

Optimized Real Time 3-D Feature Map Generation with Unmanned Aerial Systems

Swee King Phang, Syed Zeeshan Ahmed, Mohamed Redhwan Abdul Hamid

Article Info

Volume 82

Page Number: 3104 - 3112

Publication Issue:

January-February 2020

Article History

Article Received: 14 March 2019

Revised: 27 May 2019

Accepted: 16 October 2019

Publication: 19 January 2020

Abstract

The advancement in unmanned aerial vehicle (UAV) automation has greatly increased the application of autonomous UAVs in both civilian and military tasks. Powerful yet light weight onboard computers are now available to further reduce the size of UAVs to a great extent and thus shrinking them to a size suitable for indoor missions. The main challenge with indoor navigation is usually the lack of GPS reception. Thus, simultaneous localization and mapping algorithms (SLAM) must utilize proximity sensors such as LiDAR, SONAR, and Stereo Cameras. In order to keep the size and weight of the UAV to a minimum, extracting more information from a limited number of sensors is vital. Hence, this research aims at the improvement, implementation and testing of localization and 3D mapping algorithms using a single rotating 2D LiDAR.

Keywords: 3-D Mapping, LiDAR SLAM, Rotating LiDAR.

I. INTRODUCTION

Autonomous UAVs have grown in their use in the last few decades with the focus on developing the autonomous capabilities of these vehicles. Autonomous control using GPS position and velocity signals as feedback has matured in that time as evidenced by the hobby grade UAVs widely available today. Some examples include the toy-grade UAVs (commonly known as drones) from DJI, Yuneec and 3D Robotics.

The attention has in more recent years switched to autonomous navigation within GPS denied environments. Light detection and ranging sensors, or LiDAR, have grown in popularity in these situations however the early single ray laser scanners cannot provide sufficient geometrical information in complex environments. LiDAR sensors have since then evolved from the earlier versions to sensors with spinning mirrors capable of detecting objects 270 deg about its spinning axis and plotting out maps in 2-dimensions (2D). Robotics researchers adopted these sensors quickly, by developing 2D SLAM algorithms to align a sequence of local planar scans into a global 2D map [1]. The laser scanners help estimate autonomous

vehicles' relative position and generate 2D floor maps.

While 2D maps are sufficient for robots to precisely localize itself within a given area, it does not capture obstacles on the single plane map generated from the 2D laser scan. The same is true for outdoor environments thus more detailed information are required in these scenarios. To overcome this, researchers have designed ways to capture full 3D proximity information (point cloud) with the same 2D LiDAR sensors by rotating the whole sensor on a second axis.

Two rotating schemes prove popular, the Nodding and Turning LiDAR. Nodding LiDAR is achieved by placing a 2D LiDAR sensor on a mount controlled by a servo tilting mechanism. Coverage volume of the LiDAR is dependent on the tilting angle, which normally forms a sinusoidal wave. On the other hand, Turning LiDAR has its rotating axis pointing towards the front of the 2D LiDAR sensor. When the sensor is rotated, it forms a sphere of sensor beams. These configurations generate 3D point clouds with different density distributions and are the most widely used. Mechanical constraints, however, limit their application in real-time

operations as each scan takes a few seconds to complete.

For a Nodding LiDAR system, the coverage of the laser scan in the vertical direction depends on the nodding angle which due to mechanical constraints and the update frequency of each scan usually less than a full swipe (180 deg). These systems are generally used to generate maps of the environment to ensure safe autonomous navigation [2]. The 2D map is usually sufficient for UAV navigation if it flies at a constant height [3]. However, the limited field of view and the lack of overlap between adjacent scans taken by the laser sensor pose a problem with this approach, especially when the approach is applied to environment mapping. Essentially, this solution is not able to capture parts of the environment directly above and below the UAV [4].

To overcome this blind-spot problem, Csiro has introduced a device called Zebedee for 360 deg mapping. Zebedee was made with a Hokuyo LiDAR scanner of model UTM-30LX combined with a MicroStrain 3DM-GX3-MEMS inertial measurement unit (IMU) mounted on a spring platform connected to the handle [5]. By waving the device, laser scans on various random angles will be recorded and then fed to a customized software to generate 3D point clouds. Many researchers have also developed a Turning LiDAR system by mounting the laser scanner and the IMU on a servo, which continuously turn between ± 90 degree [6].

