3D MODELING OF INDOOR ENVIRONMENTS BY A MOBILE PLATFORM WITH A LASER SCANNER AND PANORAMIC CAMERA

Peter Biber[†], *Sven Fleck*[†], *Florian Busch*[†], *Michael Wand*[†], *Tom Duckett*[‡], *Wolfgang Strasser*[†]

email: {biber,fleck,busch,wand,strasser}@gris.uni-tuebingen.de, tom.duckett@tech.oru.se
[†] Wilhelm Schickard Institute, Graphical-Interactive Systems (WSI/GRIS), University of Tübingen, 72070 Tübingen, Germany
[‡]Applied Autonomous Sensor Systems (AASS), University of Örebro, 70182 Örebro, Sweden

ABSTRACT

One major challenge of 3DTV is content acquisition. Here, we present a method to acquire a realistic, visually convincing 3D model of indoor environments based on a mobile platform that is equipped with a laser range scanner and a panoramic camera. The data of the 2D laser scans are used to solve the simultaneous localization and mapping problem and to extract walls. Textures for walls and floor are built from the images of a calibrated panoramic camera. Multiresolution blending is used to hide seams in the generated textures. The scene is further enriched by 3D-geometry calculated from a graph cut stereo technique. We present experimental results from a moderately large real environment. [1]

1. INTRODUCTION

A 3D model can convey much more useful information than the typical 2D maps used in many applications. By combining vision and 2D laser range-finder data in a single representation, a textured 3D model can provide remote human observers with a rapid overview of the scene, enabling visualization of structures such as windows and stairs that cannot be seen in a 2D model. In the context of 3DTV such models can help planning camera paths and can provide realistic previews of large scenes with moderate effort.

We present an easy to use method to acquire such a model. A mobile robot equipped with a laser range scanner and a panoramic camera collects the data needed to generate a realistic, visually convincing 3D model of large indoor environments. Our geometric 3D model consists of planes that model the floor and walls (there is no ceiling, as the model is constructed from a set of bird's eye views). The geometry of the planes is extracted from the 2D laser range scanner data. Textures for the floor and the walls are generated from the images captured by the panoramic camera. Multiresolution blending is used to hide seams in the generated textures stemming, e.g., from intensity differences in the input images.

The scene is further enriched by 3D-geometry calculated from a graph cut stereo technique to include non-wall structures like stairs, tables, etc. An interactive editor allows fast postprocessing of the automatically generated stereo data to remove outliers or moving objects.

So our approach builds a hybrid model of the environment by extracting geometry and using image based approaches (texture mapping). A similar approach was applied by Früh and Zakhor [7] for generating a 3D model of downtown Berkley. A complete review of hybrid techniques is beyond the scope here and we refer to references in [7] and to the pioneering work of Debevec [5]. We believe that such hybrid techniques are superior to pure image based techniques like Aliaga's work [1] that needs advanced compression and caching techniques and still provides only a limited set of viewpoints (a single plane). The acquired indoor model presented here is much larger than other indoor models reported, yet it is possible to view it in our point cloud viewer from arbitrary viewpoints in real time.

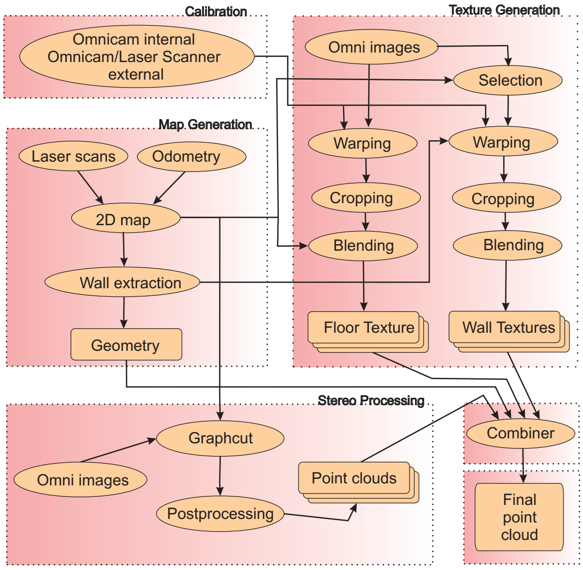

Figure 1: An overview of our method to build a 3D model of an indoor environment. Shown is the data flow between the different modules.

