



A 3D Rotating Laser-Based Navigation Solution for Micro Aerial Vehicles in Dynamic Environments

Hailong Qin^{*,†,||}, Yingcai Bi^{*,†,||}, Lin Feng^{***}, Y. F. Zhang^{§,††}, Ben M. Chen^{‡,##}

^{*}Temasek Laboratories, National University of Singapore, 117411, Singapore

[†]Department of Electrical & Computer Engineering, NUS 117583, Singapore

[‡]Department of Mechanical & Automation Engineering, Chinese University of Hong Kong, Shatin, N.T., Hong Kong

[§]Department of Mechanical Engineering, NUS 117575, Singapore

In this paper, we present a 3D rotating laser-based navigation framework for micro aerial vehicles (MAVs) to fly autonomously in dynamic environments. It consists of a 6-degree of freedom (DoF) localization module and a 3D dynamic mapping module. A self-designed rotating laser scanner generates dense point clouds in which 3D features are extracted and aligned. The localization module is able to solve scan distortion issue while estimating the 6-DoF pose of MAVs. At the same time, the dynamic mapping module can further eliminate dynamic trails so that a clear dense 3D map is reconstructed. The dynamic targets are detected based on the spatial constraints and therefore without the need of dense point cloud clustering. Through filtering the detected dynamic obstacles, the localization approach can be robust to the dynamic environment variations. To verify the robustness and effectiveness of our proposed framework, we have tested our system in both real indoor environment with dynamic obstacles and outdoor foliage condition using a customized MAV platform.

Keywords: 3D rotating laser scanner; MAV; motion estimation; dynamic obstacle detection; 3D mapping.

US

1. Introduction

The applications of MAVs on autonomous tasks have been intensively studied [1–3]. Despite numerous configurations of MAVs, the fundamental requirements for automation of MAVs essentially tend to coincide: (1) Onboard sensing capability: the MAV should utilize the onboard sensors such as LIDAR [4], vision [5] and multiple sensors (sonar and radar) [6] to obtain information about an unknown environment. (2) Onboard processing capability: based on the sensing information, the onboard processor should achieve accurate state estimation and environment mapping for navigation [7]. (3) Onboard state feedback capability: the sensing information from external sensors should be

fused with onboard inertial measurement unit (IMU) for a real-time closed feedback control loop [8]. The above capabilities are critical to the safe navigation of MAVs. More importantly, they are in fact interdependent, performance deterioration of any single component could degrade the whole performance of the autonomous MAV. To summarize, the above-mentioned capabilities can be formed into sensing, perception (including state estimation and mapping) and control modules correspondingly.

Regarding the execution of autonomous tasks in GPS-denied environments for an MAV, the crucial first requirement is to locate itself. Moreover, obstacles should be reliably detected in real time to generate an obstacle-free path for flying [9]. Traditionally, the underlying assumption for the simultaneous localization and mapping (SLAM) is a static environment because the current map measurement should be registered into a previous global map. The practical situation includes a lot of moving targets which could confuse the perception algorithm. The perception module is no longer dealing with the static environment alone.

Received 30 May 2017; Revised 19 September 2018; Accepted 19 September 2018; Published 16 November 2018. This paper was recommended for publication in its revised form by editorial board member, Guido de Croon. Email Addresses: ^{*}e0114868@u.nus.edu, ^{||}yingcaibi@u.nus.edu, ^{**}tsllinf@nus.edu.sg, ^{††}mpzyf@nus.edu.sg, ^{##}bmchen@cuhk.edu.hk

For an autonomous system, the moving targets could degrade the performance in following conditions: (1) If the perception module still works (can estimate the ego-motion of MAVs), then the map cannot update correctly in the dynamic environments and leave dynamic trails [10]. The dynamic trails could further affect the planning since the obstacles cannot be detected appropriately. (2) The worst condition is the failure of perception module because the environment is no longer static, the ego-motion of MAVs could not be estimated anymore.

In this paper, we present a customized integrated system consisting of an MAV platform with an onboard processor, a rotating laser-based localization module, and a dynamic mapping module. The MAV platform provides onboard computation and flying capability. The localization module utilizes the continuous rotating laser scans to achieve a high-frequency motion estimation update. Also, the dynamic mapping module detects and eliminates the moving target for both localization and real-time mapping. The developed solution allows for reliable motion estimation and efficient dynamic map update. The effectiveness of the proposed framework is evaluated in multiple environments.

2. Related Works

The demanding of MAVs to fly autonomously in GPS-denied environments such as indoor and forest are rapidly growing. Making use of different platforms and sensors for perception, several research groups have carried out some demonstrations for practical applications.

