Automatic Reconstruction of Colored 3D Models

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

Abstract. A basic issue of mobile robotics is the automatic generation of environment maps. This paper presents novel results for the reconstruction of textured 3D maps with an autonomous mobile robot, a 3D laser range finder and two pan-tilt color cameras. Building 3D maps involves a number of fundamental scientific issues. This paper adresses the issue of how to fuse the geometry data of the 3D laser range finder with camera images. The proposed algorithm allows to texturize geometrical 3D scenes-models.

1 Introduction

One fundamental problem in the design of autonomous mobile cognitive systems is the perception of the environment. A basic issue of mobile robotics is automatic map building of environments. Digital 3D models of the environment are needed in rescue and inspection robotics, facility management and architecture. Autonomous mobile robots equipped with 3D laser scanners are well suited for the gaging task [14]. To create realistic virtual realities from geometric models, textures, i.e., photos of the environments, have to be acquired and must be precisely mapped onto the scene. This mapping has to be computed automatically from the 3D point cloud of the scanned scene and the acquired photographs.
To compute the correct texture for a scanned scene, four steps are necessary: First, the cameras are calibrated; second, a meshing method generates a triangle mesh of the 3D data, and third, the texture for every triangle is chosen and mapped. Drawing to the screen is done by OpenGL. Finally and fourth, global color corrections are made to remove the systematic color and illumination differences between the individual texture maps. After discussing the state of the art in 3D reconstruction and presenting the robot Kurt3D these four steps are described in detail.

2 State of the Art

Some groups have attempted to build 3D volumetric representations of environments with 2D laser range finders. Thrun et al. [15], Früh et al. [6] and Zhao et al. [17] use two 2D laser range finder for acquiring 3D data. One laser scanner is mounted horizontally and one is mounted vertically. The latter one grabs a vertical scan line which is transformed into 3D points using the current robot pose. Since the vertical scanner is not able to scan sides of objects, Zhao et al. [17] use two additional vertical mounted 2D scanner shifted by 45° to reduce occlusion. The horizontal scanner is used to compute the robot pose. The precision of 3D data points depends on that pose and on the precision of the scanner. All these approaches have difficulties

to navigate around 3D obstacles with jutting out edges. They are only detected while passing them.
A few other groups use 3D laser scanners [12, 2]. 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 [12]. They used a RIEGL laser range finder on two robots, called EST and AEST (Autonomous Environmental Sensor for Telepresence). They use the Iterative Closest Points (ICP) algorithm [3] for scan matching and a perception planning module for minimizing occlusions. The AVENUE project develops a robot for modeling urban environments [2] using a CYRAX laser scanner. They match 3D scans with camera images semi automatically to yield a textured model.
Other techniques for acquiring range data are stereo vision and photogrammetry. Stereo vision has difficulties with producing dense depth maps and depends on the illumination to some degree. Photogrammetric methods produce high quality models that are textured. Nevertheless, these methods are usually manual and computationally expensive, thus cannot be computed in real time or on a mobile system, i.e., on a robot. Tab. 1 compares laser scanning with photogrammetry. State of the art photogrammetric methods [4] are combined with laser scanning to yield a measuring methodology that is more flexible. Dias et al. uses this combination to extract texture and to refine the model based on 3D laser scanning by images [4]. It is possible to complete the models in areas where data is missing or to increase the resolution in areas of high interest and 3D contents.

	Acquisition	Resolution	Lighting	3D measurment	Costs	Reliability
Laser scanning	large sensors	limited spatial resolution	*ext. light independent*	*directly by time of flight or phase shift*	high	*highly reliable*
Photogrammetry	*small cameras*	*high resolution photos*	measures ext. light	extract 3D from photos by correspondences	*low*	texture is required

Table 1: Comparison of laser scanning with photogrammetric model acquisition methods. Advantages are printed in red italic.

3 The Autonomous Mobile Robot Kurt3D

3.1 The Robot Platform

Kurt3D (Fig. 1, top left) is a mobile robot platform with a size of 45 cm (length) × 33 cm (width) × 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 increases 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, whereas 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.