The main idea of our method to build a 3D model of an indoor environment is to remotely steering a mobile robot through it. At regular intervals, the robot records a laser scan, an odometry reading and an image from the panoramic camera. The robot platform is described in section 2. From this data, the 3D model is constructed. Fig. 1 gives an overview of the method and shows the data flow between the different modules. Five major steps can be identified as follows (the second step, data collection, is omitted from Fig. 1 for clarity).

1. Calibration of the robot's sensors.
2. Data collection.
3. Map generation
4. Texture generation
5. Stereo processing

Our method consists of manual, semi-automatic and automatic parts. Recording the data and calibration is done manually by teleoperation, and extraction of the walls is done semi-automatically with an user interface. Stereo matching is automatic, but selection of extracted 3D geometry and postprocessing includes semi-automatic and manual parts.

2. HARDWARE PLATFORM

The robot platform used in these experiments is an ActivMedia Peoplebot (see Fig. 3). It is equipped with a SICK LMS 200 laser scanner and a panoramic camera consisting of an ordinary CCD camera (interlaced and TV resolution) with an omni-directional lens attach-

[1] This work is supported by EC within FP6 under Grant 511568 with the acronym 3DTV.

ment (NetVision360 from Remote Reality). The panoramic camera has a viewing angle of almost 360 degrees (a small part of the image is occluded by the camera support) and is mounted on top of the robot looking downwards, at a height of approximately 1.6 meters above the ground plane. It has been calibrated before recording data using a calibration pattern.

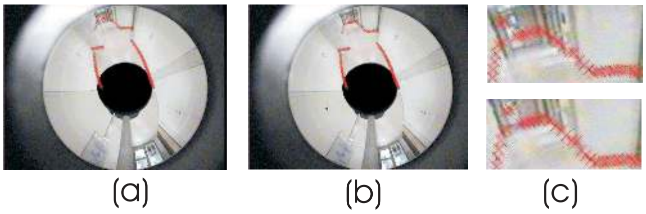

Figure 2: Joint external calibration of laser, panoramic camera and ground plane tries to accurately map a laser scan to the edge between floor and wall on the panoramic image. (a) without calibration (b) with calibration (c) zoom

All methods in the rest of the paper assume that the laser scanner and the panoramic camera are mounted parallel to the ground plane. It is difficult to achieve this in practice with sufficient precision. While a small slant of the laser scanner has less effect on the measured range values in indoor environments, a slant of the panoramic camera has considerably more effect. Fig. 2(a) shows one panoramic image along with the corresponding laser scan mapped onto the ground plane under the above assumption. Especially for distant walls, the alignment error is considerable. As a mapping like this is used to extract textures for walls, we have to correct this error.

A model for the joint relation between panoramic camera, laser scanner and ground plane using three parameters for the rotation of the panoramic camera turned out to be accurate enough. The parameters can be recovered automatically using full search (as the parameters' value range is small). To get a measure for the calibration, an edge image is calculated from the panoramic image. It is assumed that the edge between floor and wall produces also an edge on the edge image and therefore count the number of laser scan samples that are mapped to edges according to the calibration parameter. Fig 2(b) shows the result of the calibration: the laser scan is mapped correctly onto the edges of the floor.

3. BUILDING THE 2D MAP BY SCAN MATCHING

An accurate 2D map is the basis of our algorithm. This map is not only used to extract walls later, it is also important to get the pose of the robot at each time step. This pose is used to generate textures of the walls and floor and provides the external camera parameters for the stereo processing.