Among all the proposed solutions, the vision-based approach is selected commonly by ignoring the illumination change issue [11]. Stephan Weiss *et al.* [7] from ASL use an onboard monocular camera together with an IMU to achieve autonomous flight. Similarly, Forster *et al.* [12] deployed a similar sensor setup, but a semi-dense image registration approach. The proposed approach for MAV autonomous fly shows that the motion estimation can be efficient and robust through pixel intensity-based alignment. However, a computationally expensive dense mapping module is required for a complete 3D navigation [13].

To observe scene depth directly, many groups utilize the stereo camera [14]. However, this approach could fail in the textureless environment. The RGBD camera is widely applied for MAVs' indoor navigation tasks because it can measure the depth from projecting infrared pattern [2]. Compared with passive camera systems, the infrared pattern can estimate the depth of the textureless condition such as the white wall. However, the view field of the infrared pattern is limited which could cause problems when the obstacles are out of the field of view. To cope with this

issue, it is straightforward to equip with multiple RGBD cameras. However, the computational cost rises correspondingly.

The laser-based state estimation technologies are adopted mainly by unmanned ground vehicles (UGVs). For instance, the utilization of 2D LIDAR on small size UGVs — Hector SLAM [15] estimates the 2D motion and boost it to 3D with multiple sensors. Together with the height measurement, a 3D Octomap can be constructed for obstacle detection. However, this approach relies on the reliable and varying height measurements or else it will result in an inconsistent or incomplete 3D map. Similar to this approach, Morris *et al.* [16] utilize the sparse visual features together with the 2D LIDAR for motion estimation. It is still not a direct 3D measurement. Different from 2D LIDAR, the 3D LIDAR could provide accurate 3D dense range measurements regardless of the illumination variation. The sensing capability of 3D LIDAR makes it possible for UGVs to detect obstacles in all directions. Combining with camera systems for state estimation, the sensing and perception modules are robust and effective. Considering the size and weight limitations, the 3D LIDARs are rarely utilized on MAVs. Besides that, the power consumption of a 3D LIDAR could lead to the reduction of endurance.

A combination of a camera for localization and laser for obstacle detection is another choice: Either monocular camera or stereo camera can provide a 6-DoF motion so that the state estimation is not limited to 2D planar space. Based on the state estimation information, either the static or rotating LIDAR can construct a 3D map. A similar approach is proposed by Cover *et al.* [11], but this combination could not work in the dark environment and with obstacles out of view. Matthias *et al.* [4] build a MAV platform with the camera system for odometry and rotating laser scanner for obstacles detection. However, illumination variation and the textureless scene will bring the failure of motion estimation.

To deal with moving objects in the scene, most of the vision-based approaches reject moving targets implicitly in the motion estimation phase through the iterative process [17], but this approach is not very effective to handle slow-moving targets and is computationally expensive. Laser-based methods typically handle this by checking the inconsistency between different scans. Burgard *et al.* [18] propose to differentiate the dynamic and static cells in 2D grid map by the expectation maximization. This approach is straightforward to extend to 3D but expensive in computation due to the ray-tracing. Azim and Aycard [19] proposed a moving object detection method by representing point clouds as an octree-based occupancy grid. Each voxel in the occupancy grid is labeled as free or occupied by ray-tracing. Dynamic voxels are detected with both free or dynamic labels in different scans. They are further clustered

and filtered to provide dynamic object bounding box. Asvadi *et al.* [20] used voxel grids to represent the dense 3d structure. Each voxel is qualified by counting the number of points that fall into the same voxel. After consecutive scans, the objects with large occupancy value are regarded as stationary since multiple scan points should be mapped into the same voxel. Otherwise, those with small occupancy values are detected as moving obstacles. The detection result is further refined with a 2D counters and log likelihood ratio.

In this paper, we develop a motor-driving 2D laser scan system that can estimate the ego-motion and structure of the 3D environment. It is specially designed for MAVs by compromising size, range and weight. Motion is estimated by matching laser scans in an online-built 3D map. Because the sequential range measurements are relative to the dynamic position of the rotating sensor, hence careful scan distortion compensation is implemented to get an accurate motion measurement in this setup. To address the dynamic mapping problem, we explicitly detect the moving objects in the point cloud and remove them during the mapping process. The moving object detection method is inspired from [10] by checking the change between scans. This method represents point cloud on a spherical coordinate system and does not rely on either voxel grid or octree-based occupancy grid. The inconsistency checking is simplified to a distance checking problem, which can detect the moving objects efficiently.