3.2 The AIS 3D Laser Range Finder

The AIS 3D laser range finder (Fig. 1, middle) [14, 13] is built on the basis of a 2D range finder by extension with a mount and a small servodrive. The 2D laser range finder is attached in the center of rotation to the mount for achieving a controlled pitch motion and reducing

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

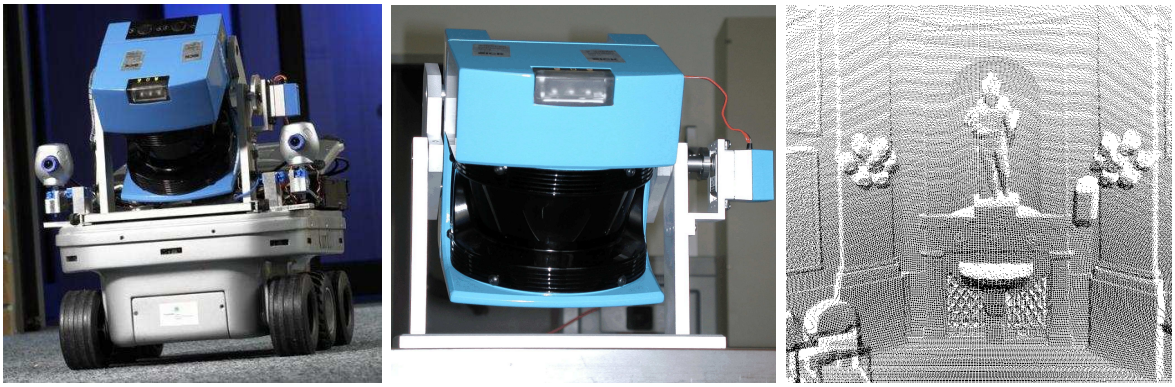

Figure 1: Left: The autonomous mobile robot Kurt3D equipped with the AIS 3D laser range finder. Middle: The AIS 3D laser range finder. Its technical basis is a SICK 2D laser range finder (LMS-200). Right: Scanned scene as point cloud (viewing pose 1 meter behind scanner pose).

torsional moments. On the left side, the high grade servo is connected. One battery charge (Scanner: 17 W, 20 NiMH cells with a capacity of 4500 mAh, Servo: 0.85 W, 4.5 V with batteries of 4500 mAh) is sufficient for 5h operating time.
The area of $180°(\text{h}) \times 120°(\text{v})$ is scanned with different horizontal (181, 361, 721) and vertical (210, 420) resolutions. A plane with 181 data points is scanned in 13 ms by 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 181×210 data points needs 2.8 seconds. Fig. 1 (top right) shows an example of a point cloud with a viewing pose one meter behind the scanner pose.

3.3 The Camera System

The camera system (Fig. 2, left) consists of two TerraCAM USB Pro webcams. They are equipped with a manual focus lens and the resolution is limited to 640×480 pixels with 7 fps as the maximum frame rate. To cover the whole area, scanned by the laser range finder, each camera needs to take 6 different images (Fig. 2, right). To handle this, the webcams are mounted on pan-tilt units each of which is based on 2 servo drives (Volz Micro-Maxx), one for the horizontal axis and the other for the vertical axis. Each axis can be rotated by $\pm 45°$. Due to the high-grade servo drives, an excellent repeat accuracy in positioning is guaranteed.