Our approach belongs to a family of techniques where the environment is represented by a graph of spatial relations obtained by scan matching [11, 8, 6]. The nodes of the graph represent the poses where the laser scans were recorded. The edges represent pairwise registrations of two scans. Such a registration is calculated by a scan matching algorithm, using the odometry as initial estimate. The scan matcher calculates a relative pose estimate where the scan match score is maximal, along with a quadratic function approximating this score around the optimal pose. The quadratic approximations are used to build an error function over the graph, which is optimized over all poses simultaneously (i.e., we have $3\times$ `nrScans` free parameters). Details of our method can be found in [3]. Fig. 3 shows a part of the map's graph and the final map used in this paper.

4. GENERATION OF GEOMETRY

The geometry of our 3D model consists of two parts: the floor and the walls. The floor is modeled by a single plane. Together with the texture generated in the next section, this is sufficient: the floor's texture is only generated where the laser scans indicate free space.

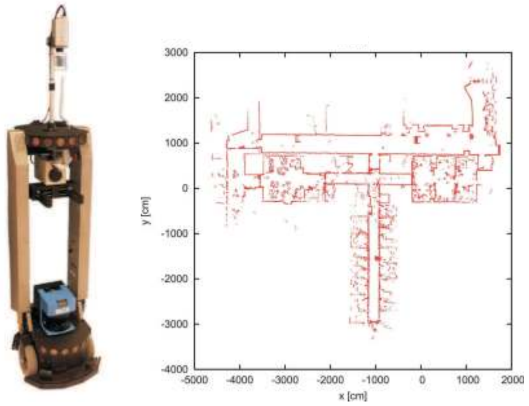

Figure 3: Robot used in our experiments and 2D map created by laser scan matching.

The walls form the central part of the model. Their generation is a semi-automatic step, for reasons described here. The automatic part of this process assumes that walls can be identified by finding lines formed by the samples of the laser scans. So in a first step, lines are detected in each single laser scan using standard techniques. The detected lines are projected into the global coordinate frame. There, lines seeming to correspond are fused to form longer lines. Also, the endpoints of two lines that seem to form a corner are adjusted to have the same position. In this way, we try to prevent holes in the generated walls.

This automatic process gives a good initial set of possible walls. However, the results of the automatic process are not satisfying in some situations. These include temporarily changing objects and linear features, which do not correspond to walls. Doors might open and close while recording data, and especially for doors separating corridors, it is more desirable not to classify them as walls. Otherwise, the way would be blocked for walk throughs. Also, several detected lines were caused by sofas or tables. Such objects not only caused the generation of false walls, they also occluded the real walls, which were then not detected. So we added a manual post-processing step, which allows the user to delete, edit and add new lines. Nearby endpoints of walls are again adjusted to have the same position. In a final step, the orientation of each wall is determined. This is done by checking the laser scan points that correspond to a wall. The wall is determined to be facing in the direction of the robot poses where the majority of the points were measured.

5. GENERATION OF TEXTURES

The generation of textures for walls and for the floor are similar. First, the input images are *warped* onto the planes assigned to walls and floor. A floor image is then cropped according to the laser scan data. Finally, corresponding generated textures from single images are fused using multi-resolution blending.

The calibration of the panoramic camera, the joint calibration of robot sensors and ground plane, and the pose at each time step allows for a simple basic acquisition of textures for floor and for walls from a single image. Both floor and walls are given by known planes in 3D: the floor is simply the ground plane, and a wall's plane is given by assigning the respective wall of the 2D map a height, following the assumption that walls rise orthogonally from the ground plane. Then textures can be generated from a single image by backward mapping (*warping*) with bilinear interpolation, as is included in many image processing packages.

