Journal of Intelligent and Robotic Systems, Volume 84, Issue 1–4, pp 199–216, Springer, 2016.

Journal of Intelligent and Robotic Systems manuscript No.
(will be inserted by the editor)

Autonomous Navigation for Micro Aerial Vehicles in Complex GNSS-denied Environments

Matthias Nieuwenhuisen · David Droeschel · Marius Beul · Sven Behnke

Received: date / Accepted: date

Abstract Micro aerial vehicles, such as multirotors, are particular well suited for the autonomous monitoring, inspection, and surveillance of buildings, e.g., for maintenance in industrial plants. Key prerequisites for the fully autonomous operation of micro aerial vehicles in restricted environments are 3D mapping, real-time pose tracking, obstacle detection, and planning of collision-free trajectories. In this article, we propose a complete navigation system with a multimodal sensor setup for omnidirectional environment perception. Measurements of a 3D laser scanner are aggregated in egocentric local multiresolution grid maps. Local maps are registered and merged to allocentric maps in which the MAV localizes. For autonomous navigation, we generate trajectories in a multi-layered approach: from mission planning over global and local trajectory planning to reactive obstacle avoidance. We evaluate our approach in a GNSS-denied indoor environment where multiple collision hazards require reliable omnidirectional perception and quick navigation reactions.

Keywords GNSS-denied localization · muliresolutional mapping · obstacle detection · multi-layered planning · trajectory generation · state estimation

1 Introduction

Micro aerial vehicles (MAVs) are enjoying increasing popularity. Due to their low cost and flexibility, they are used for aerial photography, inspection, surveillance, and search and rescue (SAR) missions. In most cases, a human operator pilots the MAV remotely to fulfill a specific task or the MAV is following a predefined path of GNSS waypoints in an obstacle-free altitude.

MAVs allow to quickly visit otherwise inaccessible volumes, but permanent line of sight to the MAV may not be maintainable. Also, passages may be narrow and surrounding environmental structures may be hard to perceive for a human operator. Hence, remotely controlling an MAV in complex 3D environments is much

Autonomous Intelligent Systems Group
University of Bonn
Bonn, Germany
{nieuwenh, droeschel, mbeul, behnke}@ais.uni-bonn.de

Fig. 1: Our MAV is equipped with eight co-axial rotors and a plurality of sensors, including a continuously rotating 3D laser scanner, two stereo camera pairs and eight ultrasonic sensors.

more demanding than controlling a ground vehicle. Narrow passages and fully closed rooms also prevent the use of global navigation satellite systems (GNSS) like GPS or GLONASS such that GNSS-based hovering or waypoint following is not an option.

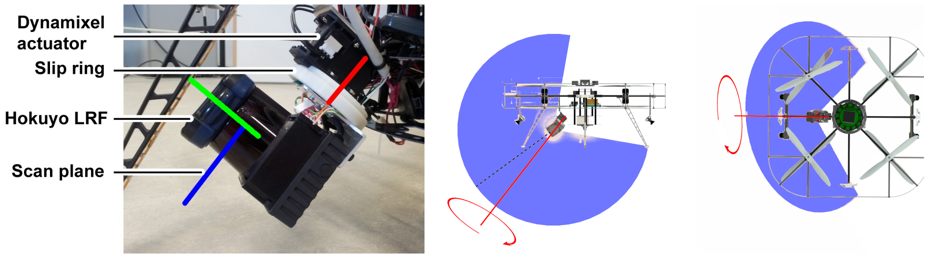
In order to safely navigate in such environments, an alternative is to make MAVs autonomous, such that they can on their own—without interaction with the operator—solve well-defined sub-tasks. For example, the operator may specify a set of regions within a building and the MAV autonomously approaches all regions and collects sensor information.

For the autonomous operation of MAVs, key prerequisites are localization in unmodified GNSS-denied environments, real-time obstacle detection, and planning of collision-free trajectories. Fundamental aspects in the implementation of such a system are robustness—all obstacles need to be reliably detected and mapped while avoiding false positives—and real-time onboard processing.

In this article, we present a complete integrated system consisting of an MAV with a multimodal omnidirectional sensor setup (see Fig. 1), a laser-based 3D mapping and 6D localization, and a multilayered navigation approach, tailored to the special needs of MAVs. Each layer uses and builds its own environment representation: allocentric maps for global path and mission planning and egocentric obstacle maps for local trajectory planning and reactive collision avoidance.

We employ Simultaneous Localization And Mapping (SLAM) to build initial allocentric maps of the environment, used for navigation and localization. This prior knowledge aids our mission planning—in contrast to fully autonomous exploration of unknown space. Using this map also enables us to define the regions, the MAV is supposed to reach with respect to the environment.

The initial 3D map does not include dynamic obstacles, which constitute a collision hazard for the MAV. Also, previously acquired maps can be outdated. Especially in dynamic environments like industrial plants, machines and tools are moved, lamps and cables are hanging from the ceiling and transportation equipment changes its position. Thus, the initial mission plans need to be adjusted on the fly, whenever more information becomes available during flight.



(a) (b) (c)

Fig. 2: 3D scanner setup: (a) The Hokuyo 2D LRF is mounted on a bearing and rotated around the red axis. Its mirror is rotated around the green axis, resulting in a 2D measurement plane (blue). (b + c) CAD drawings illustrating the FoV of individual scans of the laser scanner (blue) from side and top view. The black dashed line illustrates the center of the measurement plane. The 2D LRF is rotated around the red axis.

Designing sensory systems and perception algorithms is challenging for MAVs due to their size and weight constraints and limited computing power. In order to enable autonomous navigation in complex 3D environments, we developed a small and lightweight continuously rotating 3D laser scanner that measures distances of up to 30 m in almost all directions. It consists of a Hokuyo UTM-30LX-EW 2D laser range finder (LRF), which is rotated by a servo actuator to gain a 3D field of view (FoV), as shown in Fig. 2. Additionally, our MAV is equipped with two stereo camera pairs and ultrasonic sensors, covering the volume around the MAV up to 6 m range (Holz et al., 2013).

All these sensors have only relative precision. Their measurements are less dense and less precise with increasing distance. This is reflected in the local multiresolution property of our MAV-centric obstacle map. We employ 3D local multiresolution path planning. This techniques allows for efficient map updates and frequent replanning, which makes 3D navigation in dynamic, unpredictable environments possible.

This article extends our previous work on autonomous outdoor navigation (Droeschel et al., 2015). Our main contributions here are I) robust 6D indoor localization by means of fast multilevel surfel registration and II) global navigation with a cost function, tailored to the requirements of our localization. We integrated our new components together with our fast egocentric obstacle perception and avoidance into an MAV system capable of executing fully autonomous missions in complex indoor environments. The robustness is evaluated in multiple autonomous indoor flight experiments.

2 Related Work

The use of MAVs in indoor applications is an active research topic with increasing popularity. Multiple groups use MAVs to obtain a quick overview of an area and to guide unmanned ground vehicles (UGV). Michael et al. (2012) carry an AscTec Pelican MAV on a UGV to map and inspect hardly accessible areas in a building after an earthquake. Similar to our work, their MAV can operate in autonomous and semi-autonomous mode, depending on the mission requirements. However, in contrast to our work they focus on a computationally constrained MAV and omit full 3D navigation—restricting it to 2.5D environments, a restriction we do not make. Luo et al. (2011) show that the guidance of UGVs by MAVs can reduce the time needed to fulfill missions.

Particularly important for fully autonomous operation is the ability to perceive obstacles and to avoid collisions. Obstacle avoidance is often neglected, e.g., by flying in a sufficient height when autonomously flying between waypoints.

Due to the limited payload of MAVs, most approaches to obstacle avoidance are camera-based (Mori and Scherer, 2013; Ross et al., 2013; Schmid et al., 2014; Magree et al., 2014; Tripathi et al., 2014; Flores et al., 2014; Schauwecker and Zell, 2014; Park and Kim, 2014). Nolan et al. (2013) employ monocular depth estimation from MAV motion to perceive obstacles. In order to estimate depth of object points instantaneously, stereo camera pairs are used on MAVs.

A particularly well-suited approach for indoor environments is to use RGB-D cameras that can measure depth even on textureless surfaces by projecting an infrared pattern (Bachrach et al., 2012; Flores et al., 2014). The limited field of view of cameras poses a problem when flying in constrained spaces where close obstacles can surround the MAV. To overcome these limitations, some MAVs are equipped with multiple cameras.