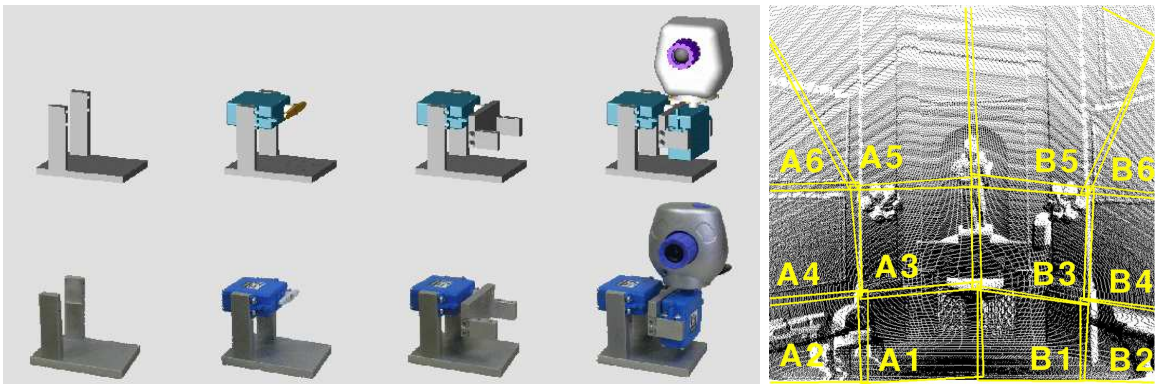

Figure 2: Left: The pan-tilt camera system. Right: Scanned scene as point cloud (viewing pose 2 meter behind scanner pose). The scene is covered by 12 camera images with some overlapping areas.

The webcams are powered over the USB interface and the servo drives are fed by the same batteries as the 3D laser range finder servo (cf. section 3.2).

4 Camera Calibration

The camera is modeled by the usual pinhole approach. A camera projects a 3D point $\mathbf{p} \in \mathbb{R}^3$ to the 2D image, resulting in $\mathbf{p}' \in \mathbb{R}^2$. The relationship between a 3D point $\mathbf{p}$ and its image projection $\mathbf{p}'$ is given by

$$s \begin{pmatrix} \mathbf{p}' \\ 1 \end{pmatrix} = \mathbf{A} \begin{pmatrix} & \mathbf{R} & & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix} \qquad \text{with} \quad \mathbf{A} = \begin{pmatrix} \alpha & \gamma & u \\ 0 & \beta & v \\ 0 & 0 & 1 \end{pmatrix}.$$

$\mathbf{A}$ is the camera matrix, i.e., internal camera parameters, with the principal point with the coordinates (u, v). $\mathbf{R}$ and $\mathbf{t}$ specify the external camera parameters, i.e., the orthonormal 3×3 rotation matrix and translation vector of the camera in the world coordinate system. In addition to these equations, we consider the distortion, resulting in 4 additional parameters to estimate [16].

Camera calibration uses a new technique based on Zhang's method. We give a brief sketch here, details can be found in [16]. The key idea behind Zhang's approach is to estimate the intrinsic, extrinsic and distortion camera parameters by a set of corresponding point. These 3D-to-2D point correspondences are first used to derive an analytical solution, i.e., the general 4×4 homography matrix $\mathbf{H}$ is estimated:

$$s \begin{pmatrix} \mathbf{p}' \\ 1 \end{pmatrix} = \mathbf{H} \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix}.$$

The estimation is done by solving an over specified system of linear equations. Since the points of 3D-to-2D point correspondences have usually small errors and only a few points are used to solve the equations for $\mathbf{H}$, this first estimation needs to be optimized. A nonlinear optimization technique, i.e, the Levenberg-Marquardt algorithm based on the maximum likelihood criterion is used to optimize the error term:

$$\sum_{i=1}^{n} \sum_{j=1}^{m} \left|\left| \mathbf{p}'_{i,j} - \hat{\mathbf{p}}'_{i,j}(\mathbf{H}_j) \right|\right|.$$

Hereby $\mathbf{p}'_{i,j}$ are the points $\mathbf{p}_{i,j}$ projected by $\mathbf{H}_j$, and $\hat{\mathbf{p}}_{i,j}$ are the given corresponding points; i is the point index and j is the image index. After the calculation of $\mathbf{H}$, the camera matrix $\mathbf{A}$, the rotation matrix $\mathbf{R}$ and the translation $\mathbf{t}$ are calculated from $\mathbf{H}$. Again, an over specified system of linear equation is solved, followed by a nonlinear optimization of

$$\sum_{i=1}^{n} \sum_{j=1}^{m} \left|\left| \mathbf{p}'_{i,j} - \hat{\mathbf{p}}'_{i,j}(\mathbf{A}, \mathbf{R}_j, \mathbf{t}_j) \right|\right|.$$

