

MRS LaserMap: Local Multiresolution Grids for Efficient 3D Laser Mapping and Localization

David Droeschel and Sven Behnke

Autonomous Intelligent Systems, Computer Science, Univ. of Bonn, Germany
{droeschel, behnke}@ais.uni-bonn.de

Abstract. In this paper, we present a three-dimensional mapping system for mobile robots using laser range sensors. Our system provides sensor preprocessing, efficient local mapping for reliable obstacle perception, and allocentric mapping with real-time localization for autonomous navigation. The software is available as open-source ROS-based package and has been successfully employed on different robotic platforms, such as micro aerial vehicles and ground robots in different research projects and robot competitions. Core of our approach are local multiresolution grid maps and an efficient surfel-based registration method to aggregate measurements from consecutive laser scans. By using local multiresolution grid maps as central data structure in our system, we gain computational efficiency by having high resolution in the near vicinity of the robot and lower resolution with increasing distance. Furthermore, local multiresolution grid maps provide a probabilistic representation of the environment—allowing us to address dynamic objects and to distinguish between occupied, free, and unknown areas. Spatial relations between local maps are modeled in a graph-based structure, enabling allocentric mapping and localization.

1 Introduction

In recent years, robotic competitions, such as RoboCup or the DARPA Robotics Challenge¹ (DRC), became increasingly popular. They foster robotics research by benchmarking robotic systems in standardized test scenarios. When integrating robot systems for such competitions, limited resources are one of the main constraints. For instance, recently, Micro Aerial Vehicles (MAVs), such as quadrotors, have gained attention in the robotics community. While being attractive for their wide range of applications, their size and weight limitations pose a challenge realizing cognitive functions—such as autonomous navigation—since onboard memory and computing power are limited. To meet these restrictions, efficient algorithms and data structures are key. We developed our mapping system for the purpose of efficient online mapping and successfully employed it in multiple research projects and robotic competitions. Input to our method are measurements of a continuously rotating 3D laser scanner.

¹ <http://www.theroboticschallenge.org/>

In order to gain both memory and runtime efficiency, we build local multiresolution grid maps with a high resolution close to the sensor and a coarser resolution farther away. Compared to uniform grids, local multiresolution leads to the use of fewer grid cells without losing information and, consequently, results in lower computational costs. Furthermore, it corresponds well to the sensor measurement characteristics in relative distance accuracy and measurement density.

Measurements are aggregated in grid cells and summarized in surface elements (surfels) that are used for registration. Our registration method matches 3D scans on all resolutions concurrently, utilizing the finest common resolution available between both maps.

The system aggregates sparse sensory data from a 3D laser scanner to a dense environment representation used for obstacle avoidance and simultaneous localization and mapping (SLAM) as shown in Fig. 1. While being developed in our previous works [1,2], it has been successfully applied for autonomous outdoor [3] and indoor [4] navigation of MAVs. Furthermore, it has been used on our mobile manipulation robot Momaro during the DRC—supporting operators with a dense environment representation and localization—and during the DLR SpaceBot Camp 2015 to allow for autonomous navigation in rough terrain.

In conjunction with this paper, our software *MRS Laser Map* is published open-source², making it available to other researchers and RoboCup teams in order to facilitate developing robotic applications, contributing to the system, and for comparing and reproducing results. This paper gives an overview of the software architecture to ease integration of the system and presents results from different data sets which we also made publicly available³.

2 Related Work

Different publicly available frameworks for laser-based mapping exist. In the context of RoboCup, mostly 2D approaches are used for mapping and localization. For example, in the RoboCup@Home league, where navigation in planar environments is sufficient, *GMapping* [5] is popular. In contrast, the RoboCup Rescue league necessitates navigation in uneven terrain and approaches such as *Hector Mapping* [6] are used. While a variety of 3D SLAM approaches exist [7,8,9,10,11], maintaining high run-time performance and low memory consumption is an issue.