For mobile ground robots, 3D laser scanning sensors are widely used due to their accurate distance measurements even in bad lighting conditions and their large FoV. For instance, autonomous cars often perceive obstacles by means of a rotating laser scanner with a 360° horizontal FoV, allowing for detection of obstacles in every direction (Montemerlo et al., 2008). Up to now, such 3D laser scanners are rarely used on lightweight MAVs—due to payload limitations. Instead, two-dimensional LRFs (Tomić et al., 2012; Grzonka et al., 2009; Bachrach et al., 2009; Shen et al., 2011; Grzonka et al., 2012; Huh et al., 2013) are used. Using a statically mounted 2D LRF restricts the FoV to the two-dimensional measurement plane of the sensor. This poses a problem especially for reliably perceiving obstacles surrounding the MAV. When moving, however, and in combination with accurate pose estimation, these sensors can very well be used to build 3D representations of the measured surfaces. Fossel et al. (2013), for example, use Hector SLAM (Kohlbrecher et al., 2011) for registering horizontal 2D laser scans and OctoMap (Hornung et al., 2013) to build a three-dimensional occupancy model of the environment at the measured heights. Morris et al. (2010) follow a similar approach and in addition use visual features to aid motion and pose estimation. Still, perceived information about environmental structures is constrained to lie on the 2D measurement planes of the moved scanner. In contrast, we use a continuously rotating LRF that does not only allow for capturing 3D measurements without moving, but also provides omnidirectional obstacle perception at comparably high frame rates. An MAV with a similar sensor is used by Cover et al. (2013) to autonomously explore rivers using visual localization and laser-based 3D obstacle perception. In contrast to their work, we use the 3D laser scanner for both omnidirectional obstacle perception and mapping the environment in 3D.

Similar to our work, Israelsen et al. (2014) present an approach to local collision avoidance that works without global localization and can aid a human operator to navigate safely in the vicinity of obstacles. Our work extends the safety layer by a deliberative planning layer based on local maps and navigation targets.

Heng et al. (2014) use a multiresolution grid map to represent the surroundings of a quadrotor. A feasible plan is generated with a vector field histogram. Schmid et al. (2014) autonomously navigate to user-specified waypoints in a mine. The map used for planning is created by an onboard stereo camera system. By using rapidly exploring random belief trees (RRBT), Achtelik et al. (2014) plan paths that do not only avoid obstacles, but also minimize the variability of the state estimation. Recent search-based methods for obstacle-free navigation include work of MacAllister et al. (2013). They use A* search to find a feasible path in a four-dimensional grid map incorporating the asymmetric shape of the MAV. Cover et al. (2013) also use a search-based method. These methods assume complete knowledge of the scene geometry—an assumption that we do not make here.

Creating 3D environment representations through SLAM has been investigated first with mobile ground robots (Nuechter et al., 2005; Magnusson et al., 2007). A research topic in SLAM with 3D laser scanners is how to maintain high run-time performance and low memory consumption simultaneously. While many methods assume the robot to stand still during 3D scan acquisition, some approaches also integrate scan lines of a continuously rotating laser scanner into 3D maps while the robot is moving (Bosse and Zlot, 2009; Elseberg et al., 2012; Stoyanov and Lilienthal, 2009; Maddern et al., 2012; Anderson and Barfoot, 2013).

Takahashi et al. (2008) also build environment maps with a moving 3D laser scanner. They localize the robot using GPS and IMU sensors. Thrun et al. (2003) realized a 3D mapping system with a rigidly mounted 2D laser scanner on a helicopter. The laser scanner measures in a vertical plane perpendicular to the flight direction. In order to localize the helicopter, measurements from GPS and IMU are fused and consecutive 2D scans are registered, assuming scan consistency in flight direction. In our approach, we do not make such an assumption on scan consistency. Shen et al. (2014) estimate the MAV state in combined indoor and outdoor flights by fusing different sensor modalities. A 2D laser scanner is used for indoor localization. In contrast, our laser-based localization can estimate the complete 6D pose relative to a map coordinate frame.

Many approaches consider mapping in 3D with a voxel being the smallest map element, e.g., Ryde and Hu (2010) who use voxel lists for efficient neighbor queries. Similar to our approach, the 3D-NDT (Magnusson et al., 2007) represents point clouds as Gaussian distributions in voxels at multiple resolutions. Our local multiresolution surfel grids adapt the maximum resolution with distance to the sensor to match the measurement characteristics. Moreover, our registration method aligns 3D scans on all resolutions concurrently.

3 System Setup and Overview

Our MAV is an octorotor with a co-axial arrangement of rotors (see Fig. 1). This yields a compact flying platform that is able to carry a plurality of sensors and a fast computer (Intel Core i7-3820QM 2.7 GHz). For sensor data processing and navigation planning, we employ the Robot Operating System (ROS) by Quigley et al. (2009) as middleware. For low-level velocity and attitude control, the MAV is equipped with a PIXHAWK Autopilot flight control unit (Meier et al., 2012). To allow for safe

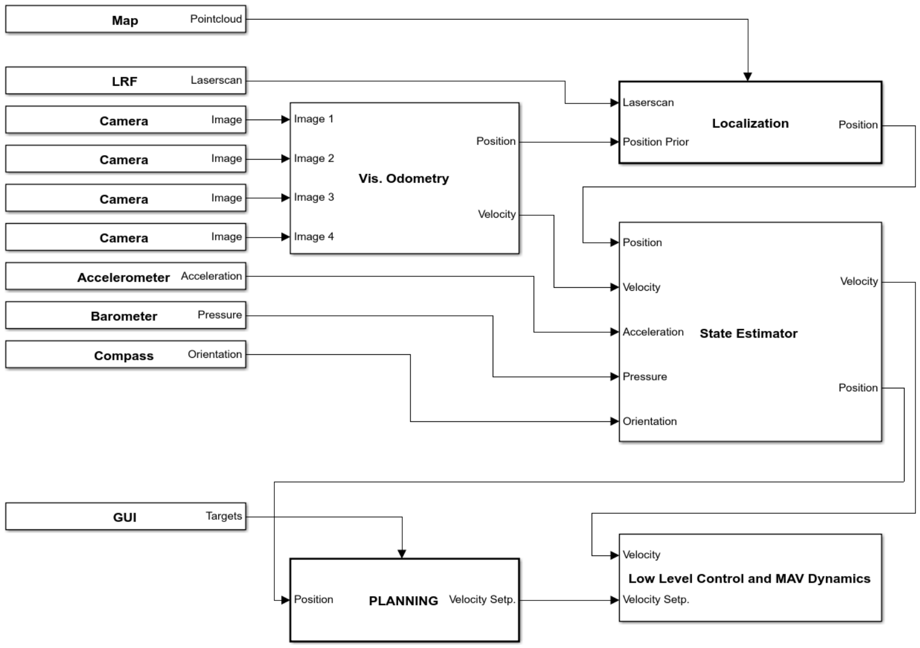


Fig. 3: Sensor measurements and simplified processing pipeline.

omnidirectional operation in challenging environments, our MAV is equipped with a multimodal sensor setup:

- Main sensor for obstacle perception is a continuously rotating 3D laser scanner. Only a small conical volume on the upper rear is occluded by the MAV core.
- Two monochrome stereo camera pairs pointing in forward and backward direction are used for visual odometry and obstacle perception. Equipped with fish-eye lenses, they cover a large area around the MAV.
- Eight ultrasonic sensors measure distances in all directions around the MAV. Despite their limited accuracy and range of 6 m, they detect transparent obstacles, like windows.

The stream of sensor data and the interaction between the subsystems is shown in Fig. 3.

To allow for direct reactions on obstacle perceptions on one end and consistent mapping and complex planning on the other end, our system architecture is layer-based (see Fig. 4) with slower global layers on the top (deliberative planning, allocentric mapping) and faster local layers on the bottom (reactive obstacle avoidance, egocentric obstacle maps). From top to bottom the abstraction level of planning and mapping is reduced and the processing frequency approaches the sensors measurement frequency.

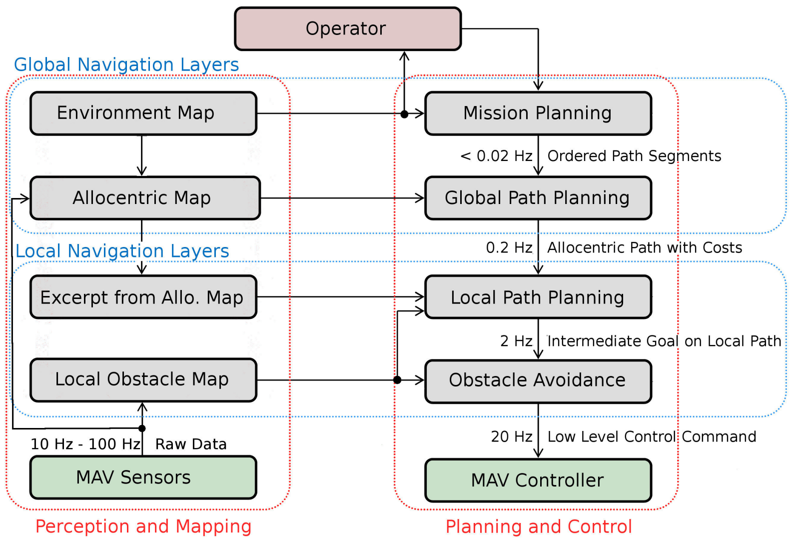


Fig. 4: Multi-layered navigation approach: slow planners on the top yield coarse trajectories which are refined on faster lower layers. The layers build and use different environment representations: from static allocentric maps to local ego-centric obstacle maps updated at the sensor rate.