3. System Configuration

We define the body frame of the MAV platform as B and local laser coordinate as L . We use a Dynamixel motor to provide high torque with available rotation speed. For the rotating laser device, one rotation of the laser range finder is from -90° to 90° in CW or 90° to -90° in CCW. This is denoted as θ while the x - y plane is the 0° planar plane. Thus, for k th rotation, the laser coordinate is expressed as L_k . The global coordinate is O . The coordinate system illustration is shown in Fig. 1. For a single laser point p in k th rotation, the coordinate is expressed as $X_{k,l,p}^B$, $X_{k,l,p}^L$ and $X_{k,l,p}^O$ correspondingly. All the defined coordinate systems are right-handed.

In this paper, a sensing and perception module for MAV in dynamic environments is presented. The proposed algorithm contains the following components:

- Feature extraction and alignment: extract the defined feature from 2D laser scan and align the scans in 3D point cloud sequence.
- Motion estimation: estimate the motion based on the optimization of feature alignment error metric.



Fig. 1. Coordinate system illustration in front view.

Algorithm 1 Feature-Based Localization and Dynamic Mapping Algorithm

```

for Points  $\in$  scan ( $L$ ) do
  SearchEdgePoint  $p$ 
  SearchFlatPoint  $q$ 
  for Points  $\in$  scan ( $L + 1$ ) do
    if HasMatchingPoints then
       $D_{\text{Point2Line}} \leftarrow \text{DistanceFunction}(p)$ 
    if HasMatchingPoints then
       $D_{\text{Point2Plane}} \leftarrow \text{DistanceFunction}(q)$ 
  MinimizeDistance( $D_{\text{Point2Line}}, D_{\text{Point2Plane}}$ )
for Points  $\in$  scan ( $L$ ) do
  SphericalCoordinate( $p$ , Localization)
for Points  $\in$  scan ( $L + 1$ ) do
  if SpatialDifference( $p$ , Localization) then
    Points  $p \in$  static
return Localization, Points  $p$ 

```

- Dynamic mapping: update the 3D map for dense mapping and motion estimation refinement.

The following section will describe the above components in detail, and the algorithm is given in Algorithm 1.

3.1. Feature extraction and alignment

We use feature-based method for motion estimation. This section explains the feature detection and association. The challenges of point cloud in a real environment include the large data size, data noise and outliers which make the feature extraction quite difficult as described by Abdul [21]. In their work, outlier rejection and robust fitting are applied to extract reliable features from the point cloud. Fortunately, edge and plane points on a single 2D scan are a good

choice as indicated in [22]. To this end, we select the feature points p in each scan instead of all points in an assembled point cloud to build the correspondence between consecutive point clouds. For 2D laser scan, the variation of a scanning point in a local neighborhood is used to represent the feature point. For a point p in a single scan l of k th rotation, the average length of the edges $l_{\text{avg},p}$ adjacent to it in a local neighborhood:

$$l_{\text{avg},p} = \frac{1}{N} \left\| \sum_{q \in N(l)} (X_{k,l,p}^L - X_{k,l,q}^L) \right\|, \quad (1)$$

where N stands for total number of points adjacent to p and q stands for the neighbor point of p . A relative variation ratio $R_{k,l,p}$ is defined to select edge point $e_{k,l,i}$ (with a variation ratio larger than threshold) or flat point $f_{k,l,i}$ (with a variation ratio smaller than threshold).

$$R_{k,l,p} = \frac{l_{\text{avg},p}}{\|X_{k,l,p}^L\|}. \quad (2)$$

In our work, since we have extracted feature points as edge points and flat points, two Euclidean distance-based relationship will be built accordingly to associate the feature points in different rotations.

- To an edge point p in k th rotation, we can search for the nearest neighbor point of p , m in $(k+1)$ th rotation from the beginning of $(k+1)$ th rotation through reprojection. In order to build a point to line correspondence, we further search for a neighbor point of m in consecutive scan and denote it as n . We can calculate the distance from the edge point p to its corresponding line by the following equation:

$$d_e = \frac{|v_l \times v_e|}{|v_l|}, \quad (3)$$

where v_l stands for $(X_{k+1,l,m}^L - X_{k+1,l,n}^L)$ and v_e stands for $(X_{k,l,p}^L - X_{k+1,l,m}^L)$.