In more recent developments, it has been seen that 360 deg turning LiDAR systems running on-board a small-scale UAV [7], or ground vehicle [8] have been achieved. Technically in hardware design, a 360 deg turning system requires a more complex mechanism, such as a precise encoder odometry to measure rotating velocity, and the need of rotating slip ring for wiring. With the benefit of a full rotation, objects above and below the machines can be observed.

One particular challenge that arises in the applications of mapping using turning LiDAR system is when the sensor motion is too fast relative to the sensor update rate, resulting in distorted

scanned data. Turning LiDAR systems faces a challenge when the sensor motion outpaces the measurement time. This results in locally distorted scanned data. A solution to this problem is proposed in [9]–[11], by generating visual images from laser intensity, and then proceed to match visually distinct features between each image to recover the motion of a vehicle. In these works, the vehicle motion is modelled as either constant velocity or Gaussian processes.

For state-of-the-art SLAM algorithm utilizing rotating LiDAR systems, most notably LiDAR Odometry And Mapping (LOAM) work from CMU [12], [13] and Bentwing from Csiro [14], [15] have shown positive results with actual implementation in UAVs. While both solutions employ different algorithms, they have achieved successful results with accurate localization and mapping performance.

Csiro has implemented their localization and mapping algorithm using a freely rotating 2D LiDAR sensor on their UAV codenamed Bentwing. The 2D LiDAR is passively rotated by the downwash of the UAV propellers in the frequency of 1 to 2 Hz. While their algorithm is too computationally intensive to run on-board, they have achieved a high accuracy in mapping with average translation error of 0.1% in indoor, and 0.2% error in outdoor. The data was collected with Bentwing flying at 1 to 2 m/s speed.

On the other hand, researchers from CMU have developed their LOAM algorithm using a different approach, either via a half rotating 2D LiDAR, a fully rotating 2D LiDAR, and also with Velodyne 3D LiDAR sensors. Their algorithm was implemented on a UAV, flying at 1 m/s speed. The result is less accurate as compared to Csiro's method, with translation accuracy of 1% error in indoor environment, and 2.5% error in outdoor environment.

In this manuscript, the LOAM algorithm by CMU will be adopted as our base point to implement on a UAV system with SLAM capability. Section II describes the LOAM algorithm; UAV hardware setup will be listed in Section III; Improvement on

LOAM algorithm implemented on the UAV is shown in Section IV; while the next section shows the flight test results of the designed UAV with the proposed algorithm; finally, concluding remarks are made in Section VI.

II. LIDAR ODOMETRY AND MAPPING ALGORITHM

LOAM is proven to achieve low drift motion estimation at relatively lower computational costs. This is done through the division of SLAM algorithm into two parallel parts. One part performs odometry calculations at high frequency and low fidelity to achieve fast motion estimation. While the other part carries out fine matching and feature registration of the 3D point-cloud at a much lower frequency. For completeness of this manuscript, key equations and idea of the LOAM algorithm will be shown in this Section. They are extracted and summarized from [12] and [13].

A Feature Point Extraction

Feature points are selected from data points that are either on edges or on planar patches. A data point is then sorted as an Edge point E_k or Planar Point H_k , depending upon the c value, which refers to the smoothness of the surface:

$$c = \frac{1}{|S| \cdot \|X_{k,i}^L\|} \left\| \sum_{j \in S, j \neq i} (X_{k,i}^L - X_{k,j}^L) \right\| \quad (1)$$

Here, S is the set consists of consecutive points of i returned by the LiDAR in one scan, such that $i \in P_k$. In the implementation, the environment is sub-divided into four regions to distribute the feature points evenly. Each region can supply a maximum of two Edge features and four Planar features. Each feature point i can be selected as an Edge feature or Planar feature, if its c value is greater or less than a given threshold, respectively.

B Feature Point Correspondences

The point-cloud perceived at the end of one sweep is projected to the starting time stamp of the next sweep, along with the E_k and H_k . Let P_k denote the point cloud perceived for a scan starting at t_k , \bar{P}_{k-1} denotes the projected point-cloud, \bar{E}_k and \bar{H}_k

represent the projections of E_k and H_k respectively. For every point i in \bar{E}_k and \bar{H}_k a corresponding closest neighbor, which is part of an edge-line and planar patch respectively will be found in \bar{P}_{k-1} .