4 Local Perception

We construct an MAV-centric multiresolution grid map that is used to accumulate sensor measurements. We first register newly acquired 3D scans with the so far accumulated map and then update the map with the registered 3D scan. The map is utilized by our path planning and obstacle avoidance algorithms described in subsequent sections.

4.1 3D Scan Assembly

When assembling 3D scans from raw laser scans we account for the rotation of the scanner w.r.t. the MAV and for the motion of the MAV during acquisition. Thus, scan assembling mainly consists of two steps.

First, measurements of individual scan lines are undistorted with regards to the rotation of the 2D LRF around the servo rotation axis (red axis in Fig. 2). Here, the rotation between the acquisition of two scan lines is distributed over the measurements by using spherical linear interpolation.

Second, we compensate for the motion of the MAV during acquisition of a full 3D scan. To this end, we incorporate a visual odometry estimate from the two stereo cameras. Here, a keyframe-based bundle adjustment is performed (Schneider et al., 2013) on the synchronized images with 18 Hz update rate. Since the update rate of the 2D LRF is 40 Hz, we linearly interpolate between the estimates of the visual odometry. The 6D motion estimate is used to assemble the individual 2D scan lines of each half rotation to a 3D scan.

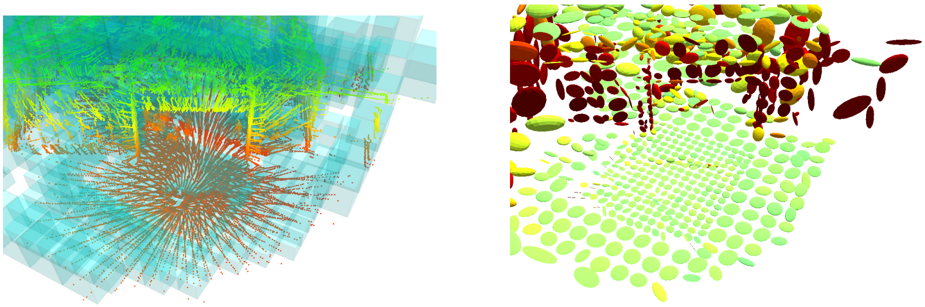


Fig. 5: Left: Grid-based local multiresolution map with a higher resolution in proximity to the sensor and a lower resolution with increasing distance. Color encodes height. Right: The surfel representation of the map (colored by surfel orientation).

4.2 Local Multiresolution Map

We use a hybrid local multiresolution map that represents both occupancy information and the individual distance measurements. The most recent measurements are stored in ring buffers within grid cells that increase in size with distance from the robot center. Thus, we use a high resolution in the close proximity to the sensor and a lower resolution far away from our robot, which correlates with the sensor characteristics in measurement accuracy and density. Compared to uniform grid-based maps, multiresolution leads to the use of fewer grid cells, without losing relevant information and consequently results in lower computational costs. Fig. 5 shows an example of our local multiresolution grid-based map.

We aim for efficient map management for translation and rotation. To this end, individual grid cells are stored in a ring buffer to allow shifting of elements in constant time. We interlace multiple ring buffers to obtain a map with three dimensions. The length of the ring buffers depends on the resolution and the size of the map. In case of a translation of the MAV, the ring buffers are shifted whenever necessary to maintain the egocentric property of the map. For sub-cell-length translations, the translational parts are accumulated and shifted if they exceed the length of a cell.

4.3 Registration Approach

We register each newly acquired 3D scan with the local multiresolution map of the environment with the surfel-based method of Droeschel et al. (2014a). Instead of considering each point individually, we represent the 3D scan as local multiresolution grid and match surfels. In each cell of the grid, we maintain one surfel that summarizes the individual 3D points that lie within the cell (cf. Fig. 5). A surfel is defined by the sample mean and the sample covariance of these points. We align a newly acquired scan (scene) and the local multiresolution map (model) by finding a rigid 6 degree-of-freedom (DoF) transformation $T(\theta)$ that best aligns the scene surfels to the model surfels. By summarizing measurements in surfels, and therefore considering several orders of magnitudes less elements for registration, we gain efficiency. When matching surfels we choose the finest common resolution available

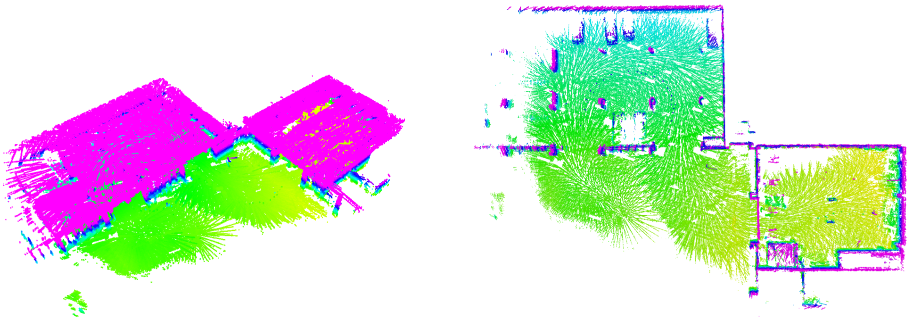


Fig. 6: Allocentric map from a combined indoor/outdoor flight after pose graph optimization. Left: Perspective view. Right: Top-view with removed ceiling.

between both maps to achieve accuracy. Compared to dense RGB-D images (Stückler and Behnke, 2014) or high-resolution static 3D laser scans used in our previous work (Schadler et al., 2013), 3D scans obtained from our LRF are much sparser. We cope with this sparsity through probabilistic assignments of surfels during the registration process.

5 Allocentric Mapping and Localization

For fast estimation of the MAV motion, we incorporate IMU and visual odometry measurements into velocity and pose estimates. While these estimates allow us to control the MAV and to track its pose over a short period of time, they are prone to drift and thus are not suitable for localization on the time scale of a mission. Furthermore, they do not provide a fixed allocentric frame for the definition of mission-relevant poses independent from the MAV. Thus, we build an allocentric map by means of laser-based SLAM before mission execution and employ laser-based pose tracking w.r.t. this map during autonomous operation.

5.1 Mapping

This allocentric map is build by aligning multiple local multiresolution maps, acquired from different view poses (Droeschel et al., 2014b). We model the different view poses as nodes in a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that are connected by edges. A node consists of the local multiresolution map from the corresponding view pose. Each edge in the graph models a spatial constraint between two nodes.

After adding a new 3D scan to the local multiresolution map as described in Sec. 4.3, the local map is registered towards the previous node in the graph using the multiresolution surfel registration with probabilistic assignments (Droeschel et al., 2014a). A new node is generated for the current local map, if the MAV moved sufficiently far. The registration result x_i^j between a new node v_i and the previous node v_j is a spatial constraint that we maintain as values of edges $e_{ij} \in \mathcal{E}$. In addition to edges between the previous node and the current node, we add spatial constraints between close-by view poses that are not in temporal sequence.

On each scan update, we check for one new constraint between the current reference v_{ref} and other nodes v_{cmp}. We determine a probability

$$p_{\mathrm{chk}}(v_{\mathrm{cmp}}) = \mathcal{N}\left(d(x_{\mathrm{ref}}, x_{\mathrm{cmp}}); 0, \sigma_d^2\right) \tag{1}$$

that depends on the linear distance $d(x_{\mathrm{ref}}, x_{\mathrm{cmp}})$ between the view poses x_{ref} and x_{cmp}. According to $p_{\mathrm{chk}}(v)$, we choose a node v from the graph and determine a spatial constraint between the nodes using our surfel registration method.

From the graph of spatial constraints, we infer the probability of the trajectory estimate given all relative pose observations

$$p(\mathcal{V} \mid \mathcal{E}) \propto \prod_{e_{ij} \in \mathcal{E}} p(x_i^j \mid x_i, x_j). \tag{2}$$

Each spatial constraint is a normally distributed estimate with mean and covariance determined by our probabilistic registration method. This pose graph optimization is efficiently solved using the g^{2}o framework by Kuemmerle et al. (2011), yielding maximum likelihood estimates of the view poses x_i.

After the MAV has traversed the environment, the allocentric map is build from the optimized pose graph by merging all local surfel maps. Here, we use surfels with uniform resolution. Fig. 6 shows an example map acquired from a combined indoor and outdoor flight.