- To a flat point q in k th rotation, we need to search for the nearest neighbor point of q in $(k+1)$ th rotation through reprojection, denoted as r . After that, another two neighbor points of r , s and t need to be searched so that three non-collinear points can construct a plane. The objective is to minimize the distance from the flat point q to its corresponding plane. The unit normal of the plane can be calculated as

$$n = \frac{v_{p1} \times v_{p2}}{|v_{p1} \times v_{p2}|}, \quad (4)$$

where v_{p1} is $(X_{k+1,l,s}^L - X_{k+1,l,r}^L)$ and v_{p2} is $(X_{k+1,l,t}^L - X_{k+1,l,r}^L)$. The distance from flat point q to its corresponding plane can be calculated as

$$d_p = n \cdot v_p, \quad (5)$$

where v_p stands for $(X_{k,l,q}^L - X_{k+1,l,r}^L)$.

3.2. Motion estimation

Translation and rotation of a 3D point p in $(k+1)$ th rotation $X_{k+1,l,p}^L$ with respect to last laser frame L_k are expressed by

$$X_{k,l,p} = \widehat{X}_{k+1,l,p}^L = R X_{k+1,l,p}^L + P, \quad (6)$$

where

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} \begin{bmatrix} c\gamma & -s\gamma & 0 \\ s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$P = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}.$$

Here, α, β and γ are the MAV body euler angles correspondingly.

Motion estimation is to calculate the transformation relationship $T_{k+1}^k = (R_{k+1}^k, P_{k+1}^k)$ utilizing feature correspondences. We can further express the transformation relationship as a nonlinear function f ,

$$f(X_{k+1,l,p}^L) = X_{k,l,p}. \quad (7)$$

Consider the point to line correspondence in Eq. (3). To estimate the motion between $(k+1)$ th rotation and k th rotation, we need to minimize the distance d_e to 0 as described in following function:

$$d_e = w_1(f(X_{k+1,l,m}), f(X_{k+1,l,n})) \rightarrow 0. \quad (8)$$

Similarly, the point to plane distance d_p can be expressed by

$$d_p = w_2(f(X_{k+1,l,r}), f(X_{k+1,l,s}), f(X_{k+1,l,t})) \rightarrow 0. \quad (9)$$

Through combining the feature point correspondence, we can build a nonlinear function $w(T_{k+1}^k) = d$ to express (8) and (9). Therefore, the squared error function can be established as

$$S = (w(T_{k+1}^k))^2. \quad (10)$$

Since the objective value of $w(T_{k+1}^k)$ is 0. To minimize the objective function, mainly two methods can be adopted for real time application, Levenberg-Marquardt(LM) [23] method and Trust-Region-Reflective method (TRR) [24]. Compared with the LM approach, TRR is more accurate and less costly when the incrementally updated result is far away from solution. The update step $\sigma_i = (T_{k+1}^k)_{i+1} - (T_{k+1}^k)_i$ can be obtained by solving

$$\min_{\sigma_i \in N} \psi_i(\sigma_i), \quad (11)$$

where $\psi_i(\sigma) = g^T \sigma + \frac{1}{2} \sigma^T H \sigma$. g and H are the gradient and Hessian, respectively, of w evaluated at $(T_{k+1}^k)_i$ and $\Delta_k > 0$ is the trust region radius. We can further define the

acceptable ratio by

$$r_k = \frac{(\psi_i(\mathbf{0}) - \psi_i(\sigma_i))}{(w(T_{k+1}^k) - w(T_{k+1}^k + \sigma_i))}, \quad (12)$$

which is used to decide the iteration of trial step σ_i .

3.3. Dynamic mapping

The safety of autonomous navigation of MAVs requests the 3D map updating with dynamic obstacles detection. Once the dynamic obstacles are properly detected and removed, the perception module could perform a good localization based on the static objects. In the proposed framework, an efficient structure is kept for dense 3D mapping so that it could maintain a consistent dense representation in the long term. In this paper, we only discuss the 3D map without dynamic objects for perception.

The fundamental assumption to identify the dynamic objects is the consistency of visibility. In the continuously updating environment, if a point along the laser ray is visually blocked by a point that we previously observed, the previous point could be further considered as the dynamic moving target. Instead of directly adopting the ray-tracing for visibility validation [25], the point cloud representation for the map is utilized to ensure smooth information flow. Moreover, the spherical coordinates representation is introduced to represent the geometrical information. The usage of spherical coordinate allows storing the depth of the point in a 2D distance map format [26] so that the dynamic obstacles can be efficiently detected. To a point p in one point cloud set, its corresponding spherical coordinate form is

$$\begin{aligned} \rho &= \sqrt{x^2 + y^2 + z^2}, \\ \theta &= \text{atan2}(y, x), \\ \phi &= \arccos\left(\frac{z}{\rho}\right). \end{aligned} \quad (13)$$

To a point cloud set which is generated by the k th rotation of laser V_k^o in global coordinate, its associated spherical coordinate form is V_k^s . Initially, we assume that most of the points in V_k^s are static. The spatial difference between a point $X_{k+1,l,p}^s$ and $X_{k,l,p}^s$ is calculated by

$$\|X_{k+1,l,p}^s - X_{k,l,p}^s\|. \quad (14)$$

The spatial difference serves to distinguish between the dynamic points and static points. Moreover, the angle between $X_{k+1,l,p}^s$ and upcoming new point in local spherical coordinate could further propagate the knowledge on the points. The higher the angle difference is, the less the previous knowledge has changed. Through this approach, the dynamic points are removed so that the motion could be correctly updated.