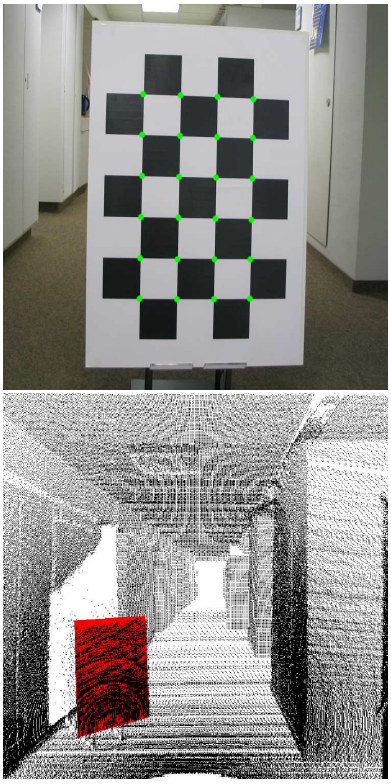

Figure 3: Top: Chessboard plane for calibration. Bottom: Chessboard detection in a 3D laser range scan.

Finally the term for optimization is set to

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left|\left|\mathbf{p}'_{i,j}-\hat{\mathbf{p}}'_{i,j}(\mathbf{A},\mathbf{R}_j,\mathbf{t}_j,k_1,k_2,l_1,l_2)\right|\right|,$$

where k_1, k_2 are parameters for the radial distortion, and l_1, l_2 the ones for tangential distortion. The minimum is found by the Levenberg-Marquardt algorithm that combines gradient descent and Gauss-Newton approaches for function minimization [5, 11]. An important concept of the Levenberg-Marquardt algorithm is the vector of residuals, i.e., $\mathbf{e}(\mathbf{a}) = \{E_i(\mathbf{a})\}_{i=1}N$, so that $E(\mathbf{a}) = ||\mathbf{e}(\mathbf{a})||$ is one of the error terms above. The goal at each iteration is to choose an update $\mathbf{x}$ to the current estimate $\mathbf{a}_c$, such that setting $\mathbf{a}_{c+1} = \mathbf{a}_c + \mathbf{x}$ reduces the error $E(\mathbf{a})$. A Taylor approximation of $E(\mathbf{a}+\mathbf{x})$ results in

$$E(\mathbf{a}+\mathbf{x}) = E(\mathbf{a}) + (\nabla E(\mathbf{a})\cdot\mathbf{x}) + \frac{1}{2!}((\nabla E(\mathbf{a})\cdot\mathbf{x})\cdot\mathbf{x}) + \cdots$$

Expressing this approximations in terms of $\mathbf{e}$ yields [5]:

$$\begin{aligned} E(\mathbf{a}) &= \mathbf{e}^T\mathbf{e} \\ \nabla E(\mathbf{a}) &= 2(\nabla\mathbf{e})^T\mathbf{e} \\ \nabla^2 E(\mathbf{a}) &= 2(\nabla^2\mathbf{e})\mathbf{e} + 2(\nabla\mathbf{e})^T\nabla\mathbf{e} \end{aligned}$$

By neglecting the therm $2(\nabla^2\mathbf{e})\mathbf{e}$ the Gauss-Newton approximation is derived, i.e.,

$$E(\mathbf{a}+\mathbf{x}) = \mathbf{e}^T\mathbf{e} + \mathbf{x}^T\mathbf{J}^T\mathbf{e} + \mathbf{x}^T\mathbf{J}^T\mathbf{J}\mathbf{x},$$

with $\mathbf{J}$ the Jacobian matrix ∇e, i.e., $J_{i,j} = \frac{\partial E_i}{\partial a_j}$. The task at each iteration is to determine a step $\mathbf{x}$ that will minimize $E(\mathbf{a}+\mathbf{x})$. Using the approximation of E differentiating with respect to $\mathbf{x}$ equating with zero, yields

$$\nabla_{\mathbf{x}} E(\mathbf{x}+\mathbf{a}) = \mathbf{J}^T\mathbf{e} + \mathbf{J}^T\mathbf{J}\mathbf{x} = 0.$$