5.2 Pose Tracking

When executing a mission, the MAV traverses a set of goal poses w.r.t. the coordinate frame defined by our allocentric map. Since the laser scanner acquires complete 3D scans with a relatively low frame rate, we incorporate the egomotion estimate from the visual odometry and measurements from the IMU to track the pose of the MAV. The egomotion estimate is used as a prior for the motion between two consecutive 3D scans. In detail, we track the pose hypothesis by alternating the prediction of the MAV movement given the filter result and alignment of the current local multiresolution map towards the allocentric map of the environment. To align the current local map with the allocentric map, we also use the surfel-based registration (Droeschel et al., 2014a). The allocentric localization is triggered after acquiring a 3D scan and adding it to the local multiresolution map. We update the allocentric robot pose with the resulting registration transform. To achieve real-time perfomance of the localization module, we track only one pose hypothesis. We assume that the initial pose of the MAV is known, either by starting from a predefined pose or by means of manually setting the pose. Fig. 7 shows the registration of a 3D scan to the map and an estimated 6D trajectory.

The resulting robot pose estimate from the allocentric localization is used as a measurement update in a lower-level state estimation filter. We propagate this allocentric pose over time with visual odometry and IMU to obtain allocentrically consistent pose and velocity estimates at a sufficiently high rate for planning and control.

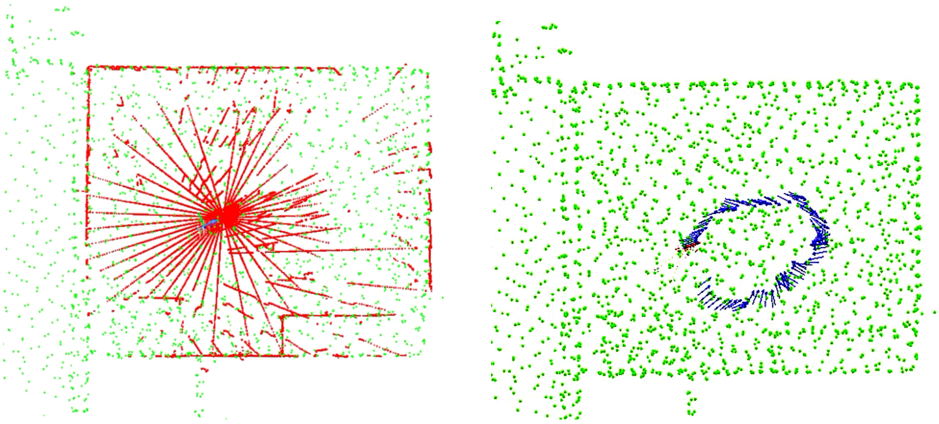


Fig. 7: Undistorted local 3D laser scans (red) are matched to the allocentric SLAM map (green) to localize the robot in the allocentric frame. Here, a downsampled version of the allocentric map is shown. Left: The estimated robot pose is depicted by the axes. Right: Blue arrows depict the estimated MAV trajectory.

6 Navigation Planning

To operate MAVs in indoor environments for inspection missions, safe navigation in the vicinity of obstacles is key. Compared to outdoor missions, the free space is restricted and keeping a large safety margin to obstacles is not an option. Hence, only quick reactions given the observed vehicle- and environment state ensure the successful and safe mission completion. To allow for deliberative planning and quick reactions, we employ hierarchical navigation: from mission planning to low-level motion control. These tasks require different abstractions of the environment, as illustrated in Fig. 4.

To plan an inspection mission, we need a coarse model of the environment; to plan collision-free paths, we need a finer and up-to-date consistent geometric model; and to avoid collisions, we need a non-aggregated local representation of the close vicinity of the MAV. The planned actions also have different granularity, which is represented by the planning frequency, from once per mission to multiple times per second. The higher-layer planners set goals for the lower-level planners which produce more concrete action sequences based on more local and up-to date environment representations.

6.1 Mission Planning

The topmost layers are a mission planner and a global path planner. Both use a similar representation of the environment that is built by means of SLAM by the MAV beforehand, as described in the previous section, converted to an OctoMap (Hornung et al., 2013).

Input to the mission planner is a set of view poses defined by the user. The mission planner employs a global path planner on a coarse uniform grid map to determine the approximate costs between every pair of mission goals. In order to

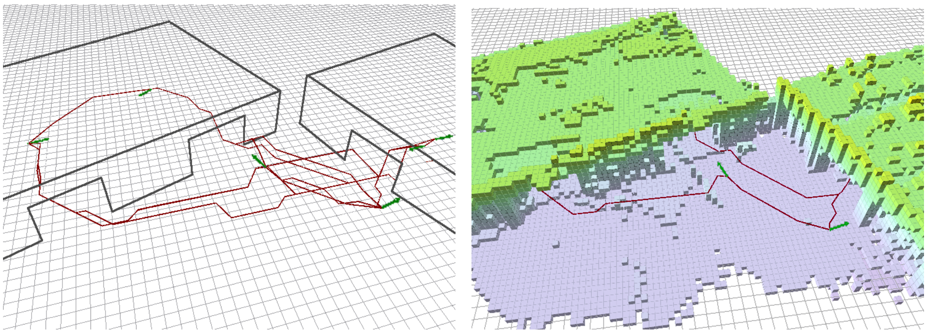


Fig. 8: On the top layer, a mission planner evaluates the best execution order of mission poses (green arrows). Red lines depict planned cost-optimal paths between mission poses. Left: Planned paths between each pair of mission relevant poses. Right: Cost optimal flight path that yields an cost-optimal order of mission poses from the current MAV pose. Color encodes height.

speed up the process, we reuse computation results, e.g., the obstacle costs per grid cell stay the same for every combination of view poses. Furthermore, the costs of reaching grid cells from one start pose stay constant in this offline processing step. After calculating all pair-wise edge weights, the cost-optimal sequence of view poses is determined by means of Concorde (Applegate et al., 2006), a fast solver for the traveling salesman problem (TSP). Please note, that the instances of the TSP for one mission are sufficiently small, so that exact solutions are tractable.

The result of mission planning is a flight plan composed of an ordered list of 4D waypoints (x, y, z, yaw). Fig. 8 shows an example solution for a mission to inspect parts of a hall and a garage.

6.2 Global Path Planning

The next layer in the planning hierarchy is a global path planner. This layer plans globally consistent plans, based on I) the (updated) environment model, discretized to grid cells with 0.5 m edge length, II) the current pose estimate of the MAV, and III) the next mission waypoint, including 3D position, yaw orientation, and required accuracies. Planning frequency is 0.2 Hz and we use the A* algorithm to find cost-optimal paths.

In our application domain, most obstacles not represented in the allocentric map can be surrounded locally, without the need for global replanning. Hence, it is sufficient to replan globally on a more long-term time scale to keep the local deviations of the planner synchronized to the global plan and to avoid the MAV to get stuck in a local minimum that the local planner cannot solve due to its restricted view of the environment.

As via-points that are not mission critical can be blocked by locally perceived obstacles, it is not sufficient to send the next waypoint of the global path to the local planning layers. Instead, the input to the local planner is the complete global plan. The global path is cost-optimal with respect to the allocentric map. Hence, the path costs of the global path are a lower bound to path costs for refined plans, based on

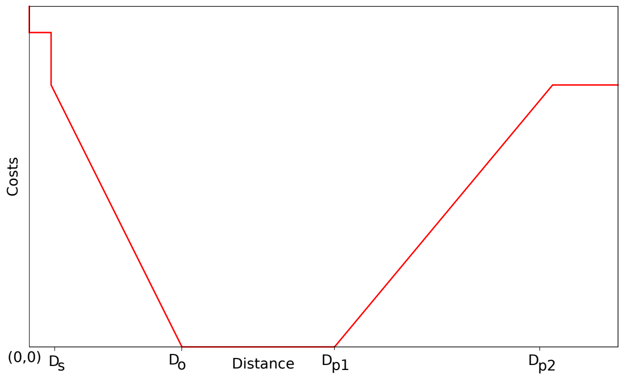


Fig. 9: We model the traversal costs for a grid cell in our global planners as a piecewise linear function of the distance to the next obstacle. With increasing distance we have fixed maximum costs in the direct vicinity of obstacles, linear decreasing costs to fly farther away from obstacles where possible, and a range with zero obstacle costs. As our laser-based localization needs structure, the traversal costs increase linear again, when flying too far away from obstacles.

newly acquired sensor information—mostly dynamic and static previously unknown obstacles—and a local path deviating from the global plan cannot be shorter in terms of path costs. Locally shorter plans on lower layers with a local view on the map may yield globally suboptimal paths. Also, mission goals are marked as the local planner has to reach these exactly. If this is not possible, the mission planning has to resolve this failure condition.

For application-specific SLAM maps, it is often not necessary to cover the whole reachable environment, but only the parts that are relevant for the mission execution. In particular, they cannot cover the complete space outside the buildings. Our laser-based localization needs to perceive sufficient structure to work robustly. Hence, the MAV should not fly in completely unmapped or free space, e.g., at a height where the ground is no longer observed by the LRF. To ensure robust localization even in partial maps and unbound environments, we employ an approach inspired by coastal navigation (Roy et al., 1999).