4. Experimental Results

The proposed framework is verified in the real environment directly instead of simulation environments to show the robustness and efficiency of the solution. In the designed experiments, the MAV has a maximum flying speed of 1.5 m/s. An onboard camera is installed for online visualization of the scene.

The MAV platform is shown in Fig. 2. It is designed by NUS UAV Group with the capability of wind resistance and heavy payload. The platform has a 128 cm tip to tip length and a maximum 2 kg payload weight. A self-designed power distribution module provides multiple power sources for onboard electronics. The onboard flight controller is Pixhawk with customized flight control algorithms. We use the Intel NUC computer (i7-5557U) as the onboard processing unit.

The Dynamixel RX-24F servo motor with 0.29° rotation resolution and $\pm 150^\circ$ rotation range is used to spin a Hokuyo UTM-30LX laser scanner with a frequency of 1 Hz. The laser scanner provides range data with 270° field of view and effective range of 30 m at 40 Hz. This rotating laser scanner is installed forward-facing on the MAV with a 3D-printing case. We also installed dampers between the casing and MAV frame to reduce vibration.

4.1. Flying in indoor corridor environment

We first test our system in an indoor corridor environment to verify the accuracy of the motion estimation module and mapping module without dynamic obstacles. Figure 3 shows the image of the corridor.

Since the ground truth of the motion estimation is not available, the accuracy of the motion estimation is evaluated



Fig. 2. Customized MAV platform.



Fig. 3. Corridor environment.

by the reconstructed point cloud map. Figure 4 shows the reconstructed corridor.

The total travel distance in the corridor is around 50 meters. The final closed-loop error is less than 0.25 m. We can see from the map that vertical walls keep upright and floor can be recognized as a flat surface.

Figure 5 shows close details of the structure in the map. The image sample of the related area is shown in Fig. 6. We measure the width of each cubicle with the 3D information from the map and compare it with the distance measured manually with a rangefinder. The average error is within 5 cm. It is demonstrated that the accuracy of motion estimation is good enough since the reconstructed environment well represented real condition.

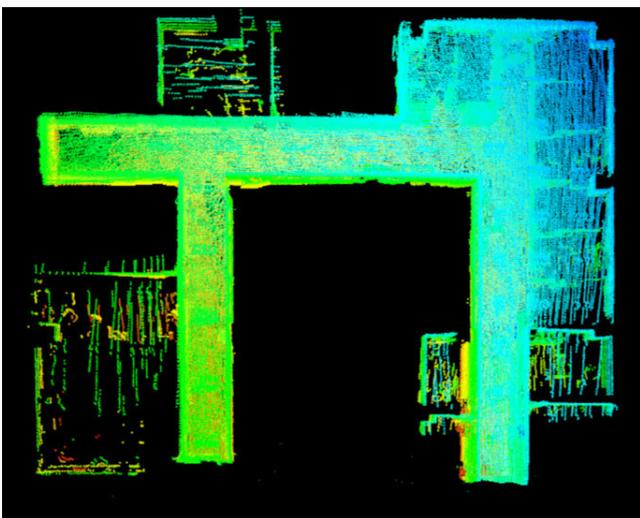


Fig. 4. Reconstructed corridor environment.

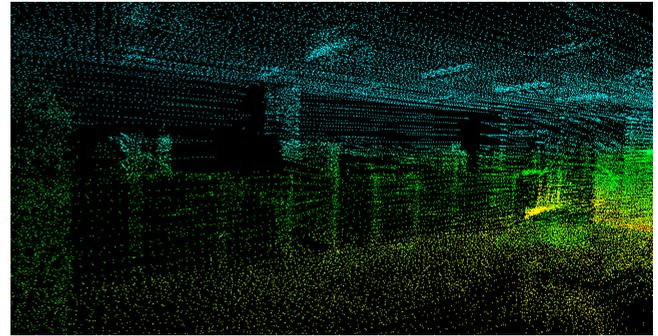


Fig. 5. Reconstructed detail of the corridor environment.