Hornung et al. [8] implement a multiresolution map based on octrees (OctoMap). Ryde et al. [12] use voxel lists for efficient neighbor queries. Both of these approaches consider mapping in 3D with a voxel being the smallest map element. The 3D-NDT [13] discretizes point clouds in 3D grids and aligns Gaussian statistics within grid cells to perform scan registration.

In contrast to the mentioned approaches, our mapping system is divided into an efficient local mapping—with constant runtime and memory consumption—

² https://github.com/AIS-Bonn/mrs_laser_map

³ http://www.ais.uni-bonn.de/laser_mapping

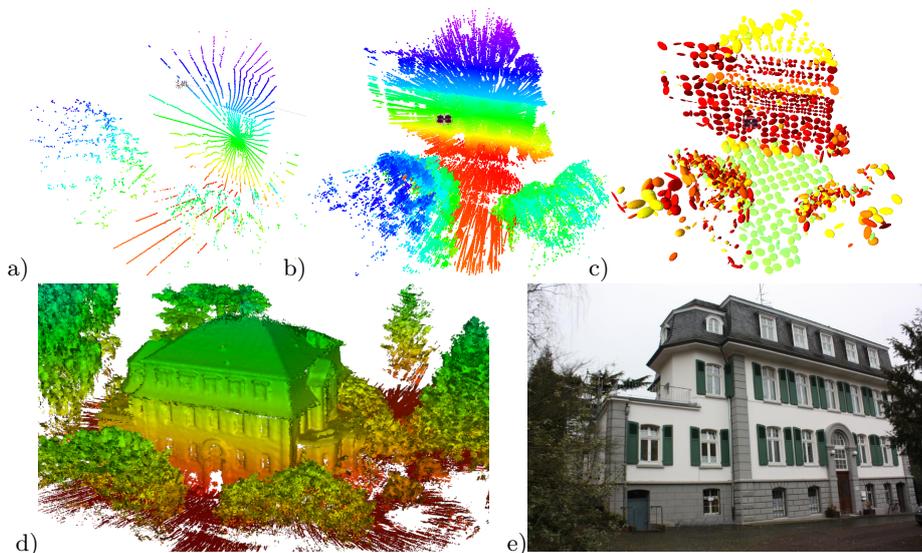


Fig. 1. Results of our mapping system facilitating autonomous navigation of an MAV. Sparse laser scans (a) are aggregated in a local multiresolution grid map (b) by registering surfels (c). The resulting allocentric map (d) of the scene (e).

and a graph-based allocentric mapping module, which allows for online mapping by leveraging local mapping results.

3 Key Components

The key components of our system have been published in our previous work and will therefore be addressed only briefly. The reader is referred to [1] for details on local multiresolution grid maps and the surfel-based registration method and to [2] for the allocentric mapping.

3.1 Local Multiresolution Grid Map

The assembled 3D point clouds are accumulated in a robot-centric grid map with increasing cell sizes from the robot center. It consists of multiple robot-centric 3D grid maps with different resolutions, called levels. Each grid map is embedded in the next level with coarser resolution and double cell length.

Each level is composed of circular buffers holding the individual grid cells. Multiple circular buffers are interlaced to obtain a map with three dimensions. The length of the circular buffers depends on the resolution and the size of the map. In case of a translation of the robot, the circular buffers are shifted whenever necessary to maintain the egocentric property of the map. In case of a translation equal or larger than the cell size, the circular buffers for respective

dimensions are shifted. For sub-cell-length translations, the translational parts are accumulated and shifted if they exceed the length of a cell.

In each grid cell, individual measurements are stored along with occupancy probability and a surface element (surfel). A surfel summarizes its attributed points by their sample mean and covariance. Point measurements of consecutive 3D scans are stored in fixed-sized circular buffers, allowing for point-based data processing and facilitating efficient nearest-neighbor queries. Surfels store statistics about the surface that is mapped. The occupancy probability is used to determine if a cell is free, occupied, or unknown—accounting for uncertainty in the measurements and dynamic objects.

