

# Adaptive Lidar Scan Frame Integration: Tracking Known MAVs in 3D Point Clouds

Li Qingqing\*, Yu Xianjia\*, Jorge Peña Queralta\*, Tomi Westerlund\*

\*Turku Intelligent Embedded and Robotic Systems (TIERS) Lab, University of Turku, Finland.

Emails: <sup>1</sup>{qingqli, xianjia.yu, jopequ, toveve}@utu.fi

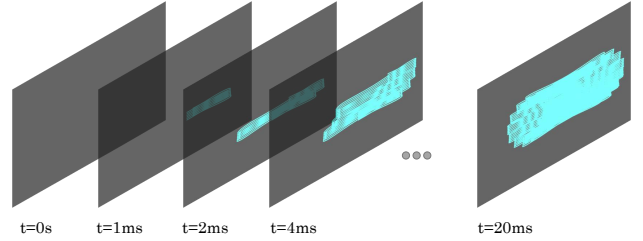
**Abstract**—Micro-aerial vehicles (MAVs) are becoming ubiquitous across multiple industries and application domains. Lightweight MAVs with only an onboard flight controller and a minimal sensor suite (e.g., IMU, vision, and vertical ranging sensors) have potential as mobile and easily deployable sensing platforms. When deployed from a ground robot, a key parameter is a relative localization between the ground robot and the MAV. This paper proposes a novel method for tracking MAVs in lidar point clouds. In lidar point clouds, we consider the speed and distance of the MAV to actively adapt the lidar's frame integration time and, in essence, the density and size of the point cloud to be processed. We show that this method enables more persistent and robust tracking when the speed of the MAV or its distance to the tracking sensor changes. In addition, we propose a multi-modal tracking method that relies on high-frequency scans for accurate state estimation, lower-frequency scans for robust and persistent tracking, and sub-Hz processing for trajectory and object identification. These three integration and processing modalities allow for an overall accurate and robust MAV tracking while ensuring the object being tracked meets shape and size constraints.

**Index Terms**—Micro-aerial vehicles, MAV, UAV, UGV, detection, tracking, lidar detection, lidar tracking, adaptive scanning.

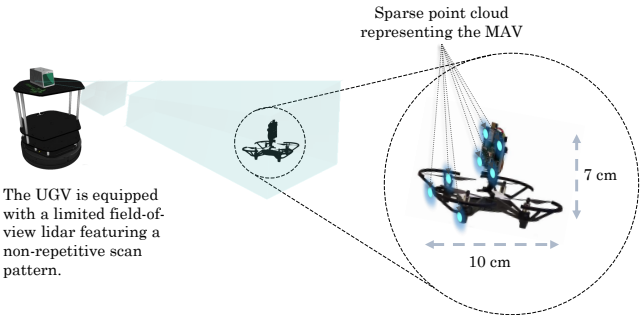
## I. INTRODUCTION

Micro-aerial vehicles (MAVs) have seen an increasing adoption across a variety of application domains in recent years [1]. Multiple works have been devoted to the navigation of MAVs in GNSS-denied environments [2], and state estimation in both single [3] and multi-MAV systems [4]. In this paper, we are particularly interested in tracking and state estimation from an external system, for those applications where MAVs are deployed together with or from unmanned ground vehicles (UGVs) [5], [6].

From the perspective of deployment within multi-robot systems, being able to track MAVs from UGVs enables miniaturization and higher degrees of flexibility lowering the need for high-accuracy onboard localization. A recent and significant example of multi-robot system deployment in GNSS-denied environments is the DARPA Subterranean challenge [7], [8]. Reports from participating teams indicate that localization and collaborative sensing were among the key challenges, with MAVs being deployed from UGVs dynamically during the challenge. Since MAVs often rely on visual-inertial odometry (VIO) for self and relative estate estimation [9], relying on external lidar-based tracking can also extend the operability



(a) Illustration of the field of view (FoV) coverage with different point cloud integration times in a non-repetitive lidar scanning device.