4.2. Flying in outdoor foliage environment

Our system is further tested in a typical outdoor foliage environment shown in Fig. 7. In this experiment, the MAV can autonomously localize itself without any prior knowledge. The 3D point cloud is shown in Fig. 8. As we can see, the ground and wall in front are well-reconstructed



Fig. 6. Image detail of the corridor.



Fig. 7. Outdoor foliage environment.

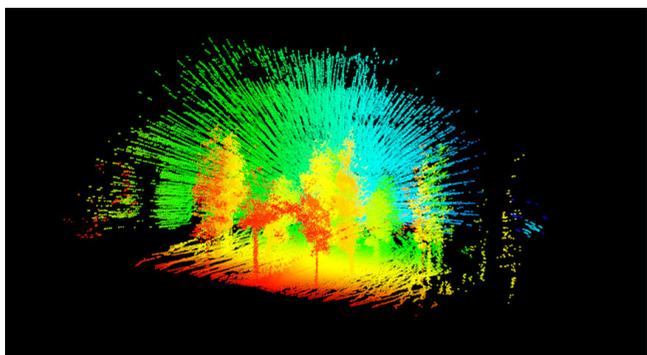


Fig. 8. Reconstructed foliage environment in side view (The reconstructed foliage environment is in front and a vertical wall is behind for visual reference).

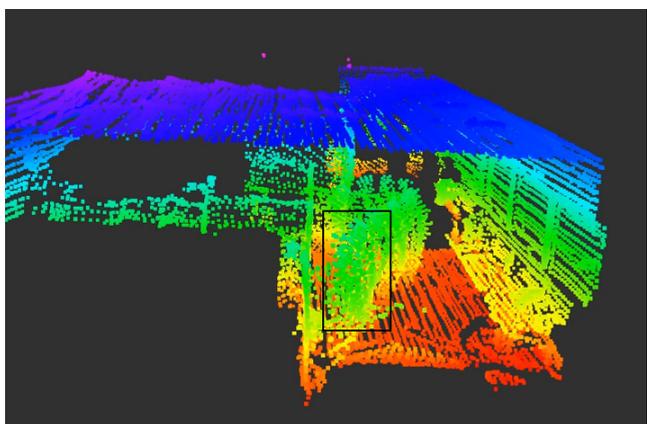


Fig. 9. Moving obstacles caused dynamic trails.

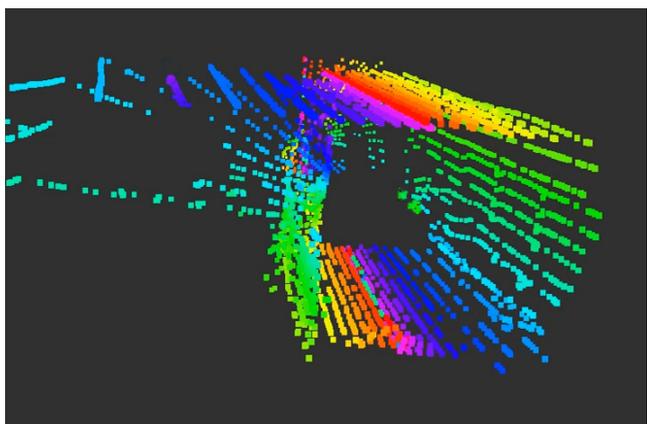


Fig. 10. Dynamic obstacles filtering.

as planes, and the branches of trees are well-depicted. This experiment demonstrates the capability of our system working in outdoor foliage environment. Based on the reconstructed environment, the robustness of the

feature-based localization approach is verified since the Manhattan world assumption does not exist.

4.3. Hovering with dynamic obstacles

This experiment is designed to test the dynamic module specifically. The MAV is controlled by the information from the motion estimation module in the presence of moving obstacles. The failure of the map updating could lead to the dynamic trails as shown in Fig. 9. Based on the experimental result, the dynamic mapping module removes the continuously moving obstacle (target in black rectangle) successfully and keeps a static map for further motion estimation refinement as shown in Fig. 10. During the dynamic obstacle removal phase, the point cloud map is down-sampled to reduce the data size and achieve real-time performance. Hence, the final map looks sparser than the original one.

5. Conclusion

In this paper, we present a rotating laser-based motion estimation and dynamic mapping framework to achieve MAV autonomous navigation in a GPS-denied environment. The proposed framework solves the laser distortion issue by feature-based matching. The ego-motion of MAV is further estimated from point-to-line and point-to-plane optimization. In addition, the dynamic trail is resolved by point-based filtering instead of the voxel grid-based ray-tracing. To verify the accuracy and robustness of the proposed framework, we conduct multiple experiments in different environments, respectively. We demonstrate the accuracy and robustness of our proposed framework in the experiments through analysis and comparison. Based on the result analysis, the rotating laser-based framework estimates the ego-motion of MAV while filtering the dynamic obstacles successfully.