Similar to [8] we use a beam-based inverse sensor model and ray-casting to update the occupancy of a cell. For every measurement in the 3D scan, we update the occupancy information of cells on the ray between the sensor origin and the endpoint with an approximated 3D Bresenham algorithm [14].

3.2 Surfel-based Registration

Aggregating measurements from consecutive time steps necessitates a robust and reliable estimate of the sensor motion. To this end, newly acquired scans are aligned with the so far aggregated map by means of scan registration. We use our surfel-based registration method, which has been designed for this data structure. It leverages the multiresolution property of the map and gains efficiency by summarizing 3D points to surfels that are used for registration. Measurements from the aligned 3D scan replace older measurements in the map and update the occupancy information.

3.3 Simultaneous Localization and Mapping

Modeling the environment in a topological graph structure consisting of nodes, which are connected by edges, allows us to map larger environments and localize the robot in an allocentric frame. Nodes are individual local multiresolution grid maps from different view poses. They are connected by edges, which are spatial constraints from aligning these local maps with each other. Loop-closure is triggered by adding spatial constraints between close-by view poses. We use the g^2o framework [15] to optimize the graph.

4 Software Architecture

Our system is implemented in C++ using ROS and divided in three modules as shown in Fig. 2. Each module is capsuled in a separate ROS nodelet, running in the same nodelet manager—ensuring fast inter-process communication.

4.1 Preprocessing

When using 3D laser scanners that provide a scan from multiple 2D scan lines—for instance by rotating an off-the-shelf 2D laser scanner—the individual 2D

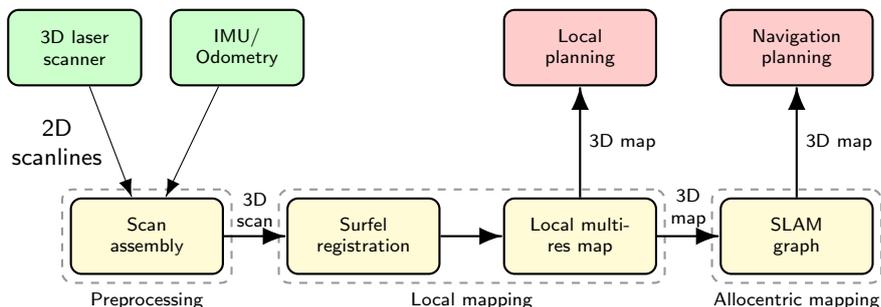


Fig. 2. Overview of our mapping system. The measurements are preprocessed to assemble a 3D scan. The resulting 3D point cloud is used to estimate the transformation between the current scan and the map. Registered scans are stored in a local multi-resolution grid map. Local maps are registered against each other in a SLAM graph.

scans need to be assembled to a 3D scan. For this purpose, we provide the *scan assembler* nodelet. Since the sensor is moving during acquisition, we undistort the individual 2D scans in two steps.

First, measurements of individual 2D scans are undistorted with regards to the rotation of the 2D laser scanner around the sensor rotation axis. Using spherical linear interpolation, the rotation between the acquisition of two scan lines is distributed over the measurements.

Second, the motion of the robot during acquisition of a full 3D scan is compensated. To estimate robot motion, different sensors can be used, e.g., wheel odometry, visual odometry, or measurements from an inertial measurement unit (IMU). The interface for motion sensors is generalized to the ROS transform library and can be parametrized.

4.2 Local Mapping

The local mapping module is the central component of the system. It aggregates consecutive 3D scans in a local multi-resolution grid map by aligning them with our efficient surfel-based registration method.