(b) Illustration of a ground robot tracking a micro-aerial vehicle (MAV) using a limited-FoV solid-state lidar.

Fig. 1: Conceptual illustration of the field-of-view coverage with different integration times on a Livox Horizon lidar (top) and its application to tracking MAVs (bottom).

to low-visibility or other domains where VIO has inherent limitations [10], [11].

Tracking and detecting MAVs has been a topic of interest for researchers in recent years. First, owing to the increasing need of identifying and detecting foreign objects or drones in areas with controlled airspace such as airports [12], [13]. Second, to optimize the utilization of MAVs as flexible mobile sensing platforms [14]. This paper focuses on the latter use. Compared to the existing literature, which relies mainly on vision-based techniques [15], we provide a lidar-based solution that can be utilized more independently of the environmental conditions. Until recently, most 3D lidars provided relatively sparse point clouds in terms of object recognition [16], with limited vertical resolution in inexpensive devices. However, solid-state lidars have recently emerged as state-of-the-art in terms of long-range scanners featuring high-density point clouds [17]. The

main caveat is the limited field of view (FoV) in most of these devices [18], but solutions include utilizing multiple lidars or correspondingly adjusting the position and orientation of the robot base where the lidar is installed.

We are particularly interested in the problem of tracking a MAV that is deployed from a ground robot. We assume thus that the initial position of the MAV after take-off is known. We also assume that its shape and size are known a priori. We develop methods targeting solid-state lidars owing to the higher density of the resulting point cloud even with more limited FoV. Moreover, in these lidars, the concept of a frame or scan frequency changes considerably. Similarly, as in rotating 3D lidars, a frame in solid-state lidars can be naturally related to a single revolution. With non-repetitive scan patterns, lidars can output point clouds at adjustable frequencies with varying FoV coverage, as illustrated in Fig. 1a. This opens the door to new lidar perception methods that exploit the possibilities of adaptively adjust the frame integration time to better sense the objects. To the best of our knowledge, this approach has not been previously studied. We apply the proposed adaptive lidar scan integration methods within the problem of a UGV tracking a MAV for external state estimation, as conceptualized in Fig. 1b. While our focus is on MAVs, the proposed methods can also be easily adapted to detect foreign objects or intruder MAVs more accurately. We first put our focus on single and known MAV detection, but present generic methods that can be extended to multi-MAV tracking as long as FoV limitations are accounted for.

The main contribution of this paper is twofold. We first introduce a novel adaptive lidar scan integration method enabling more accurate and reliable object recognition and tracking from 3D point clouds, specifically applied to MAV detection. In addition, we then define a multi-modal tracking system that relies on processing point clouds resulting in different integration times for higher accuracy and persistent tracking, while validating the trajectories using a priori known information about the MAV dimensions.

The remaining of this paper is organized as follows. In section II, we review the state-of-the-art in MAV detection, lidar-based object detection and tracking, and a handful of existing works on the vision-, radar- and lidar-based MAV detection and tracking. Section III then formulates the adaptive scanning method, and how it applies to a MAV detection and tracking problem. Section IV reports on our methodology, and Section V describes experimental results with different settings. Finally, we conclude this work and outline future research directions in Section VI.

## II. RELATED WORKS

This section reviews the literature in the areas of detection and tracking of MAVs. Owing to the scarcity of works devoted to lidar-based MAV tracking, we have focused on: (i) the state-of-the-art in MAV detection, mostly vision-based; (ii) lidar-based detection and tracking of small objects; and (iii) detection of MAVs based on lidar or radar point cloud data.