Solving this equation for $\mathbf{x}$ yields the new Gauss-Newton update $\mathbf{x} = (\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{e}$. In contrast, the update with an accelerated gradient descent is given by $\mathbf{x} = \lambda^{-1}\mathbf{J}^T\mathbf{e}$ with λ denoting the increment between two gradient descent steps. In every iteration this new update is calculated by a combination, i.e., by

$$\mathbf{x} = (\mathbf{J}^T\mathbf{J} + \lambda\mathbb{1})^{-1}\mathbf{J}^T\mathbf{e}.$$

For the above camera calibration algorithm the 3D-to-2D point correspondences are essential. Calibration is done with a chess board. From the image the board pattern corners are extracted automatically (Fig. 3 top). The corresponding 3D points are automatically extracted based on the corners of a quad in 3D (Fig. 3 bottom). The calibration algorithm extracts the 3D quad from the scanned point cloud with a modified ICP (Iterative Closest Points) algorithm [3, 9]. Given a set of 3D scan points M, a quad is matched. The algorithm computes the rotation $\mathbf{R}$ and the translation $\mathbf{t}$, such that the distances the scan points $\mathbf{m}_i \in \mathbb{R}^3$ and their projection to the quad $\mathbf{d}_i \in \mathbb{R}^3$ is minimized, i.e.,

$$\sum_{i=1}^{N_m}||\mathbf{m}_i - (\mathbf{R}\mathbf{d}_i + \mathbf{t})||^2.$$

The ICP algorithm accomplishes the minimization iteratively. It computes first the projections and then minimizes the above error term in a closed form fashion [3, 9]. The closed form solution is based on the representation of a rotation as a quaternion as proposed by Horn [7].

Figure 4: Left: Schematic illustration of the texture mapping process. Middle and right: Final 3D texture mapping from two different view poses of the scene given in Fig. 2, right.

5 Mesh Generation

The 3D scanner gages the environment by a tilt rotation, thus the scan slices are ordered. Furthermore, in every scan slice the data is ordered counterclockwise. So, a simple algorithm creates a triangle mesh by connecting neighboring data points. A threshold for the side length of a triangle is used to handle jump egdes, i.e., a discontinuity on the surface. Larger meshes are created by connecting the (i, j)th neighbor, respectively.

6 Texture Mapping

In order to assign texture to the previously generated triangle mesh, the algorithm projects the vertices of each triangle to the images by utilizing the extrinsic, intrinsic and distortion parameters, computed during the camera calibration process (cf. section 4). Due to the lens quality of an ordinary USB webcam, modeling the lens distortion is essential.
To figure out which of the 12 images holds the correct texture information, the vertices are projected to all images. If one image does not cover the according part of the 3D scene the computed vertex coordinates will be out of the image range. If the algorithm gives valid coordinates for more than one of the images (e.g., for triangles in overlapping areas, compare Fig. 2), it uses the image in which the projected vertex coordinates are closest to the image center.
Based on these projected coordinates, an OpenGL-based viewer application cuts the texture out of the images and "glues" it onto the 3D triangle mesh. Due to this relation of 3D data and texture information, the scene can be rendered from different perspectives (Fig. 4, middle and right) as a textured 3D scene.[2]

7 Global Color Correction

Different illumination conditions and the camera technology prevent color continuity at the borders of each image, leading to observable discontinuities in the color and also brightness. Based on the ideas of Agathos and Fisher we use global corrections in order to diffuse the texture from each two different views [1] and to reduce the observable discontinuities. They motivate the assumption that there exists a linear transformation matrix $\mathbf{T}_{j\to k}$ to correct the jth view to the kth view, i.e.,

$$\mathbf{T}_{j\to k}\left(N_i^{(j)}(\lambda)\right) = N_i^{(k)}(\lambda),$$

[2] An animation of the scene can be found at `http://www.ais.fraunhofer.de/~nuechter/3DIM_video` [10].