Let $i \in \bar{E}_k$, such that $(j, l) \in \bar{P}_{k-1}$ are the corresponding closest neighbors which form an edge line, the distance of any point to this line can be computed as

$$d_E = \frac{|(\bar{X}_{k,i}^L - \bar{X}_{k-1,j}^L) \times (\bar{X}_{k,i}^L - \bar{X}_{k-1,l}^L)|}{|(\bar{X}_{k-1,j}^L - \bar{X}_{k-1,l}^L)|} \quad (2)$$

where, $\bar{X}_{k,i}^L, \bar{X}_{k-1,j}^L, \bar{X}_{k-1,l}^L$ are the coordinates of the points i, j, l in L_k respectively.

Let $i \in \bar{H}_k$, such that three non-collinear points $j, k, m \in \bar{P}_{k-1}$ represent the planar patch. The distance of any point to the plane is then given by

$$d_H = \frac{|(\bar{X}_{k,i}^L - \bar{X}_{k-1,j}^L) \times (\bar{X}_{k-1,j}^L - \bar{X}_{k-1,m}^L)|}{|(\bar{X}_{k-1,j}^L - \bar{X}_{k-1,l}^L) \times (\bar{X}_{k-1,j}^L - \bar{X}_{k-1,m}^L)|} \quad (3)$$

where, $\bar{X}_{k-1,j}^L, \bar{X}_{k-1,l}^L, \bar{X}_{k-1,m}^L$ are the coordinates of the points j, l, m in L_k respectively.

C Motion Estimation

The rotating motion of the LiDAR is modelled with a constant linear and angular velocity during a sweep, which allows for linear interpolation of the pose transform for the data points received at different times within a single sweep. $T_k^L(t)$ represents the LiDAR pose transform between $[t_k, t]$, which contains the 6-DOF motion of the LiDAR as $T_k^L(t) = [\tau_k^L(t), \theta_k^L(t)]$ where, $\tau_k^L(t) = [t_x, t_y, t_z]$ is the translation and $\theta_k^L(t) = [\theta_x, \theta_y, \theta_z]$ is the rotation in L_k . For any given point $i \in P_k$, $T_{k,i}^L$ is the pose transform between $[t_k, t_{k,i}]$ which can be calculated from a linear interpolation of $T_k^L(t)$ as follows

$$T_{k,i}^L = \frac{t_{k,i} - t_k}{t - t_k} T_k^L(t) \quad (4)$$

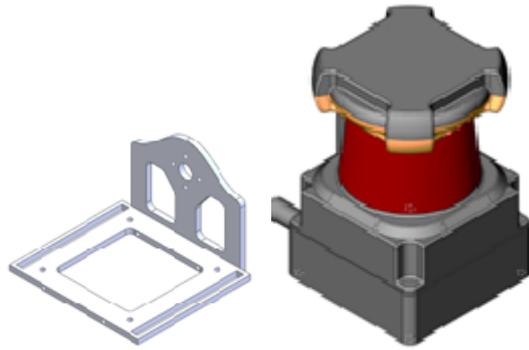


Fig. 2. Solidworks Model of the rotating mechanism

The projection of E_k and H_k as \bar{E}_k and \bar{H}_k respectively to the beginning of the sweep is done as per the following formula

$$\bar{X}_{k,i}^L = R_{k,i}^L X_{k,i}^L + \tau_{k,i}^L \quad (5)$$

where $R_{k,i}^L$ and $\tau_{k,i}^L$ are the rotation and translation matrices corresponding to $T_{k,i}^L$.