The cell traversal costs for the path planner are calculated according to the function depicted in Fig. 9: With increasing distance to obstacles, our obstacle cost function $h_c(d)$ decreases until distance D_o. Starting at a distance D_{p1}, the perception cost function h_p increases up to a maximum at D_{p2} to keep the obstacles in the observable range of the MAV. Our cost model $h(d)$ is:

$$
\begin{aligned}
h_c(d) &= \begin{cases} \infty & \text{if } d \leq D_s, \\ h_{max} \frac{1-d+D_s}{D_o-D_s} & \text{if } D_s < d < D_o, \\ 0 & \text{otherwise;} \end{cases} \\
h_p(d) &= \begin{cases} 0 & \text{if } d \geq D_{p1}, \\ h_{max} \frac{d-D_{p1}}{D_{p2}-D_{p1}} & \text{if } D_{p1} < d < D_{p2}, \\ h_{max} & \text{otherwise;} \end{cases} \\
h(d) &= w_1 \cdot h_c(d) + w_2 \cdot h_p(d),
\end{aligned}
$$

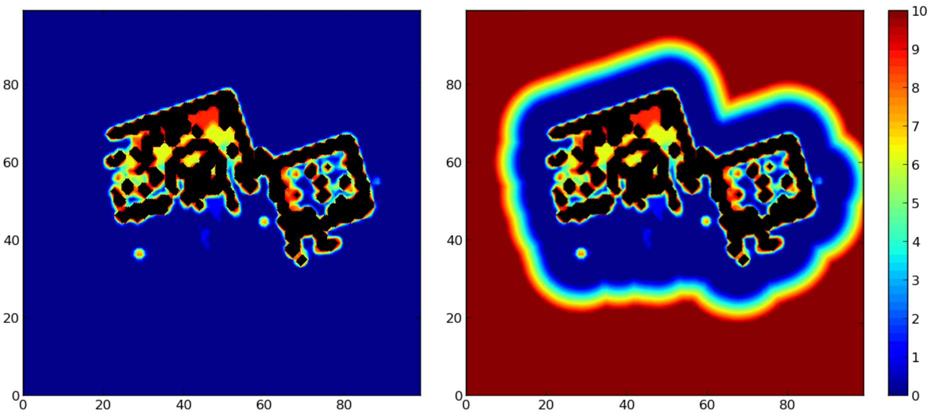


Fig. 10: For robust LRF-based localization, laser scans have to contain sufficient structure. We modify the traversal costs to not only avoid obstacles, but also keep them within sensor range. We compare the resulting traversal costs for a cut through a map some meters above the ground plane. Left: Obstacle avoidance only. Right: Obstacle avoidance and robust localization.

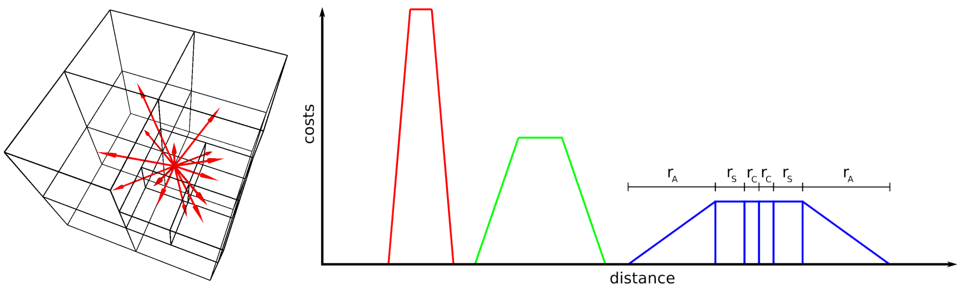


Fig. 11: Local multiresolution path planning. Left: Cut through the robot-centered multiresolution planning grid. Red arrows depict edges from one example cell to its neighbors. Right: We model obstacles in the local multiresolution grid as a fixed core r_C, a safety area with maximum costs r_S, and an avoidance zone with linear decreasing costs r_A. With increasing distance to the grid's origin the radii of these areas increase and their maximum cost decreases to account for the uncertainty in measurements (red: close, green: medium, blue: far away obstacle).

with equal weights $w_1 = w_2$. D_s is the safety distance around obstacles the robot should never enter (at least the robot radius). Fig. 10 illustrates the resulting traversal costs with and without our approach.

6.3 Local Multiresolution Path Planning

On the local path planning layer, we employ a 3D local multiresolution path planner. This layer plans based on the allocentric path from the global path planner, a local excerpt of the global map, and local distance measurements which have been aggregated in a 3D local multiresolution map (Sec. 4.2). It refines the global path according to the actual situation and a finer trajectory is fed to the potential field-based reactive obstacle avoidance layer on the next level (cf. Sec. 6.4).

To resemble the relative accuracy of onboard sensors—i.e., they measure the vicinity of the robot more accurate and with higher density than distant space—we plan with a higher resolution close to the robot and with coarser resolutions with increasing distance.

Local multiresolution for path planning is also motivated by map dynamics. Since parts of the plan, that are farther away from the MAV are more likely to change, e.g., due to newly acquired sensor measurements, it is reasonable to spend more effort into a finer plan in the close vicinity of the robot. Overall, our approach reduces the planning time and makes frequent replanning feasible.

Our planner operates on grid-based robot-centric obstacle maps with higher resolution in the center and decreasing resolution in the distance, similar to the representation in Sec. 4.2. We embed an undirected graph into this grid (Fig. 11) and perform A* search (Hart et al., 1968) from the center of the MAV-centered grid to the goal. The edge costs are given by the base obstacle costs of the cells it is connecting and its length given by the Euclidean distance between the cell centers.

An obstacle is modeled as a core with maximum costs, determined by obstacle radius r_F and enlarged by the approximate robot radius r_R. The obstacle costs are multiplied by the fraction of the edge length within the respective cells. Fig. 11 shows our obstacle model: a core of the perceived obstacle enlarged by the approximate robot radius r_F and a distance-dependent part r_D that models the uncertainty of farther away perceptions and motions with high costs. Added is a part with linearly decreasing costs with increasing distance to the obstacle r_S that the MAV shall avoid if possible. The integral of the obstacle stays constant by reducing its maximum costs h_{max} with increasing radius. For a distance d between a grid cell center and the obstacle center the obstacle costs h_c are given by

$$h_c(d) = \begin{cases} h_{max} & \text{if } d \leq (r_F + r_D) \\ h_{max} \frac{1-d-(r_F+r_D)}{2*(r_F+r_D)} & \text{if } (r_F + r_D) < d < 3*(r_F + r_D) \\ 0 & \text{otherwise} \end{cases} .$$

The local planner is coupled to the solution of the allocentric path planner by a cost term h_a, which is the shortest distance between a grid cell and any segment of the allocentric plan (see Fig. 12). The total cost h for traversing a grid cell is $h = w_1 \cdot h_c(d) + w_2 \cdot h_a$.

6.4 Local Obstacle Avoidance

On the next lower layer, we employ a fast reactive collision avoidance module based on artificial potential fields (Ge and Cui, 2002) as a safety measure reacting directly on the available sensor inputs. The robot-centered local multiresolution occupancy grid, the current motion state, and a target velocity, serve as input to our algorithm. The obstacle map induces repulsive forces on the MAV pushing it away from obstacles. Analysis of the MAV motion model (Beul et al., 2014) gives us a maximum stopping distance of 0.8 m in 0.5 s under ideal conditions. Hence, we can safely operate in the vicinity of obstacles. For more details on the potential field obstacle avoidance layer, please refer to our prior work (Nieuwenhuisen and Behnke, 2014b).

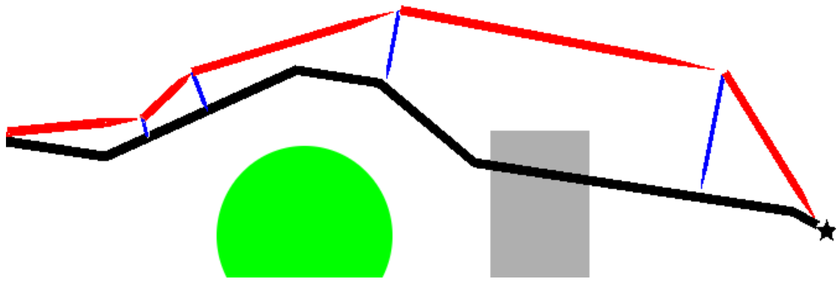


Fig. 12: The local plan (red) is coupled with the allocentric plan (black) by a cost term that penalizes deviations from the allocentric plan. The blue lines depict the deviation vectors at example points, the star is the planner's goal. The green circular obstacle is in the allocentric map, the gray rectangular obstacle has to be surrounded based on the local map.

7 Evaluation

In order to assess the performance and reliability of our system as well as the involved components, we conducted a set of experiments. These range from testing individual components in isolation, to reporting the results of an integrated mission where the complete system accomplishes a series of autonomous missions in complex 3D environments.