The construction of the final texture for a single wall requires the following steps. First, the input images used to extract the textures are selected. Candidate images must be taken from a position such that the wall is facing towards this position. Otherwise, the image would be taken from the other side of the wall and would supply an incorrect texture. A score is calculated for each remaining image that measures the maximum resolution of the wall in this image. The resolution is given by the size in pixels that corresponds to a real world distance on the wall, measured at the closest point on the wall. This closest point additionally must not be occluded according to the laser scan taken at that position. A maximum of ten images is selected for each wall; these are selected in a greedy manner, such that the minimum score along the wall is at a maximum. If some position along the wall is occluded on all images, the nonocclusion constraint is ignored. This constraint entails also that image information is only extracted from the half of the image where laser scan data are available (the SICK laser scanner covers only 180°). Finally, a wall texture is created from each selected image, then these are fused using the blending method described in the following.

The generation of a floor texture from a single image is demonstrated in Fig. 4. The image is warped onto the ground plane. Then it is cropped according to the laser scanner range readings at this position, yielding a single floor image. This entails again that one half of the image is not used. Such a floor image is generated from each input image. Then, these images are mapped onto the global 2D coordinate frame.

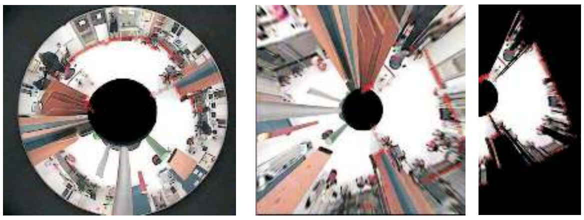

Figure 4: Generation of floor texture from a single image.

Both floor and wall textures are fused from multiple input images (Fig. 5 shows an example). The fusion is faced with several challenges, among them

- image brightness is not constant,
- calibration and registration may be not accurate enough,
- parts of the input image may be occluded by the robot or support of the panoramic camera, and
- walls may be occluded by objects in front of them and thus effects of parallax play a role.

Additionally, the quality along a wall texture degrades with the distance from the closest point to the robot position (this effect is due to scaling and can be seen clearly in Fig. 5). Similar effects can be observed for floor textures. These problems also exist in other contexts, e.g. [2, 12].

We use an adaption of Burt and Adelson multiresolution blending [4]. The goal of the algorithm is that visible seams between the images should be avoided by blending different frequency bands using different transition zones.

The outline is as follows: a Laplacian pyramid is calculated for each image to be blended. Each layer of this pyramid is blended separately with a constant transition zone. The result is obtained by reversing the actions that are needed to build the pyramid on the single blended layers. Typically, the distance from an image center is used to determine where the transition zones between different images should be placed. The motivation for this is that the image quality should be best in the center (consider e.g., radial distortion) and that the transition zones can get large (needed to blend low frequencies). To adapt to the situation here, we calculate a distance field for each texture to be blended, which simulates this “distance to the image center”. For the walls, this image center is placed at an x-position that corresponds to the closest point to the robot’s position (where the scaling factor is smallest). Using such a distance field, we can also mask out image parts (needed on the floor textures as in Fig.4 to mask both the region occluded by the robot and regions not classified as floor according to the laser scanner).

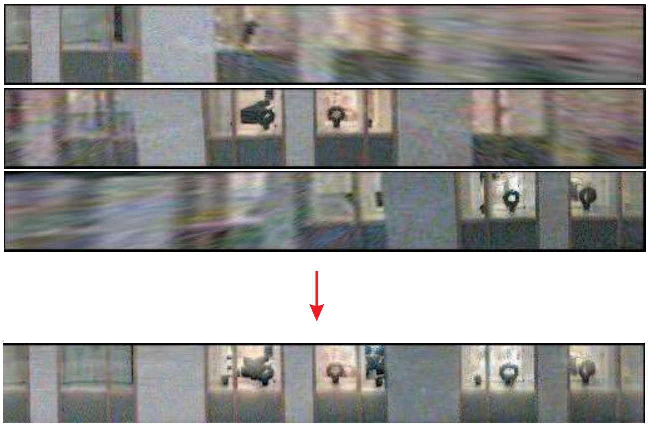

Figure 5: Final textures of walls are generated by blending multiple textures generated from single panoramic images. Shown here are three of ten textures which are fused into a single texture.