From (1) which calculates the distance between a point and its corresponding closest neighbor in an edge line, upon combining this equation with (5). The following relationship between E_k and corresponding edge line can be found by

$$f_E \left(X_{k,i}^L, T_k^L(t) \right) = d_E, \quad i \in E_k \quad (6)$$

Similarly, combining (3) and (5), another geometric relationship between H_k and the corresponding planar patch can be established

$$f_H \left(X_{k,i}^L, T_k^L(t) \right) = d_H, \quad i \in H_k \quad (7)$$

Finally, the motion of the LiDAR can be solved by using the Levenberg-Marquardt method. Stacking (6) and (7) for each feature point in E_k and H_k results in a nonlinear function

$$f \left(T_k^L(t) \right) = d \quad (8)$$

where each row in f represents a feature point where d contains the corresponding distances. With (8), the Jacobian matrix of f with respect to $T_k^L(t)$ can be computed as $J = \partial f / (\partial T_k^L(t))$. Then (8) can be solved via non-linear iterations by minimizing d , as follows

$$T_k^L(t) \leftarrow T_k^L(t) - \left(J^T J + \lambda_{\text{diag}} (J^T J) \right)^{-1} J^T d \quad (9)$$

where λ is a factor identified by Levenberg-Marquardt method.



Fig. 3. Solidworks model of the Digital servo

D LiDAR Mapping

This algorithm runs at a much slower frequency, when compared to LiDARodometry. It is called at the end of each sweep k , where in it registers and matches \bar{P}_k to World coordinates W . Let Q_{k-1} be the point cloud on the global map, accumulated to sweep $k - 1$. T_{k-1}^W is the pose of the LiDAR at the end of sweep $k - 1$, t_k . Using the outputs of LiDAR Odometry, the mapping algorithm extends T_{k-1}^W for one sweep from t_k to t_{k+1} , in order to generate T_{k+1}^W . The transformed \bar{P}_k in the world frame W , is denoted as \bar{Q}_k , which is matched to Q_{k-1} by optimizing the LiDAR pose T_{k+1}^W .

The feature points are extracted ten times as many points as described in the previous section. The matching of the feature points are then be found from points in Q_{k-1} that intersect with \bar{Q}_k within a certain region around the feature points, in this case, $10 \text{ cm} \times 10 \text{ cm} \times 10 \text{ cm}$. Let S' denote the surrounding points, then for any edge point only those points are kept in S' which are a part of edge lines. Similarly, for planar points only those points, which are a part of a planar patch are kept in S' . The co-variance matrix of S' denoted by M is computed along with the eigen-values E and eigen-vectors V .

For an Edge line in S' , V contains one eigenvalue that is significantly greater than the other two, and the E associated with this value represents the direction of the edge line. Whereas if S' contains a value significantly smaller than the other two, the E associated with that value represents the direction of the planar patch.

The distances of the edge and the planar points to their correspondences is calculated as per (2) and (3). Then (7) and (8) are used to determine individual relationships for edge points and planar points. Finally, the optimization problem is again solved using the LevenbergMarquadt method, which has been adapted to robust fitting and \bar{Q}_k is finally registered on to the map. To ensure even distribution of the registered points each time a new scan is merged to the map, the point cloud is down-sized by using Voxel-Grid filters.

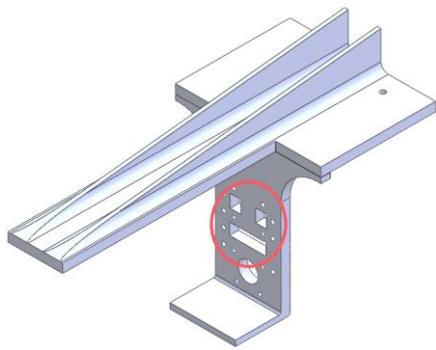


Fig. 4. Solidworks model of the LiDAR holder



Fig. 5. Intel NUC without shell

The edge points and planar points are filtered through Voxel-Grid filter to a different sized voxel. The edge points use 5 cm \times 5 cm \times 5 cm sized voxels, the planar points use 10 cm \times 10 cm \times 10 cm. In the final map, the entire map is resized to a 500 m \times 500 m \times 500 m region to ensure low memory usage.

III. UAV HARDWARE SETUP

To realize the 3D mapping algorithm discussed in the last section, a UAV platform is specifically designed and constructed to include the mentioned sensors together with a powerful processor. In this section, the hardware setup on the proposed UAV will be described in detail.

A Rotating 2-D LiDAR

The model of 2-D LiDAR sensor used is the Hokuyo UTM30EW. The Solidworks 1:1 model of the sensor is developed with actual dimension and weight for the visualization and assembly of the UAV (see Fig. 1).