7.1 Global Registration and Allocentric Mapping

In order to assess the performance of our global registration and allocentric mapping approach, we tested our approach on a dataset of a parking garage[1]. Without pose graph optimization, the trajectory aggregates drift which results in inconsistencies, indicated by a misalignment of the walls. Our registration method with graph optimization yields accurate results. Fig. 13 shows details of the map depicted in Fig. 6. Here, even narrow structures like pipes can be identified in the globally aligned 3D scans. For a detailed comparision with other registration methods see (Droeschel et al., 2015).

7.2 Global and Local Path Planning

We tested the combination of our global and local path planning layers to evaluate the computing time and the resulting flight trajectories. The MAV follows a globally planned path and has to avoid obstacles that are not in the a priori known world model. For evaluation, we manually removed the furniture from our allocentric map and plan global paths based on this modified map and local paths based on our local

[1] Datasets recorded in-flight with our MAV are available at: `http://www.ais.uni-bonn.de/mav_mapping`.

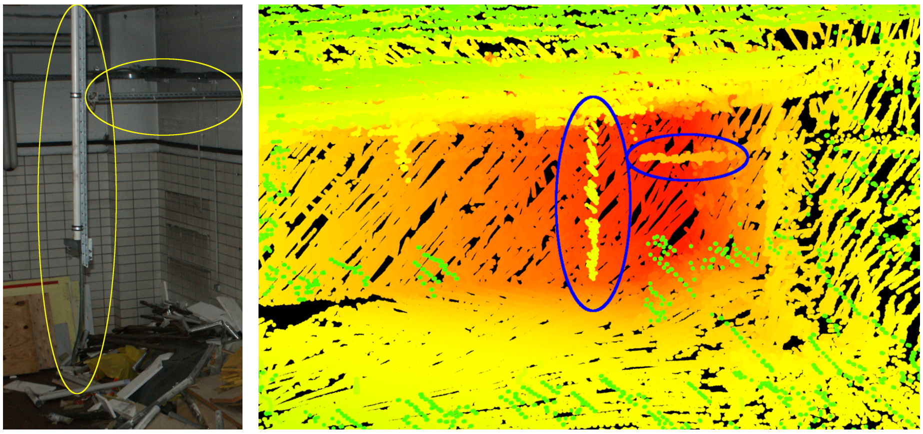


Fig. 13: Impressions of the quality of the built 3D map. Environmental structures are consistently mapped. Even details such as the narrow pipe structure and a cable canal (circled) are accurately modeled. Color encodes the distance to the view-point.

Table 1: Planning time of local path planner. Case A: Maximum planning time if the allocentric plan can be followed. Case B: Maximum planning time when deviating from the allocentric plan.

grid representation	**cell size**	**planning time (ms)**		
		min.	max. Case A	max. Case B
multiresolution	0.25 m	10	40	60
uniform	0.25 m	10	210	720
uniform	1.00 m	10	210	8.200

map incorporating laser measurements. Fig. 14 shows plans while locally surrounding one of the shelves in the evaluation area. When newly perceived obstacles have to be avoided, the planning time for a uniform grid with high resolution can substantially exceed the time window for replanning. In contrast, the local multiresolution planning is always fast enough for continuous replanning.

We compared the computation times of our local planner when employing different planner representations: two uniform grids and our local multiresolution grid with $8 \times 8 \times 8$ cells per level. All grids have a size of $32 \times 32 \times 32$ m. We measured the runtimes on the onboard PC employing data recorded during autonomous operation for our integration experiments described in the next section. Tab. 1 summarizes the results. While the planning times remain sufficiently low for frequent replanning when employing the multiresolution grid, the maximum planning times become prohibitively long when using the uniform grids. For further results, please refer to our previous work on local multiresolution path planning (Nieuwenhuisen and Behnke, 2014a).

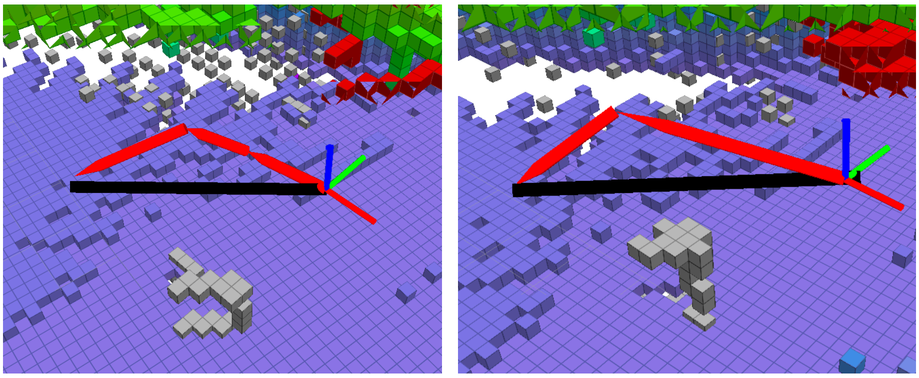


Fig. 14: The allocentric path planner (black line) plans globally consistent paths based on our allocentric map (blue and green boxes). Our local path planner (red arrows) is coupled to the allocentric plan, but surrounds obstacles in the vicinity of the MAV based on our local obstacle map (gray and red boxes). The MAV pose is depicted by the axes.

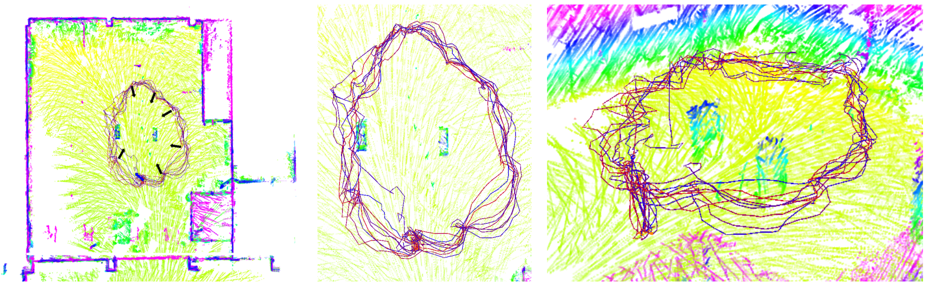


Fig. 15: We conducted indoor flight experiments in a $20 \times 16 \times 6$ m hall. The trajectories show the laser-based localization estimates of 10 consecutive autonomous flights. From left to right: map top-view, closeup of the trajectories, perspective view. Observation poses are depicted by black arrows, the blue arrow shows the return pose. The map color encodes height.

7.3 Evaluation of the Integrated System

We evaluated the integration of our components into one working MAV mapping and inspection system. Here, we describe one exemplary indoor inspection session of our MAV. The main goal of this experiment was to autonomously navigate to certain predefined waypoints in a hall, employing solely means of localization that are available in both indoor and outdoor environments. During flight, the MAVs mission was to detect visual features (AprilTags (Olson, 2011)), representing interesting locations near the trajectory.

Local navigation and control were performed using visual odometry and allocentric localization was performed employing our laser-based approach.

First, we navigated the MAV manually through a hall, a garage, and an outdoor part connecting these two buildings. Subsequently, we built a map from the data

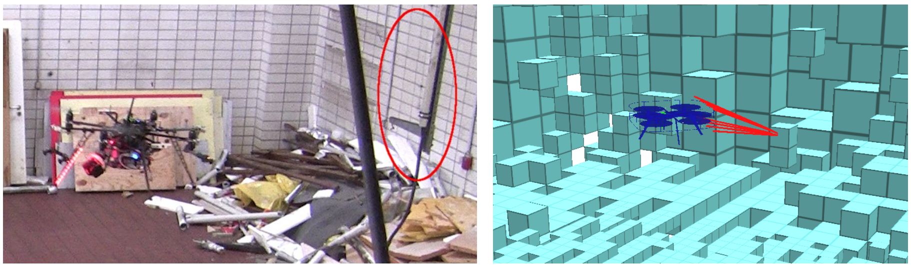


Fig. 16: The MAV is repelled from a small pipe structure (red circled) hanging from the ceiling close to the MAV's view pose. The right figure shows how the obstacle is perceived by the MAV. Red lines depict the artificial repelling forces of the reactive collision avoidance.

collected during this flight (Fig. 6). The map is used for localization and we derived an OctoMap for mission and path planning. In applications where such an initial flight is not feasible, building construction plans could be used instead.

Second, we defined a mission with six observation poses—plus a return pose 2 m above the start pose—in the smaller building part, a decommissioned car service station. Our mission planner plans paths between every pair of mission poses and determines the best visiting order. After takeoff, the global planner begins to continuously plan paths to the next mission-relevant pose. The local obstacle avoidance keeps the MAV successfully away from obstacles like hanging cables and debris, lying around in the hall. In these experiments, we planned allocentric paths in a grid with a cell size of 0.5 m. An excerpt from the map, the inspection poses, and the traversed trajectories of ten missions are shown in Fig. 15. The MAV successfully reached all poses in the experiments without colliding with an obstacle and was localized all the time. In experiments without running localization module, the MAV still was able to avoid dynamic and static obstacles with the local collision avoidance layer, since this layer solely relies on egocentric velocity estimates, e.g., from an integration of visual odometry, accelerometers, and gyroscopes.