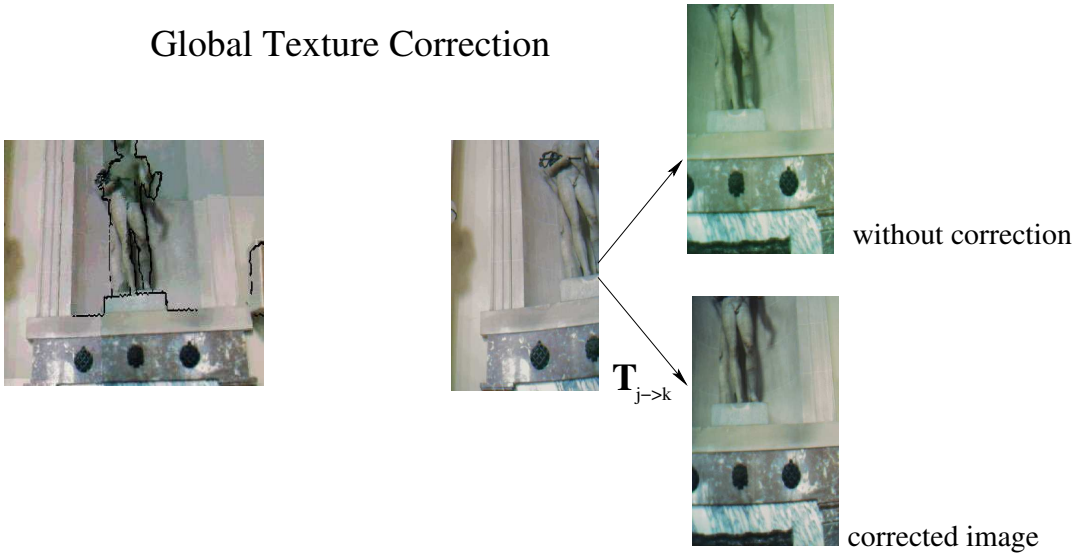

Figure 5: Left: Magnification of the textured 3D scene in Fig. 4, right. Right: Uncorrected and corrected photo No. B3. The correction is based on the pixels shared with photo No. A3. The photo numbers are defined in Fig. 2.

for $i \in \{\text{R,G,B}\}$. Two vectors of pixels $\mathbf{V}_k, \mathbf{V}_j$ are formed from the views k and j respectively. They contain R,G,B pixels from the overlap of the views. The global correction matrix $\mathbf{T}_{j\to k}$ is estimated as follows [1]:

$$\mathbf{V}_k = \mathbf{T}_{j\to k}\mathbf{V}_k \quad \Leftrightarrow \quad \mathbf{T}_{j\to k} = (\mathbf{V}_k\mathbf{V}_j^T)(\mathbf{V}_j\mathbf{V}_j)^{-1}$$

Fig. 5 shows an image part of the scene of Fig. 4 with an uncorrected and corrected image. The result is presented in Fig. 4 and shows still some color incontinuities that cannot be resolved with the correction. The method requires enough image overlap, precise 3D-to-2D calibration and sufficient input image quality.

8 Conclusions

This paper has presented a framework for the reconstruction of textured 3D maps with an autonomous mobile robot, a 3D laser range finder and two pan-tilt color cameras. The developed systems allows to gage environments in 3D and fuse the data with camera images. A wide range of applications using 3D models benefit from the proposed automatic acquisition method, e.g., virtual reality applications, architecture, factory and facility management and rescue and inspection robotics.
Needless to say, much work remains to be done. Future work will concentrate on four aspects:

- Build 3D maps with texture information involving 6D robot poses, including loop tours, i.e., 6D SLAM [9].
- Use local color/texture correction in addition to our global correction method.
- Calibrate the camera parameters dynamically without the chess board quad, i.e., calculate it from the 3D scene and the corresponding images.
- Generate high level descriptions and semantic maps of environments including the 3D information, e.g., in XML format. The semantic maps contain spatial 3D data with descriptions and labels [8].

