Khalaf-Allah. Using 3d data for monte carlo localization in complex indoor environments. In *Proceedings of the European Conference on mobile Robotics (ECMR)*, 2005.