A mounting bracket for the sensor was milled out of aluminum to hold the sensor in place (see Fig. 2). The mount was designed to withstand strong torque as the LiDAR sensor will rotate back and forth in

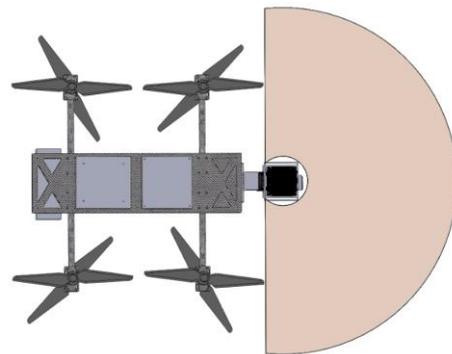


Fig. 6. LiDAR scanning range

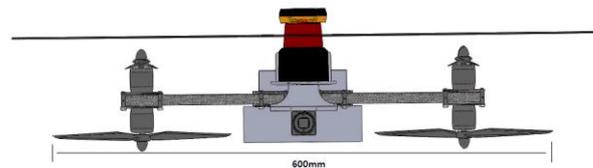


Fig. 7. Largest dimension of UAV

the proposed system.

Dynamixel AX-18a, a high torque digital servo with built in servo controller and position encoder was selected for this research work (see Fig. 3). It eliminates the need for external additional circuitry, thus providing a good power to weight ratio to the system. The highest possible torque (1.75 Nm) provided by the servo is more than sufficient to handle the load caused by the rotating sensor.

A mount for the servo was designed and fabricated to hold the servo motor (see Fig. 4). All the holes and recesses with exact dimension for the wiring and connectors are indicated in the Solidworks design. To provide extra rigidity and strength to the structure of the mount, two raised

columns at the base of the mount were added. Finally, the model was 3D printed with ABS plastic for actual implementation.

B On-board Intel Processor

The NUC from Intel was selected as the primary on-board computer to run the LOAM algorithm. It was programmed in Robot Operating System (ROS) for real-time application of the UAV. Intel's NUC has fast multi-thread performance, and it is running full quad-core chips in a light weight package. In addition, the outer shell of the processor was removed to further reduce unwanted weight. The bare NUC (see Fig. 5) will be hard-mounted on the UAV to ensure no unwanted contact of the circuit board to any conductive materials.

C Multirotor UAV Platform

In order to provide sufficient clearance to the front of the sensor, a H frame design for the UAV is adopted, rather than the conventional X frame UAV as shown in most literature such as [16] –

[18]. This design taken the consideration of the arms of the UAV to be swept back and parallel to the 180 degree field of view of the LiDAR sensor as shown in Fig. 6.

In this proposed design, the point-clouds to be generated would have a full 180 degree of horizontal field-of-view (FOV), to ensure a more accurate mapping of the system.

As seen from Fig. 7, the entire UAV is able to be packaged to a minimal of 600 mm wide, making it ideal size to pass through door ways and corridors in any indoor environment. This factor enables the designed UAV to be used extensively for indoor navigation and mapping.

The downside of this design is the challenge of keeping the center of gravity of the UAV close to its geometrical center. By keeping the UAV center of gravity close to the geometrical center, it will prevent imbalance in the load which might cause drastic dynamic changes of the UAV systems. In the