Acknowledgment: Special thanks to Kai Lingemann and Matthias Hennig for supporting our work.

References

[1] A. Agathos and R. B. Fisher. Colour Texture Fusion of Multiple Range Images. In *Proceedings of the 4th IEEE International Conference on Recent Advances in 3D Digital Imaging and Modeling (3DIM '03)*, Banff, Canada, October 2003.

[2] P. Allen, I. Stamos, A. Gueorguiev, E. Gold, and P. Blaer. AVENUE: Automated Site Modeling in Urban Environments. In *Proceedings of the third International Conference on 3D Digital Imaging and Modeling (3DIM '01)*, Quebec City, Canada, May 2001.

[3] 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.

[4] P. Dias, V. Sequeira, F. Vaz, and J. G.M. Goncalves. Registration and Fusion of Intensity and Range Data for 3D Modelling of Real World Scenes. In *Proc. of the 4th IEEE Int. Conf. on Recent Advances in 3D Digital Imaging and Modeling (3DIM '03)*, Banff, Canada, October 2003.

[5] A.W. Fitzgibbon. Robust Registration of 2D and 3D Point Sets. In *Proceedings of the 11th British Machine Vision Conference (BMVC '01)*, 2001.

[6] C. Früh and A. Zakhor. 3D Model Generation for Cities Using Aerial Photographs and Ground Level Laser Scans. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR '01)*, Kauai, Hawaii, USA, December 2001.

[7] B. Horn. Closed–form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629 – 642, April 1987.

[8] A. Nüchter, H. Surmann, and J. Hertzberg. Automatic Classification of Objects in 3D Laser Range Scans. In *Proc. 8th Conf. on Intelligent Autonomous Systems*, pages 963 – 970, Amsterdam, The Netherlands, March 2004.

[9] A. Nüchter, H. Surmann, K. Lingemann, and J. Hertzberg. 6D SLAM - Preliminary Report on closing the loop in Six Dimensions. In *Proceedings of the 5th Symposium on Intelligent Autonomous Vehicles (IAV '04)*, Lissabon, Portugal, June 2004.

[10] A. Nüchter, H. Surmann, K. Lingemann, K. Pervölz, and J. Hertzberg. Video: Autonomous Mobile Robots for 3D Digitalization of Environments. In *Proceedings of the 4th IEEE International Conference on Recent Advances in 3D Digital Imaging and Modeling (3DIM '03)*, October 2003.

[11] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C : The Art of Scientific Computing*. Cambridge University Press, January 1993.

[12] 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.*, San Jose, CA, USA, January 1999.

[13] H. Surmann, K. Lingemann, A. Nüchter, and J. Hertzberg. A 3D laser range finder for autonomous mobile robots. In *Proceedings of the of the 32nd International Symposium on Robotics (ISR '01)*, pages 153 – 158, Seoul, Korea, April 2001.

[14] H. Surmann, A. Nüchter, and J. Hertzberg. An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *Robotics and Autonomous Systems*, 45:181 – 198, December 2003.

[15] S. Thrun, D. Fox, and W. Burgard. A real-time algorithm for mobile robot mapping with application to multi robot and 3D mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '00)*, San Francisco, CA, USA, April 2000.

[16] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(22):1330 – 1334, 2000.

[17] H. Zhao and R. Shibasaki. Reconstructing Textured CAD Model of Urban Environment Using Vehicle-Borne Laser Range Scanners and Line Cameras. In *Second International Workshop on Computer Vision System (ICVS '01)*, pages 284 – 295, Vancouver, Canada, July 2001.