Standard ROS point cloud messages are expected as input to the module and are published as output for the dense representation of aggregated scans. Furthermore, the map is published in a custom ROS message to the allocentric mapping module. To provide local planners with information about occupied, free, and unknown areas, the grid structure is published as well. Finally, the estimated motion between consecutive 3D scans, i.e., the recovered transformation from the registration method, is published in a ROS transform to correct odometry drift relative to the local map.

During mission, operators need the ability to reconfigure the module. Thus, service calls and parameters are provided to reset the map, change occupancy mapping parameters, or delete specific points. The resolution, size of the map,

number of levels, and the length of the circular buffers storing the measurements can be parametrized to the specific application.

The local mapping module aggregates measurements in a local, robot-centric frame, i.e., measurements vanish when being farther away from the robot than the size of the local map. This results in a constant upper-bound for memory consumption, independent of the size and the structure of the environment. When developing continuously running robotics applications, this is an important property.

4.3 Allocentric Mapping

The local mapping module can be used in combination with the allocentric mapping module to map larger environments and track the robot pose in an allocentric frame. The module stores copies of local maps in a graph structure as nodes, connected by edges. After every scan update of the local mapping module, the current local map is registered to the graph by aligning it to a reference node. The reference node is a pointer to the last tracked local map and is updated if other nodes are closer to the current pose or a new node is added. A new node and the corresponding edge to the reference node is added to the graph if the robot moved sufficiently far. Edges are also added between close-by nodes that are not in temporal sequence. After adding an edge, graph optimization is triggered.

To visualize the allocentric map over a limited network connection, we implemented a visualization nodelet that is able to run off-board on a different computer. Data transmission between the allocentric mapping nodelet and the visualizer is compressed and incremental—only transmitting newly added nodes. When no node was added, optimized edges and the current tracked pose are transmitted, allowing to update the off-board graph structure.

Similar to the local mapping module, this module also provides the ability to reconfigure. Graph structures can be reset and parameters controlling when new nodes are added can be set.

5 Evaluation

Our system has been evaluated on different MAV platforms, e.g., for autonomous outdoor navigation and mapping as shown in Fig. 1. Furthermore, it has been deployed on our mobile manipulation robot Momaro during the DLR SpaceBot Camp 2015 and the DARPA Robotics Challenge as shown in Fig. 3. In our previous work [16] we evaluated the runtime and registration accuracy of the local mapping and registration system. The evaluation shows that our method is considerably more accurate and efficient than state-of-art methods.

6 Conclusion

In this paper, we present MRSLaserMap, an efficient system for laser-based online mapping and localization. The implementation is publicly available as