References

- [1] S. Shen, N. Michael and V. Kumar, Autonomous multi-floor indoor navigation with a computationally constrained mav, *IEEE Int. Conf. Robotics and Automation (ICRA)* (2011), pp. 20–25.
- [2] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox and N. Roy, Visual odometry and mapping for autonomous flight using an RGBD camera, *Int. Symp. Robotics Research (ISRR)*, Vol. 2 (Springer, 2011).
- [3] L. Heng, L. Meier, P. Tanskanen, F. Fraundorfer and M. Pollefeys, Autonomous obstacle avoidance and maneuvering on a vision-guided mav using on-board processing, *IEEE Int. Conf. Robotics and Automation (ICRA)* (2011), pp. 2472–2477.
- [4] M. Nieuwenhuisen, D. Droschel, M. Beul and S. Behnke, Autonomous navigation for micro aerial vehicles in complex gnss-denied environments, *Journal of Intelligent & Robotic Systems* **84** (2016) 199–216.

- [5] M. Blösch, S. Weiss, D. Scaramuzza and R. Siegwart, Vision based mav navigation in unknown and unstructured environments, *IEEE Int. Conf. Robotics and Automation (ICRA)* (Taylor & Francis, 2010), pp. 21–28.
- [6] J. Müller and W. Burgard, Efficient probabilistic localization for autonomous indoor airships using sonar, air flow, and imu sensors, *Adv. Robot.* **27**(9) (2013) 711–724.
- [7] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli and R. Siegwart, Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments, *IEEE Int. Conf. Robotics and Automation (ICRA)* (Springer, 2012), pp. 957–964.
- [8] L. Meier, P. Tanskanen, L. Heng, G. H. Lee, F. Fraundorfer and M. Pollefeys, Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision, *Auton. Robots* **33**(1–2) (2012) 21–39.
- [9] S. Lai, K. Wang, H. Qin, J. Q. Cui and B. M. Chen, A robust online path planning approach in cluttered environments for micro rotorcraft drones, *Control Theory Technol.* **14**(1) (2016) 83–96.
- [10] F. Pomerleau, P. Krüsi, F. Colas, P. Furgale and R. Siegwart, Long-term 3d map maintenance in dynamic environments, *IEEE Int. Conf. Robotics and Automation (ICRA)* (Springer, 2014), pp. 3712–3719.
- [11] S. Scherer, J. Rehder, S. Achar, H. Cover, A. Chambers, S. Nuske and S. Singh, River mapping from a flying robot: State estimation, river detection, and obstacle mapping, *Auton. Robots* **33**(1–2) (2012) 189–214.
- [12] C. Forster, M. Pizzoli and D. Scaramuzza, Svo: Fast semi-direct monocular visual odometry, *IEEE Int. Conf. Robotics and Automation (ICRA)* (2014), pp. 15–22.
- [13] M. Pizzoli, C. Forster and D. Scaramuzza, Remode: Probabilistic, monocular dense reconstruction in real time, *IEEE Int. Conf. Robotics and Automation (ICRA)* (2014), pp. 2609–2616.
- [14] C. De Wagter, S. Tijmons, B. D. Remes and G. C. de Croon, Autonomous flight of a 20-gram flapping wing mav with a 4-gram onboard stereo vision system, *IEEE Int. Conf. Robotics and Automation (ICRA)* (2014), pp. 4982–4987.
- [15] S. Kohlbrecher, O. Von Stryk, J. Meyer and U. Klingauf, A flexible and scalable slam system with full 3d motion estimation, *IEEE Int. Symp. Safety, Security and Rescue Robotics (SSRR)* (2011), pp. 155–160.
- [16] W. Morris, I. Dryanovski, J. Xiao et al., 3d indoor mapping for micro-uavs using hybrid range finders and multi-volume occupancy grids, *RSS 2010 Workshop on RGBD: Advanced Reasoning with Depth Cameras* (Citeseer, 2010).
- [17] R. Mur-Artal, J. Montiel and J. D. Tardós, Orb-slam: A versatile and accurate monocular slam system, *IEEE Trans. Robotics* **31**(5) (2015) 1147–1163.
- [18] W. Burgard, C. Stachniss and D. Hähnel, Mobile robot map learning from range data in dynamic environments, *Autonomous Navigation in Dynamic Environments* (Springer, 2007), pp. 3–28.
- [19] A. Azim and O. Aycard, Detection, classification and tracking of moving objects in a 3d environment, *Intelligent Vehicles Symposium (IV)*, *IEEE* (2012), pp. 802–807.
- [20] A. Asvadi, C. Premebida, P. Peixoto and U. Nunes, 3d lidar-based static and moving obstacle detection in driving environments: An approach based on voxels and multi-region ground planes, *Robotics Auton. Syst.* **83** (2016) 299–311.
- [21] A. Nurunnabi, G. West and D. Belton, Robust methods for feature extraction from mobile laser scanning 3d point clouds, *Research@Locate*, Vol. 15 (Elsevier, 2015), pp. 109–120.
- [22] J. Zhang and S. Singh, Loam: Lidar odometry and mapping in real-time, *Robotics: Science and Systems*, Vol. 2 (CEUR-WS.org, 2014), p. 9.
- [23] S. Roweis, Levenberg-marquardt optimization, Notes, University of Toronto (The MIT Press, 1996).
- [24] Y.-X. Yuan, A review of trust region algorithms for optimization, *ICIAM*, Vol. 99 (2000), pp. 271–282.
- [25] F. Ferri, M. Gianni, M. Menna and F. Pirri, Dynamic obstacles detection and 3d map updating, *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)* (2015), pp. 5694–5699.
- [26] P. Gohl, D. Honegger, S. Omari, M. Achtelik, M. Pollefeys and R. Siegwart, Omnidirectional visual obstacle detection using embedded fpga, *2015 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)* (2015), pp. 3938–3943.