### A. Vision-based MAV Detection

Most of the work to data in tracking small objects and MAVs has been related to vision-based approaches [19], [20], [15]. Vision-based approaches can be classified among those that rely on passive or active visual markers, and those that detect and track objects in general, e.g., with traditional computer vision or deep learning. In the former category, [20] provides an example of tracking based on passive artificial visual markers, which can be used to calculate the relative 3D position of the MAV from a camera. On a different direction aimed at MAV-to-MAV detection, Walter et al. presented UVDAR, an ultra-violet (UV) solution for relative localization in multi-MAV systems [21].

Regarding the latter category, the development of deep convolutional neural networks (CNNs) in recent years has facilitated the adoption within the domain of object detection and tracking. Arguably, a significant portion of the state-of-the-art in tracking is based on deep learning methods [22]. These methods often offer significantly higher degrees of accuracy and robustness. For instance, Vrba et al. have presented a marker-less system for relative localization of MAVs [15], which can be applied to detecting foreign or intruder MAVs.

The potential of depth cameras for detecting MAVs has also been showcased in the literature. For instance, deep learning models processing depth maps have been applied to tracking a MAV and aiding it in navigating and avoiding obstacles [23].

While depth cameras can provide accurate location and size measurements, and vision sensors, in general, are able of robust tracking and relative positioning, our focus in this paper is to work with lidars owing to their flexibility in terms of environmental conditions, and because of their significantly higher range and accuracy when compared to depth cameras.

### B. Lidar-based object tracking

More in line with the research presented in this paper is point-cloud-based tracking. While this generally refers to lidar point clouds, some of the work in the literature is also devoted to point clouds generated by stereo or depth cameras, or radars. In general terms, traditional approaches to tracking in point cloud data rely mostly on distance-based clustering [24].

Nonetheless, significant work has been carried out in the area of deep learning voxel-based methods for segmentation and detection of objects in 3D point clouds. For instance, VoxelNet [25] implements a voxel features extractor (VFE) on point cloud to characterize object points. Other networks have been proposed that directly process point sets, such as PointNet [26] and PointNet++ [27], to fully exploit the inherent information in the point cloud data for object tracking. Iterating over these, works such as [28] have proposed end-to-end and point-to-box networks for 3D object tracking.

When considering small objects, the specific literature is more scarce. In [16], Razlaw et al. focus on detecting people in sparse point clouds from multi-channel rotating 3D lidars. Compared with this approach, we focus on exploiting the adaptive frame integration capabilities of solid-state lidars to optimize the point cloud density and do not necessarily assume sparsity.

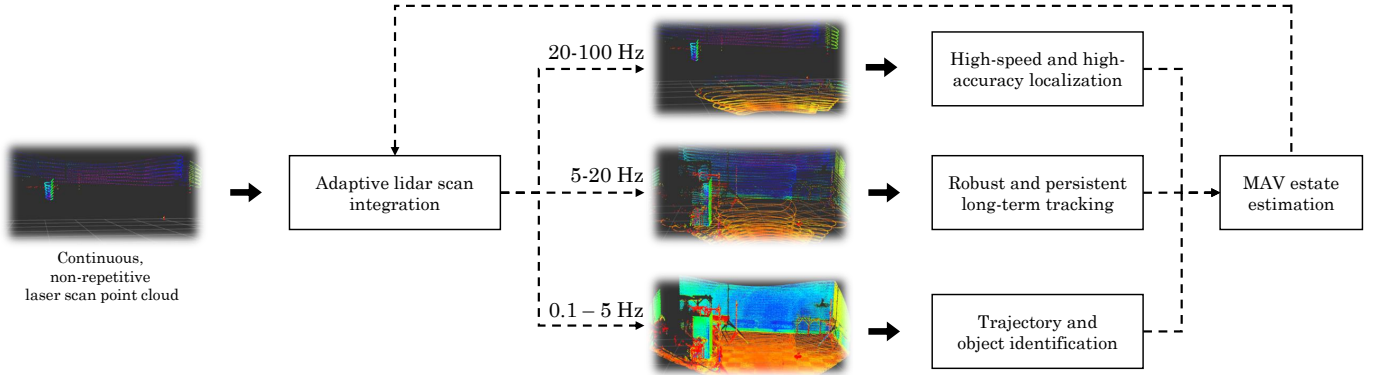


Fig. 2: Overview of the proposed methods, where tracking is simultaneously performed at three different scan frequencies. Within each of these three threads, the scan frame integration is adjusted based on the distance to the target MAV and its speed.