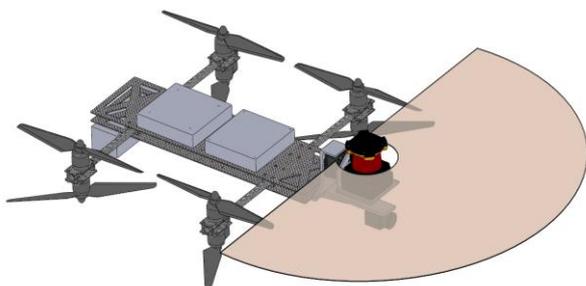


Fig. 8. Solidworks impression of the complete UAV

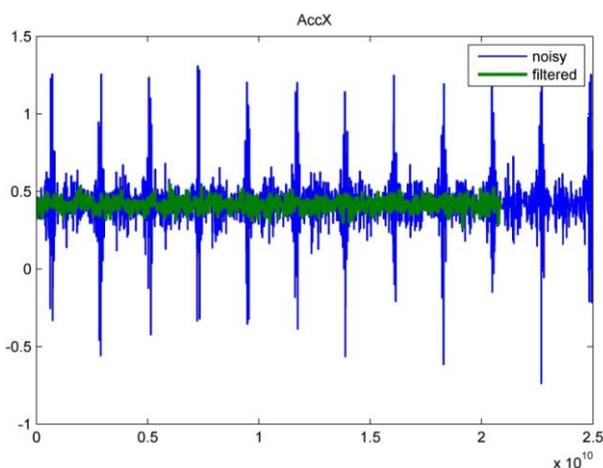


Fig. 9. Filtered acceleration on x-axis

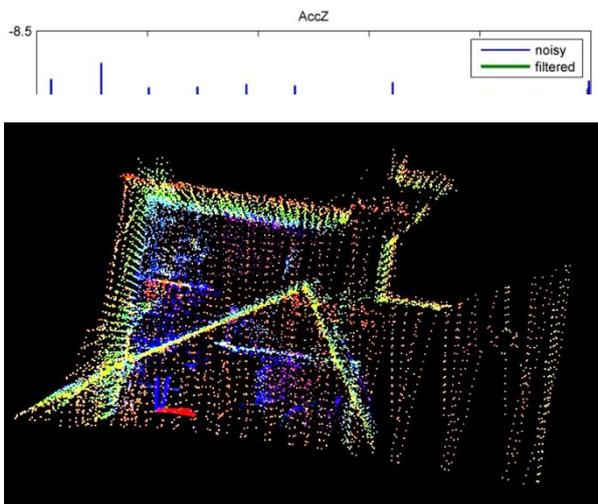


Fig. 11. Distortion due to sharp heading turn

proposed design, as substantial portion of the weight are from the sensor that is mounted in front of the UAV, therefore, the remaining components such as the processor and the battery had to be placed behind the UAV, to compensate the shift in the center of gravity. As a result, the final design can be visualized in Fig. 8. Once the design has proven to be feasible from Solidworks, a prototype of the UAV is fabricated to the exact scale as designed.

IV. IMPROVEMENT ON LOAM

The LOAM algorithm implemented to the Intel NUC was able to generate a high fidelity and undistorted map at slow speeds, but struggled to achieve good results for UAV flying speeds over 0.5 m/s due to the slow update rate of the algorithm. One way to improve the system is to incorporate IMU measurements to the algorithm. The IMU selected in the setup is a Pixhawk from 3DRobotics which is a high resolution IMU capable of publishing the orientation and acceleration data at high frequency (150 Hz). This data is utilized in two main ways: (i) The orientation data is used to align the point cloud perceived over one sweep to the initial position of the LiDAR in that sweep; and (ii) The acceleration data is used to partially remove the distortion from the mapping process.

There was one last hurdle before the IMU data could be incorporated, the hardware setup of our system introduced a very unique problem. The rotations of the LiDAR, specifically at the moment of direction change from

clockwise to counter-clockwise, induced vibration based noise into the accelerometer readings, which could not be dampened by hardware solutions. As such a High Pass Filter was designed and implemented in real-time on the on-board computer to filter out the low frequency (1 Hz, i.e. the speed of the LiDAR rotations) noise caused by vibrations. The filtering results can be seen in Fig. 9 and Fig. 10. The improvement on the LOAM accuracy especially when it is used on UAVs can be seen in Fig. 11 and Fig. 12 which shows the before and after implementation of the proposed IMU compensator described above. In both cases, the UAV is

commanded to make a sharp heading change of approximately 1 rad/s inside a closed room.



V. RESULTS

To verify the feasibility of the algorithm to be on actual UAV, several flight tests were performed. In the tests, the UAV was commanded to take-off, navigate to various part of the building, then back to the take-off point to land directly on it. This will effectively act as a closed-loop to allow ground truth data to be obtained.

In Fig. 13, a snapshot of the map generated by the improved algorithm within the interior of the building is generated. The minor distortions seen in the map are due to the movement of blinds on the windows, because of the downwash from the props. This movement shows up as uneven wall surfaces, instead of smooth flat planes. For comparison purpose, Fig. 14 shows a picture of the implemented UAV flying in the indoor building as the same instant of the generated map shown in Fig. 13. High similarity of the map can be observed.