6. ACQUISITION OF ADDITIONAL 3D GEOMETRY

6.1 Stereo matching using graph cut

Thanks to the available camera positions and the calibrated camera we are in an ideal setting to apply stereo algorithms to the input images. A high-quality state of the art stereo algorithm - namely the *graph cut* algorithm by Kolmogorov and Zabih - is used to calculate a disparity map for each panoramic image. Our implementation is based upon the graph cut implementation of Per-Jonny Käck [9] that extends the publicly available source code of Kolmogorov and Zabih [10] with a robustified matching cost.

Our stereo matching pipeline consists of the following stages: First, for each pixel in the first image the epipolar curve in the second image is created, taking into account the epipolar geometry of our panoramic camera. This epipolar curve is represented by a set of points in image space where each point denotes a different disparity. Then, an error value for each disparity on this epipolar curve is computed and saved. These two stages then provide the data needed by the graph cut algorithm. The resulting disparity map is converted into a point cloud and postprocessed: regions around the epipoles are removed because these typically provide too few constraints to extract reliable depth information. In a further step depth values that belong to the floor with high probability are corrected to be exactly on the floor. Figure 6 shows one source image and the final disparity map after postprocessing. The point cloud from this figure (fused with the walls and floor model) is rendered in Fig. 10.

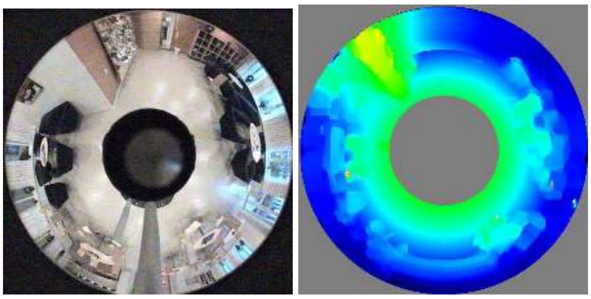

Figure 6: Panoramic image and disparity map calculated by the graph cut stereo algorithm.

6.2 Postprocessing

Point clouds created by the stereo matcher are combined applying some heuristics to suppress outliers. E.g., points are only counted as valid if they receive support also from other point clouds, points that are already represented by walls or by the floor are omitted. Finally the point clouds are combined. An interactive point cloud editor and renderer allows the user to select the objects supposed to be part of the final model, to delete outliers and to fill holes using filters. This tool uses features of modern graphics hardware (vertex and pixelshader) to allow fast rendering and editing of large point clouds (several million points). Future versions of this tool will also implement a hierarchical out-of-core mechanism to provide these capabilities on even larger point clouds that do not fit into memory.

7. RESULTS AND CONCLUSION

A data set of 602 images and laser scans was recorded at Örebro university by teleoperation. The built 2D-map was shown in Fig. 3. A screen shot of the resulting 3D model without stereo results can be seen in Fig. 7. This model can be exported as a VRML model, so that it can be viewed in a web browser with a VRML plugin. Unfortunately, viewing the model in a web browser with all point clouds that were extracted by stereo matching is too slow, for real time rendering this model has to be viewed in our point cloud viewer (see fig. 10 for a part of this model).

We see our technique as a successful feasibility study for a mobile high-quality 3D acquisition system. At the moment, the quality of our models is mainly limited by the low resolution interlaced camera. To overcome that restriction we are already building a second generation mobile platform that is equipped with an additional laser scanner and a modern eight megapixel panoramic camera that is capable of providing high dynamic range images. The output of this system will eventually even meet the high quality demands of producing content for 3DTV.