Hailong Qin received his B.S and M.S from Harbin Institute of Technology, Harbin, China and Pohang University of Science and Technology, Pohang, Korea in 2011 and 2013, respectively. He has been an associate scientist with Temasek Laboratory, National University of Singapore, Singapore since 2014. His major research interests lie in the area of MAV autonomous navigation and 3D reconstruction.

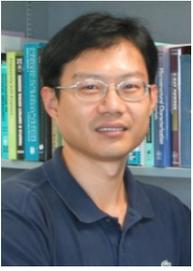


Yingcai Bi received his B.S. from Nanjing University of Aeronautics and Astronautics, China and M.S. from Beihang University, China in 2011 and 2014, respectively. He is currently an associate scientist in Temasek Laboratory, National University of Singapore (NUS), and a Ph.D. candidate in NUS Graduate School for Integrative Sciences and Engineering. His research interests include sensing and perception for MAVs in GPS-denied environments.



Lin Feng received his B.Eng. degree in Computer Science and Control, and the M.Eng. degree in system engineering from Beihang University, Beijing, China, in 2000 and 2003, respectively. He received the Ph.D. degree in Computer & Electrical Engineering from the National University of Singapore in 2011.

Dr. Lin is currently working as a Senior Research Scientist at Temasek Laboratories, National University of Singapore (NUS), and a Research Assistant Professor in the Department of Electrical & Computer Engineering at NUS. His main research interests are unmanned aerial vehicles, vision-aided control and navigation, target tracking, robot vision as well as embedded vision systems. He has served on the editorial board for *Unmanned Systems*. He was the recipient of the Best Application Paper Award, 8th World Congress on Intelligent Control and Automation, Jinan, China (2010).



Y. F. Zhang received his B.Eng. in Mechanical Engineering from Shanghai Jiao Tong University, China in 1985 and Ph.D. from the University of Bath, UK in 1991. He is currently an Associate Professor at the Department of Mechanical Engineering, National University of Singapore. His research interests include building, analyzing, and leveraging smart, interconnected systems, with a focus on data science and artificial intelligence solutions for industrial applications.

Dr. Chen has published more than 400 journal and conference articles, and a dozen research monographs in control theory and applications, unmanned systems as well as financial market modeling. He has served on the editorial boards of several international journals including *IEEE Transactions on Automatic Control* and *Automatica*. He currently serves as an Editor-in-Chief of *Unmanned Systems*. Dr. Chen has received a number of research awards nationally and internationally. His research team has actively participated in international UAV competitions and won many championships in the contests. He is an IEEE Fellow.



Ben M. Chen is currently a Professor in the Department of Mechanical and Automation Engineering, at the Chinese University of Hong Kong. He was a Provost's Chair Professor in the Department of Electrical and Computer Engineering, the National University of Singapore (NUS), where he was also serving as the Director of Control, Intelligent Systems and Robotics Area, and Head of Control Science Group, at NUS Temasek Laboratories. His current research interests are in unmanned systems, robust control, control applications, and financial market modeling.