Overall, after flying for approximately 100 meter distance navigated through multiple corridor of the building, the map of the entire floor was generated in real time and displayed live on the ground control station via a laptop. The map generated can be seen in Fig. 15. The graphical results have verified the improved algorithm to work well on the actual UAV in real time, using an Intel NUC i7 processor embedded on-board the UAV.

VI. CONCLUSION

The improved IMU integrated LOAM algorithm was tested in various environments and showed that it was able to handle 2 m/s velocity maneuvers and 1 rad/s turns. From the maps generated, it can be

concluded that the algorithm is capable of good performance, but is bottle necked by the sensors and data given to it. The ultimate bottle neck for the algorithm is the rotating 2D LiDAR itself. Which is extremely heavy along with the servos and the control board for the servo which adds more weight to the entire setup, drastically reducing flight endurance. The way the 2D LiDAR setup works fundamentally limits the amount of data that can be generated within a second, i.e., the rate of data published cannot be pushed beyond the physical limit set by the turning of Hokuyo UTM30EW. This indirectly sets a limit on how fast the pose estimates can be generated, hence ultimately limiting our ground velocity to a ceiling of 1 to 2 m/s.

VII. REFERENCES

- [1] R. C. Voorhies, "Efficient slam for scanning lidar sensors using combined plane and point features," Ph.D. dissertation, University of Southern California, 2015.
- [2] A. Harrison and P. Newman, "High quality 3d laser ranging under general vehicle motion," in *Robotics and Automation*, 2008. IEEE International Conference on. IEEE, 2008, pp. 7–12.
- [3] S. Grzonka, G. Grisetti, and W. Burgard, "Towards a navigation system for autonomous indoor flying," in *Robotics and Automation*, 2009. IEEE International Conference on. IEEE, 2009, pp. 2878–2883.
- [4] A. Bachrach, S. Prentice, R. He, and N. Roy, "Range-robust autonomous navigation in gps-denied environments," *Journal of Field Robotics*, vol. 28, no. 5, pp. 644–666, 2011.
- [5] M. Bosse, R. Zlot, and P. Flicke, "Zebedee: Design of a spring-mounted 3-d range sensor with application to mobile mapping," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1104–1119, 2012.
- [6] S. K. Phang, S. Z. Ahmed, and M. R. Abdul Hamid, "Design, dynamics modelling and control of a H-shape multi-rotor system for indoor navigation," in *2019 1st International Conference on Unmanned Vehicle Systems-Oman*, IEEE, 2019, pp. 1–6.
- [7] L. Kaul, R. Zlot, and M. Bosse, "Continuous-time three-dimensional mapping for micro aerial vehicles with a passively actuated rotating laser scanner," *Journal of Field Robotics*, vol. 33, no. 1, pp. 103–132, 2016.
- [8] M. Bosse and R. Zlot, "Continuous 3d scan-matching with a spinning 2d laser," in *Robotics and Automation*, 2009. IEEE International Conference on. IEEE, 2009, pp. 4312–4319.
- [9] H. Dong and T. D. Barfoot, "Lighting-invariant visual odometry using lidar intensity imagery and pose interpolation," in *Field and Service Robotics*. Springer, 2014, pp. 327–342.
- [10] S. Anderson and T. D. Barfoot, "Ransac for motion-distorted 3d visual sensors," in *Intelligent Robots and Systems (IROS)*, 2013 IEEE/RSJ International Conference on. IEEE, 2013, pp. 2093–2099.
- [11] C. H. Tong and T. D. Barfoot, "Gaussian process gauss-newton for 3d laser-based visual odometry," in *Robotics and Automation (ICRA)*, 2013 IEEE International Conference on. IEEE, 2013, pp. 5204–5211.
- [12] J. Zhang and S. Singh, "Low-drift and real-time lidarodometry and mapping," *Autonomous Robots*, vol. 41, no. 2, pp. 401–416, 2017.
- [13] J. Zhang and S. Singh, "Loam: Lidarodometry and mapping in real-time," in *Robotics: Science and Systems*, 2014, p. 9.
- [14] M. Bosse and R. Zlot, "Place recognition using keypoint voting in large 3d lidar datasets," in *Robotics and Automation (ICRA)*, 2013 IEEE International Conference on. IEEE, 2013, pp. 2677–2684.
- [15] R. Zlot and M. Bosse, "Efficient large-scale 3d mobile mapping and surface reconstruction of an underground mine," in *Field and service robotics*. Springer, 2014, pp. 479–493.
- [16] J. Cui, S. K. Phang, K. Ang, F. Wang, X. Dong, Y. Ke, S. Lai, K. Li, X. Li, F. Lin, J. Lin, P. Liu, T. Pang, B. Wang, K. Wang, Z. Yang and B. M. Chen, "Drones for cooperative search and rescue in post-disaster situation," in *Cybernetics and Intelligent Systems, Robotics, Automation and Mechatronics*, 2015, pp. 167–174.
- [17] J. Cui, S. K. Phang, K. Ang, F. Wang, X. Dong, Y. Ke, S. Lai, K. Li, X. Li, J. Lin, and P. Liu, "Search and rescue using multiple drones in post-disaster situation," *Unmanned Systems*, 2016, vol. 4, no. 1, pp. 83–96.
- [18] X. Chen, S. K. Phang, M. Shan, and B. M. Chen, "System integration of a vision-guided UAV for autonomous landing on moving platform," in *12th*