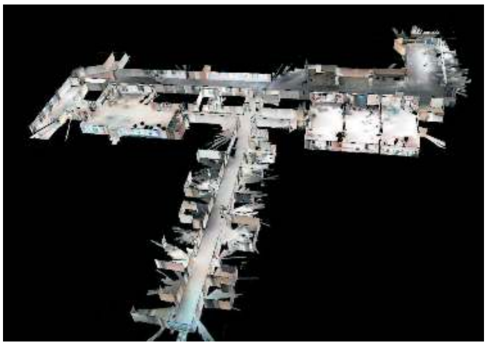

Figure 7: A view of the VRML model - yet without results from stereo matching.

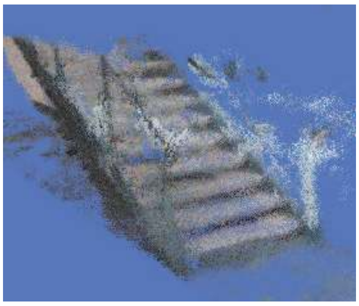

Figure 8: A stairs: output of graph cut-algorithm after removing walls and floor, but before removing outliers manually.

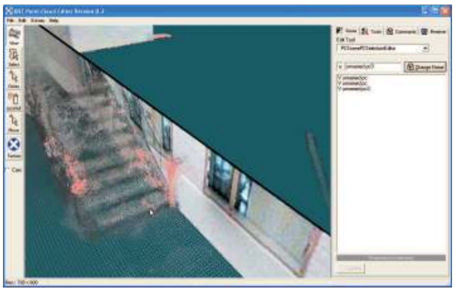

Figure 9: Screen shot of the tool that can be used to edit point clouds comfortably.

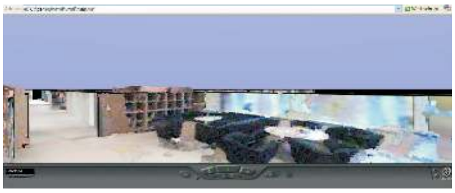

Figure 10: A view of the cafeteria with results from stereo matching included.

REFERENCES

[1] D. Aliaga, D. Yanovsky, and I. Carlbom. Sea of images: A dense sampling approach for rendering large indoor environments. *Computer Graphics & Applications, Special Issue on 3D Reconstruction and Visualization*, pages 22–30, Nov/Dec 2003.

[2] A. Baumberg. Blending images for texturing 3d models. In *Proceedings of the British Machine Vision Conference*, 2002.

[3] P. Biber and W. Straßer. The normal distributions transform: A new approach to laser scan matching. In *International Conference on Intelligent Robots and Systems (IROS)*, 2003.

[4] P. J. Burt and Edward H. Adelson. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics*, 2(4):217–236, 1983.

[5] Paul E. Debevec, Camillo J. Taylor, and Jitendra Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. *SIGGRAPH 96*, 1996.

[6] Udo Frese and Tom Duckett. A multigrid approach for accelerating relaxation-based slam. In *Proc. IJCAI Workshop on Reasoning with Uncertainty in Robotics (RUR 2003)*, 2003.

[7] C. Früh and A. Zakhor. Constructing 3d city models by merging ground-based and airborne views. *Computer Graphics and Applications*, November/December 2003.

[8] J.-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Computational Intelligence in Robotics and Automation, 1999*.

[9] Per-Jonny Käck. Robust stereo correspondence using graph cuts. Master's thesis, School of Computer Science and Engineering, Royal Institute of Technology, Stockholm, 2004.

[10] Vladimir Kolmogorov and Ramin Zabih. Computing visual correspondence with occlusions using graph cuts. In *International Conference on Computer Vision (ICCV'01)*, 2001.

[11] F. Lu and E.E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.

[12] Claudio Rocchini, Paolo Cignomi, Claudio Montani, and Roberto Scopigno. Multiple textures stitching and blending on 3D objects. In *Eurographics Rendering Workshop 1999*, pages 119–130.