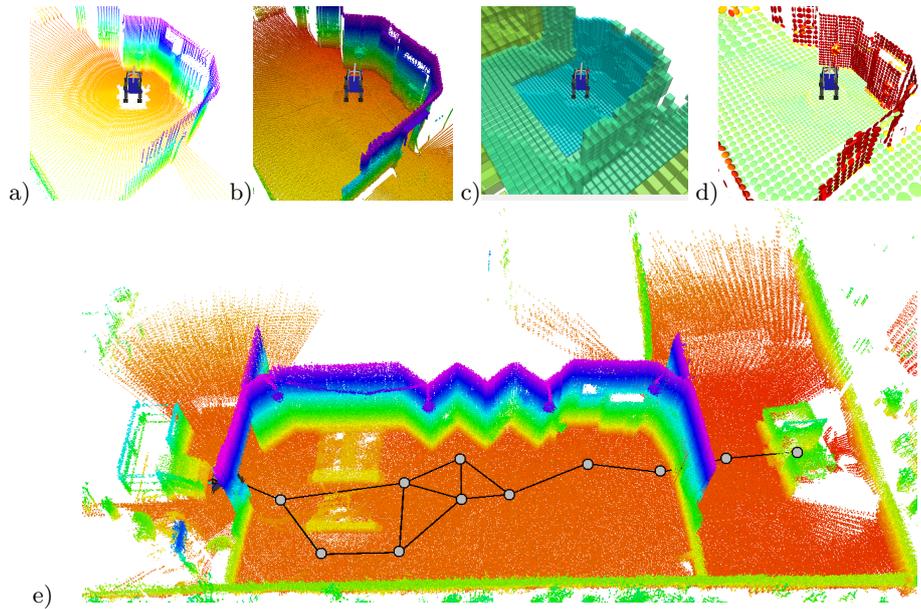


Fig. 3. Results of our mapping system on our mobile manipulation robot Momaro. The data has been acquired during the DRC. Sparse laser scans (a) are aggregated in a local multiresolution grid map (b+c) by registering surfels (d). The resulting allocentric map (e) shows graph structure, consisting of nodes (gray discs) and edges (black lines).

open-source project to facilitate research and the building of 3D robot navigation functionalities for other groups and RoboCup Teams.

Acknowledgments This work has been supported by the European Union’s Horizon 2020 Programme under Grant Agreement 644839 (CENTAURO) and grant BE 2556/7-2 of German Research Foundation (DFG).

References

1. David Droeschel, Jörg Stückler, and Sven Behnke. Local multi-resolution representation for 6D motion estimation and mapping with a continuously rotating 3D laser scanner. In *Robotics and Automation (ICRA), IEEE International Conference on*, 2014.
2. David Droeschel, Jörg Stückler, and Sven Behnke. Local multi-resolution surfel grids for mav motion estimation and 3d mapping. In *Proc. of the Int. Conf. on Intelligent Autonomous Systems*, 2014.
3. David Droeschel, Matthias Nieuwenhuisen, Marius Beul, Dirk Holz, Jörg Stückler, and Sven Behnke. Multilayered mapping and navigation for autonomous micro aerial vehicles. *Journal of Field Robotics (JFR)*, 33(4):451–475, 2016.

4. Matthias Nieuwenhuisen, David Droeschel, Marius Beul, and Sven Behnke. Autonomous navigation for micro aerial vehicles in complex gnss-denied environments. *Journal of Intelligent & Robotic Systems*, pages 1–18, 2015.
5. G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, Feb 2007.
6. S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf. A flexible and scalable slam system with full 3d motion estimation. In *Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on*, pages 155–160, Nov 2011.
7. Andreas Nüchter. *3D robotic mapping: the simultaneous localization and mapping problem with six degrees of freedom*. Springer, 2008.
8. Armin Hornung, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: an efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 34:189–206, 2013.
9. J. Elseberg, D. Borrmann, and A. Nüchter. 6DOF semi-rigid SLAM for mobile scanning. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 2012.
10. Will Maddern, Alastair Harrison, and Paul Newman. Lost in translation (and rotation): Fast extrinsic calibration for 2D and 3D LIDARs. In *Robotics and Automation (ICRA), IEEE International Conference on*, May 2012.
11. Sean Anderson and Timothy D. Barfoot. Towards relative continuous-time SLAM. In *Robotics and Automation (ICRA), IEEE International Conference on*, 2013.
12. Julian Ryde and Huosheng Hu. 3D mapping with multi-resolution occupied voxel lists. *Autonomous Robots*, 28:169 – 185, 2010.
13. Todor Stoyanov, Martin Magnusson, Henrik Andreasson, and Achim J Lilienthal. Fast and accurate scan registration through minimization of the distance between compact 3D NDT representations. *The International Journal of Robotics Research*, 31(12):1377–1393, 2012.
14. John Amanatides and Andrew Woo. A fast voxel traversal algorithm for ray tracing. In *In Eurographics '87*, pages 3–10, 1987.
15. R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. G2o: A general framework for graph optimization. In *Robotics and Automation (ICRA), IEEE International Conference on*, 2011.
16. Jan Razlaw, David Droeschel, Dirk Holz, and Sven Behnke. Evaluation of registration methods for sparse 3d laser scans. In *Mobile Robots (ECMR), 2015 European Conference on*. IEEE, 2015.