IEEE International Conference on Control and Automation, IEEE, 2016, pp. 761–766.

AUTHORS PROFILE



Swee King Phang received his B.Eng. degree in Electrical Engineering from the National University of Singapore (NUS), Singapore, in 2010 and then his Ph.D. degree in the same university in 2014.

He did his Post-doctorate in Temasek Laboratories at the NUS from 2014 to 2018, and he is currently a Lecturer with School of Engineering, Taylor's University, Subang Jaya, Malaysia. His research interests include system modeling of unmanned aerial systems (UAS), flight controller design of the UAS, indoor navigation of the UAS, trajectory optimization and path planning, and UAS applications.

Dr. Phang has published more than 40 peer reviewed articles in the field of UAS, including a book chapter in Handbook of Unmanned Aerial Vehicles, 10 articles in indexed journals, and multiple international conferences. He is the recipient of Guan Zhao-Zhi Award in the 2014 Chinese Control Conference held in Nanjing, China. He is also the finalist of Best Paper Award in the 2018 IEEE International Conference on Control and Automation held in Anchorage, Alaska. He has won numerous awards in International UAS competitions, such as the winner for 2017 and 2014 International Micro Aerial Vehicle Competition that was held in Toulouse, France and Delft, the Netherlands, respectively; Semi-finalist for Drones for Good Competition in Dubai, UAE in 2015; Runners-up in the 2013 International UAV Innovation Grand Prix held in Beijing, China; and Top 9 Finalist of DARPA UAVForge Challenge held in Atlanta, US in 2012.

He is a Young Professional member of IEEE since 2009.



Syed Zeeshan Ahmed received his B.Eng (Hons) in Electrical Engineering from BITS Pilani Dubai, U.A.E in 2016 and M.Sc in Electrical Engineering (specialization in Automation and Control) from National University of Singapore, Singapore in 2018.

He was the Team lead for the student Robotics and UAS research group at BITS Pilani, Dubai from 2014 to 2016. He has participated in multiple international UAS competitions, such as the International Aerial Robotics Competition (IARC 2015) held in Georgia, USA; Top 10

Semi-Finalist in the Drones for Good Competition (DFG 2016) held in Dubai, UAE. He has also published multiple papers in both international and regional conferences.

He is currently a Research Engineer with Institute for Infocomm Research in A*Star, Singapore. His research interests include localization and mapping for autonomous vehicles, sensor fusion, LiDAR based perception and control system design for unmanned aerial systems (UAS).



Mohamed Redhwan Abdul Hamid graduated with a B.Sc (Hons) in Aircraft Engineering in 2014 from Kingston University, London, UK. He currently works in the Intelligent Unmanned System group in Temasek Laboratories @ the National University of Singapore developing novel unmanned aerial vehicle solutions to unique applications.

He has published and presented a few conference proceedings across a wide range of subjects and was shortlisted as a finalist for the Best Paper Award in the 2018 IEEE International Conference on Control and Automation. His research interests include the design of aircraft structures, advanced manufacturing techniques and CAE analysis. He has a background in the commercial aviation industry with training towards a Category B1.1 Licensed Aircraft Engineer in Singapore and he is an approved UAV Pilot under the Civil Aviation Authority of Singapore for Temasek Laboratories.