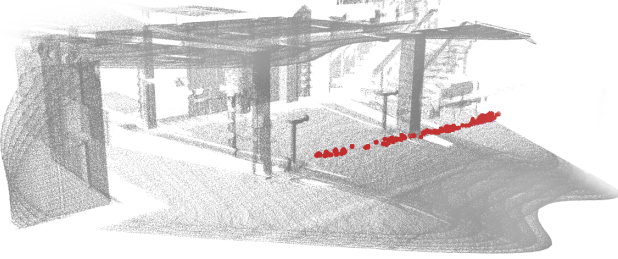


Fig. 3: Integration trajectory recovery example

### C. Lidar and radar-based MAV detection and tracking

When the focus is more on detection rather than on accurate localization or tracking of the detected MAV, radar has been proven a robust solution [29]. Lidars, in any case, have been identified as having big potential for MAV detection and tracking [30].

In summary, while object detection and tracking in point clouds is a relatively mature field, we have found a gap in the literature in terms of optimizing the way these point clouds are generated. In particular, we see most of the current work being focused on processing point clouds, while our objective is to study how we can enhance the performance of a given tracking algorithm by improving the quality of the point cloud data it is fed with. Our focus here is on actively adapting lidar-based perception for detecting and tracking a flying MAV, where the density and size of the point cloud is optimized based on, e.g., the MAVs distance to the lidar sensor, or its speed.

## III. PROBLEM DEFINITION

We consider the problem of tracking a MAV from a ground robot. The ultimate objective is, e.g., to improve the collaboration between the robots and the ability of the MAV to navigate in complex environments aided by the UGV. The rest of this paper delves into the definition, design, and implementation of methods for tracking a single MAV. Nonetheless, these can be extended to multiple MAVs. The main limitation when tracking multiple units is the FoV of the lidar sensors onboard the ground vehicle, and therefore assumptions have to be made to the spatial distribution of the MAVs (always within the FoV of the ground robot). Alternatively, more lidar scanners can be installed to increase the FoV.

### A. Rationale

The majority of 3D laser scanners available to date are multi-channel, rotating lidars. While devices with 64 or 128 vertical channels can provide high angular resolution in both horizontal and vertical dimensions, these high-end devices are not the most common. Moreover, the scanning pattern is in general repetitive, which has benefited from a geometric perspective in terms of data processing but does not enable a higher FoV coverage with longer exposure if the position of the sensor is fixed. New solid-state lidars featuring non-repetitive scan patterns, albeit having more limited FoV, can provide more dense point clouds and often feature longer detection ranges. In particular, we are interested in the possibilities of dynamically adjusting the FoV coverage and density in the point cloud to be processed for detection and tracking. Among the benefits of these new lidars and the possibilities of adaptive scanning rates is also higher resilience against one of the challenges in lidar-based perception: motion-induced distortion [31]. In general, the literature targeting tracking of MAVs using lidar scanners is scarce, and existing methods in point cloud object detection and tracking considering mainly static frames. We aim to define more optimal settings for generating point clouds based on the state (speed and distance to the sensor) of the MAV being tracked.