Fig. 16 shows a situation from one of the runs where the MAV is flying towards a mission view pose. Close to the view pose, a long thin structure is hanging from the ceiling. The structure is perceived with the onboard sensors and avoided by means of artificial repelling forces. In our test setup, many smaller and larger structures obstruct the free space. In all cases, the MAV deviated from the direct path or moved away from a hover position to avoid a collision. Fig. 17 shows the magnitudes of repelling forces while fulfilling a planned mission. The shown positions are from the ten autonomous flights shown in Fig. 15 plus one additional mission with observation poses closer to some obstacles. Samples for illustration are taken every 500 ms.

During the execution of the inspection mission, the MAV detects visual features. As seen in Fig. 17, we use AprilTags with 26 cm side length as a surrogate for interesting locations near the trajectory. Fig. 18 shows a map of the detected AprilTags during ten flights. It can be seen that the clustering of all detections is within 0.5 m and that most tags are detected many times. So, even if, e.g., smoke would prevent the perception from one side, the tag could nevertheless be perceived. The detection and mapping of AprilTags is detailed in (Beul et al., 2015).

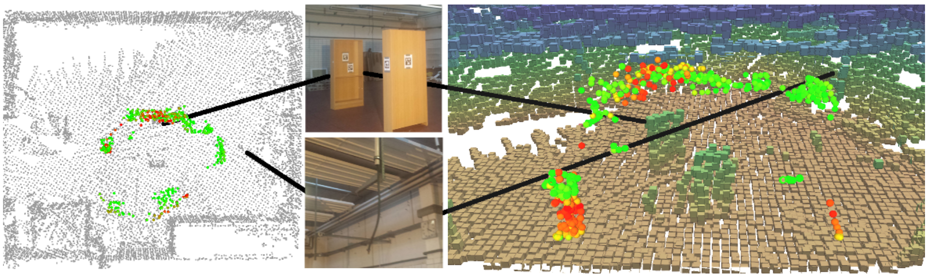


Fig. 17: Artificial forces from the reactive collision avoidance. Shown is the strength of repelling forces pushing the MAV away from obstacles at samples along the trajectories of ten autonomous flights. Green: small magnitude / Red: large magnitude. The photos show some obstacles and their approximate position in the maps.

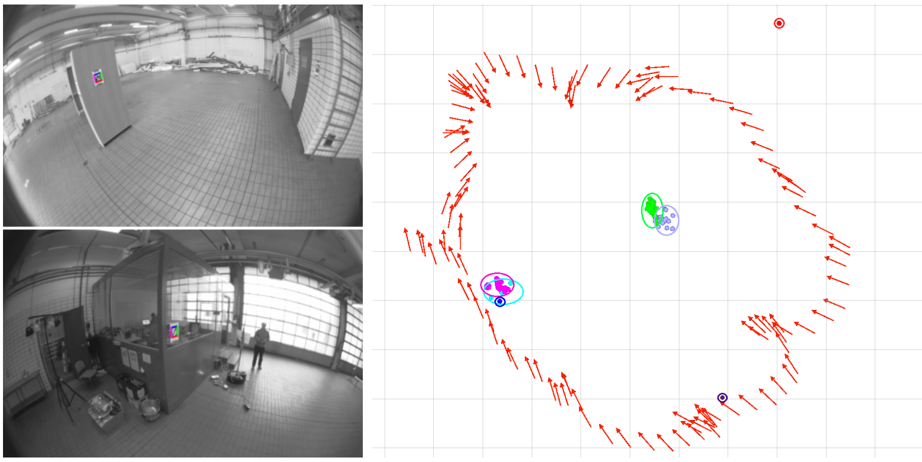


Fig. 18: Left: AprilTag detections in front and rear camera images. Right: Map of detected AprilTags of ten consecutive flights. Seven different tags are perceived and marked in magenta, cyan, blue, green, gray, violet and red. A sample trajectory is shown in red. The grid size is 1 × 1 m.

Videos of autonomous operation, collision avoidance, and mapping are available on our website[2].

8 Conclusions

In this article, we presented an integrated system to autonomously operate MAVs safely in the vicinity of obstacles. We approached this challenge by employing local multiresolution mapping and planning techniques that facilitate frequent updates and replanning.

We showed that by incorporating multimodal sensor information we are able to

[2] www.ais.uni-bonn.de/MoD

detect and avoid diverse obstacles. Pose estimation based on camera and laser data enables robust motion control in GNSS-denied environments.

Laser-range measurements are aggregated by registering sparse 3D scans with a local multiresolution surfel map. Modeling measurement distributions within voxels by surface elements allows for efficient and accurate registration of 3D scans with the local map. The incrementally built local dense 3D maps of nearby key poses are registered globally by graph optimization. This yields a globally consistent dense 3D map of the environment. We demonstrate accuracy and efficiency of our approach by showing consistent allocentric 3D maps, recorded by our MAV during flight.

We demonstrated that multilayered navigation planning results in a high capability to cope with dynamically changing environments and perpetually new obstacle perceptions. By employing a global-to-local approach in our navigation planning pipeline, we achieve replanning frequencies that match the rate of expected changes in the environment model. A reactive collision avoidance layer accounts for fast MAV and environment dynamics and refines higher-level mission plans based on onboard sensing and a priori information.

We showed the system robustness in multiple indoor experiments where the only manual interactions were the starting and landing phases. Thus, the system is able to inspect areas in a fully autonomous mission.

Acknowledgments

This work has been supported by grants BE 2556/7 and BE 2556/8 of German Research Foundation (DFG) and by the German Federal Ministry for Economic Affairs and Energy (BMWi) in the Autonomics for Industry 4.0 project InventAIRy.

References

Achtelik, M. W., Lynen, S., Weiss, S., Chli, M., and Siegwart, R. (2014). Motion- and uncertainty-aware path planning for micro aerial vehicles. *Journal of Field Robotics*, 31(4):676–698.

Anderson, S. and Barfoot, T. D. (2013). Towards relative continuous-time SLAM. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*, pages 1033–1040.

Applegate, D., Bixby, R., Chvatal, V., and Cook, W. (2006). Concorde TSP solver.

Bachrach, A., He, R., and Roy, N. (2009). Autonomous flight in unstructured and unknown indoor environments. In *Proc. of European Micro Aerial Vehicle Conf. (EMAV)*.

Bachrach, A., Prentice, S., He, R., Henry, P., Huang, A. S., Krainin, M., Maturana, D., Fox, D., and Roy, N. (2012). Estimation, planning, and mapping for autonomous flight using an RGB-D camera in GPS-denied environments. *The International Journal of Robotic Research*, 31(11):1320–1343.

Beul, M., Behnke, S., and Worst, R. (2014). Nonlinear model-based 2D-position control for quadrotor UAVs. In *Proceedings of the Joint Int. Symposium on Robotics (ISR) and the German Conference on Robotics (ROBOTIK)*.

Beul, M., Krombach, N., Zhong, Y., Droeschel, D., Nieuwenhuisen, M., and Behnke, S. (2015). A high-performance MAV for autonomous navigation in complex 3D environments. In *Proc. of the Int. Conference on Unmanned Aircraft Systems (ICUAS)*.

Bosse, M. and Zlot, R. (2009). Continuous 3D scan-matching with a spinning 2D laser. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*, pages 4312–4319.

Cover, H., Choudhury, S., Scherer, S., and Singh, S. (2013). Sparse tangential network (SPARTAN): Motion planning for micro aerial vehicles. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Droeschel, D., Nieuwenhuisen, M., Beul, M., Holz, D., Stückler, J., and Behnke, S. (2015). Multi-layered mapping and navigation for autonomous micro aerial vehicles. *Journal of Field Robotics*. published online.

Droeschel, D., Stückler, J., and Behnke, S. (2014a). Local multi-resolution representation for 6D motion estimation and mapping with a continuously rotating 3D laser scanner. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Droeschel, D., Stückler, J., and Behnke, S. (2014b). Local multi-resolution surfel grids for MAV motion estimation and 3D mapping. In *Proc. of the Int. Conference on Intelligent Autonomous Systems (IAS)*.

Elseberg, J., Borrmann, D., and Nuechter, A. (2012). 6DOF semi-rigid SLAM for mobile scanning. In *Proc. of the IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS)*, pages 1865–1870.

Flores, G., Zhou, S., Lozano, R., and Castillo, P. (2014). A vision and GPS-based real-time trajectory planning for a MAV in unknown and low-sunlight environments. *Journal of Intelligent & Robotic Systems*, 74(1-2):59–67.

Fossel, J., Hennes, D., Claes, D., Alers, S., and Tuyls, K. (2013). OctoSLAM: A 3D mapping approach to situational awareness of unmanned aerial vehicles. In *Proc. of the Int. Conference on Unmanned Aircraft Systems (ICUAS)*, pages 179–188.