### B. System Overview

We propose three simultaneous tracking modalities with three processes analyzing point cloud frames resulting in integration times ranging several orders of magnitude. A general view of the multi-modal tracking processes is shown in Fig. 2. In more detail, the three modalities are described below:

- (i) Adaptive high-frequency tracking. In this first process, sparse point clouds are integrated at frequencies up to 100 Hz. The MAV is only trackable through a reduced number of points, but we are able to estimate its position and speed with high accuracy. In this process, the MAV is not necessarily recognizable in all processed frames.
- (ii) Adaptive medium-frequency tracking. The second process operates at frequencies within the range of typical lidar scanners (i.e., 5 to 20 Hz). The frequency within that

---

**Algorithm 1:** MAV tracking with adaptive scan integration

---

**Input:**

High- and medium-freq int. rates:  $\{I_{HF}^{k-1}, I_{MF}^{k-1}\}$   
3D lidar point clouds:  $\{\mathcal{P}_k(I_{HF}^{k-1}), \mathcal{P}_k(I_{MF}^{k-1})\}$   
Last known MAV state:  $(\mathbf{p}_{MAV}^{k-1}, \dot{\mathbf{p}}_{MAV}^{k-1})$

**Output:**

MAV state:  $\{\mathbf{p}_{MAV}^k, \dot{\mathbf{p}}_{MAV}^k\}$   
UGV control:  $\dot{\mathbf{q}}_{UGV}^k$   
Int. rates:  $\{I_{HF}^k, I_{MF}^k\}$

**Function** *object\_extraction* ( $\mathcal{P}, I, \mathbf{p}_{MAV}^{k-1}, \dot{\mathbf{p}}_{MAV}^{k-1}$ ):

Ground removal:  $\mathcal{P}' \leftarrow \mathcal{P}$ ;  
Generate KD Tree:  $kdtree \leftarrow \mathcal{P}'$ ;  
MAV pos estimation:  $\hat{\mathbf{p}}_{MAV}^k \leftarrow \mathbf{p}_{MAV}^{k-1} + \frac{\dot{\mathbf{p}}_{MAV}^{k-1}}{I}$ ;  
MAV points:  $\mathcal{P}_{MAV}^k = KNN(kdtree, \hat{\mathbf{p}}_{MAV}^k)$ ;  
MAV state estimation:  $\hat{\mathbf{p}}_{MAV}^k = \frac{1}{|\mathcal{P}_{MAV}^k|} \sum_{p \in \mathcal{P}_{MAV}^k} p$ ;  
**return**  $\mathbf{p}_{MAV}^k$ ;

// Coarse but persistent tracking

**while** new  $\mathcal{P}_k(I_{MF}^k)$  **do**

$\mathbf{p}_{MAV}^k =$   
    *object\_extraction*( $\mathcal{P}_k(I_{MF}^k), I_{MF}^k, \mathbf{p}_{MAV}^{k-1}, \dot{\mathbf{p}}_{MAV}^{k-1}$ );

// Fine-grained estimation

**while** new  $\mathcal{P}_k(I_{HF}^k)$  **do**

$\mathbf{p}_{MAV}^{k''} =$   
    *object\_extraction*( $\mathcal{P}_k(I_{HF}^k), I_{HF}^k, \mathbf{p}_{MAV}^{k-1}, \dot{\mathbf{p}}_{MAV}^{k-1}$ );  
 $\mathbf{p}_{MAV}^k, \dot{\mathbf{p}}_{MAV}^k \leftarrow estimate(\mathbf{p}_{MAV}^{k''}, \mathbf{p}_{MAV}^k)$ ;

$\{I_{HF}^k, I_{MF}^k\} \leftarrow adjust\_integration\_freqs(\mathbf{p}_{MAV}^k, \dot{\mathbf{p}}_{MAV}^k)$ ;

$\dot{\mathbf{q}}_{UGV}^k \leftarrow keep\_within\_FoV(\mathbf{p}_{MAV}^k, \dot{\mathbf{p}}_{MAV}^k)$ ;

---

same range is dynamically adjusted to optimize the density of the point cloud. At these frequencies, the extracted point cloud representing the MAV is distorted by motion, and thus the localization and speed estimation accuracy is lower. However, this process enables more robust and persistent tracking as the MAV can be recognized in most if not all frames.