Ge, S. and Cui, Y. (2002). Dynamic motion planning for mobile robots using potential field method. *Autonomous Robots*, 13(3):207–222.

Grzonka, S., Grisetti, G., and Burgard, W. (2009). Towards a navigation system for autonomous indoor flying. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Grzonka, S., Grisetti, G., and Burgard, W. (2012). A fully autonomous indoor quadrotor. *IEEE Trans. on Robotics*, 28(1):90–100.

Hart, P. E., Nilsson, N. J., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. on Systems Science and Cybernetics*, 4(2):100–107.

Heng, L., Honegger, D., Lee, G. H., Meier, L., Tanskanen, P., Fraundorfer, F., and Pollefeys, M. (2014). Autonomous visual mapping and exploration with a micro aerial vehicle. *Journal of Field Robotics*, 31(4):654–675.

Holz, D., Nieuwenhuisen, M., Droeschel, D., Schreiber, M., and Behnke, S. (2013). Towards mulimodal omnidirectional obstacle detection for autonomous unmanned aerial vehicles. In *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. (ISPRS)*, volume XL-1/W2, pages 201–206.

Hornung, A., Wurm, K. M., Bennewitz, M., Stachniss, C., and Burgard, W. (2013). OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 34:189–206.

Huh, S., Shim, D., and Kim, J. (2013). Integrated navigation system using camera and gimbaled laser scanner for indoor and outdoor autonomous flight of UAVs. In *Proc. of the IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS)*, pages 3158–3163.

Israelsen, J., Beall, M., Bareiss, D., Stuart, D., Keeney, E., and van den Berg, J. (2014). Automatic collision avoidance for manually tele-operated unmanned aerial vehicles. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Kohlbrecher, S., Meyer, J., von Stryk, O., and Klingauf, U. (2011). A flexible and scalable SLAM system with full 3D motion estimation. In *Proc. of the IEEE Int. Symposium on Safety, Security and Rescue Robotics (SSRR)*.

Kuemmerle, R., Grisetti, G., Strasdat, H., Konolige, K., and Burgard, W. (2011). g^2o: A general framework for graph optimization. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*, pages 3607–3613.

Luo, C., Espinosa, A. P., Pranantha, D., and Gloria, A. D. (2011). Multi-robot search and rescue team. In *Proc. of the IEEE Int. Symposium on Safety, Security and Rescue Robotics (SSRR)*.

MacAllister, B., Butzke, J., Kushleyev, A., Pandey, H., and Likhachev, M. (2013). Path planning for non-circular micro aerial vehicles in constrained environments. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Maddern, W., Harrison, A., and Newman, P. (2012). Lost in translation (and rotation): Fast extrinsic calibration for 2D and 3D LIDARs. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Magnusson, M., Duckett, T., and Lilienthal, A. J. (2007). Scan registration for autonomous mining vehicles using 3D-NDT. *Journal of Field Robotics*, 24(10):803–827.

Magree, D., Mooney, J. G., and Johnson, E. N. (2014). Monocular visual mapping for obstacle avoidance on UAVs. *Journal of Intelligent & Robotic Systems*, 74(1-2):17–26.

Meier, L., Tanskanen, P., Heng, L., Lee, G., Fraundorfer, F., and Pollefeys, M. (2012). PIXHAWK: A micro aerial vehicle design for autonomous flight using onboard computer vision. *Autonomous Robots*, 33(1-2):21–39.

Michael, N., Shen, S., Mohta, K., Kumar, V., Nagatani, K., Okada, Y., Kiribayashi, S., Otake, K., Yoshida, K., Ohno, K., Takeuchi, E., and Tadokoro, S. (2012). Collaborative mapping of an earthquake-damaged building via ground and aerial robots. In *Int. Conf. on Field and Service Robotics (FSR)*.

Montemerlo, M., Becker, J., Bhat, S., Dahlkamp, H., Dolgov, D., Ettinger, S., Haehnel, D., Hilden, T., Hoffmann, G., Huhnke, B., et al. (2008). Junior: The stanford entry in the urban challenge. *Journal of Field Robotics*, 25(9):569–597.

Mori, T. and Scherer, S. (2013). First results in detecting and avoiding frontal obstacles from a monocular camera for micro unmanned aerial vehicles. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Morris, W., Dryanovski, I., Xiao, J., and Member, S. (2010). 3D indoor mapping for micro-UAVs using hybrid range finders and multi-volume occupancy grids. In *RSS 2010 workshop on RGB-D: Advanced Reasoning with Depth Cameras*.

Nieuwenhuisen, M. and Behnke, S. (2014a). Hierarchical planning with 3D local multiresolution obstacle avoidance for micro aerial vehicles. In *Proceedings of the Joint Int. Symposium on Robotics (ISR) and the German Conference on Robotics (ROBOTIK)*.

Nieuwenhuisen, M. and Behnke, S. (2014b). Layered mission and path planning for MAV navigation with partial environment knowledge. In *Proc. of the Int. Conference on Intelligent Autonomous Systems (IAS)*.

Nolan, A., Serrano, D., Sabaté, A. H., Mussarra, D. P., and Pena, A. M. L. (2013). Obstacle mapping module for quadrotors on outdoor search and rescue operations. In *Int. Micro Air Vehicle Conf. and Flight Competition (IMAV)*.

Nuechter, A., Lingemann, K., Hertzberg, J., and Surmann, H. (2005). 6D SLAM with approximate data association. In *Int. Conf. on Advanced Robotics (ICAR)*, pages 242 – 249.

Olson, E. (2011). AprilTag: A robust and flexible visual fiducial system. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Park, J. and Kim, Y. (2014). 3D shape mapping of obstacle using stereo vision sensor on quadrotor UAV. In *AIAA Guidance, Navigation, and Control Conference*.

Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. Y. (2009). ROS: An open-source robot operating system. In *ICRA Workshop on Open Source Software*.

Ross, S., Melik-Barkhudarov, N., Shankar, K. S., Wendel, A., Dey, D., Bagnell, J. A., and Hebert, M. (2013). Learning monocular reactive UAV control in cluttered natural environments. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Roy, N., Burgard, W., Fox, D., and Thrun, S. (1999). Coastal navigation-mobile robot navigation with uncertainty in dynamic environments. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Ryde, J. and Hu, H. (2010). 3D mapping with multi-resolution occupied voxel lists. *Autonomous Robots*, 28:169 – 185.

Schadler, M., Stückler, J., and Behnke, S. (2013). Multi-resolution surfel mapping and real-time pose tracking using a continuously rotating 2D laser scanner. In *Proceedings of 11th IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*.

Schauwecker, K. and Zell, A. (2014). On-board dual-stereo-vision for the navigation of an autonomous MAV. *Journal of Intelligent & Robotic Systems*, 74(1-2):1–16.

Schmid, K., Lutz, P., Tomic, T., Mair, E., and Hirschmüller, H. (2014). Autonomous vision-based micro air vehicle for indoor and outdoor navigation. *Journal of Field Robotics*, 31(4):537–570.

Schneider, J., Läbe, T., and Förstner, W. (2013). Incremental real-time bundle adjustment for multi-camera systems with points at infinity. In *ISPRS Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XL-1/W2.

Shen, S., Michael, N., and Kumar, V. (2011). Autonomous multi-floor indoor navigation with a computationally constrained micro aerial vehicle. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Shen, S., Mulgaonkar, Y., Michael, N., and Kumar, V. (2014). Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft mav. In *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*.

Stoyanov, T. and Lilienthal, A. (2009). Maximum likelihood point cloud acquisition from a mobile platform. In *Proc. of the Int. Conf. on Advanced Robotics (ICAR)*, pages 1–6.

Stückler, J. and Behnke, S. (2014). Multi-resolution surfel maps for efficient dense 3D modeling and tracking. *Journal of Visual Communication and Image Rep-*

resentation, 25(1):137–147.

Takahashi, M., Schulein, G., and Whalley, M. (2008). Flight control law design and development for an autonomous rotorcraft. In *Proceedings of the 64th Annual Forum of the American Helicopter Society*.

Thrun, S., Diel, M., and Hähnel, D. (2003). Scan alignment and 3-D surface modeling with a helicopter platform. In *Int. Conf. on Field and Service Robotics (FSR)*, volume 24 of *Springer Tracts in Advanced Robotics*, pages 287–297. Springer.

Tomić, T., Schmid, K., Lutz, P., Domel, A., Kassecker, M., Mair, E., Grixa, I., Ruess, F., Suppa, M., and Burschka, D. (2012). Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue. *Robotics Automation Magazine, IEEE*, 19(3):46–56.

Tripathi, A., G Raja, R., and Padhi, R. (2014). Reactive collision avoidance of UAVs with stereovision camera sensors using UKF. In *Advances in Control and Optimization of Dynamical Systems*, pages 1119–1125.