- (iii) Low-frequency trajectory and object validation. The third and last process that runs in parallel to the previous two performs long-term tracking and validates the reconstructed trajectory of the MAV based on predefined dimensional constraints. An illustration of such trajectory reconstruction is shown in Fig. 3

### C. Formulation

Let  $\mathcal{P}_k(I_r^k)$  be the point cloud generated by the lidar with an integration time  $I_r^k$ , and let  $\mathbf{s}_{UGV}^k = \{\mathbf{q}_{UGV}^k, \dot{\mathbf{q}}_{UGV}^k\}$  be the position and speed defining the state of the UGV at time  $k$ . We also denote by  $\mathbf{s}_{MAV}^k = \{\mathbf{p}_{MAV}^k, \dot{\mathbf{p}}_{MAV}^k\}$  the position and speed of the MAV. We use discrete steps represented by  $k$  owing to the discrete nature of the set of consecutive point clouds. The output of the main tracking algorithm is to extract from  $\mathcal{P}_k(I_r^k)$  the set of points representing the MAV, which we denote by  $\mathcal{P}_{MAV}^k$ , and to adjust the integration time for the next point cloud,  $I_{HF}^k, I_{MF}^k$ .

---

**Algorithm 2:** Trajectory validation

---

**Input:**

Low-freq int. rate:  $I_{LF}^{k-1}$   
3D lidar point cloud:  $\mathcal{P}_k(I_{LF}^{k-1})$   
MAV state history:  $(\mathbf{p}_{MAV}, \dot{\mathbf{p}}_{MAV})$

**Output:** Trajectory validation (*bool*)**while** new  $\mathcal{P}_k(I_{LF}^{k-1})$  **do**

    // Generate cubic splines  
    // with position and speed constraints  
     $\{B_i\} \leftarrow \{\mathbf{p}_{MAV}, \dot{\mathbf{p}}_{MAV}\}$ ;  
    // Estimate expected point cloud from  
    // known density at given distance and speed  
     $\hat{\mathcal{P}}_k \leftarrow \{\{B_i\}, \mathbf{p}_{MAV}, \dot{\mathbf{p}}_{MAV}\}$ ;  
    // Calculate IoU  
     $IoU = calc\_IoU(\mathcal{P}_k(I_{LF}^{k-1}), \hat{\mathcal{P}}_k)$ ;  
    **if**  $IoU > th$  **then**  
        | **return** True  
    **else**  
        | **return** False

---

### D. Adaptive scan integration

Since we assume that the state of the MAV ( $\mathbf{p}_{MAV}^{k-1}, \dot{\mathbf{p}}_{MAV}^{k-1}$ ) is initially known, the point cloud processing proceeds as follows. First, we perform ground removal based on the known position of the UGV and the last-known altitude of the MAV. We then proceed with finding the nearest neighbor points to a predicted MAV position. This step is repeated for both the high and medium frequency scans, the former one providing a more accurate position estimation while the latter is more persistent in time. Finally, these two estimations are combined, and the results are utilized to adjust the integration rates based on the point cloud density expected for the given distance and speed. The UGV is also controlled to maintain the MAV within the FoV of its lidar. This process is outlined in Algorithm 1.

1) *Trajectory validation*: The main purpose of the low-frequency scan stream is to validate the extracted MAV's trajectory. While the tracking with adaptive scan integration only takes into account the MAV size roughly in terms of distance within which nearest neighbors are looked for, the extracted point cloud is not validated against its known dimensions. This is done when enough points are accumulated into a reconstructed trajectory. As exposed in Algorithm 2, we first perform a cubic spline interpolation based on the history of estimated positions and speeds. To calculate the parameters of the cubic spline, we utilize constraints on the first derivative based on the speed, rather than forcing the first and second derivative to be continuous. Indeed, the acceleration of the MAV can suddenly change. Based on predetermined values of point cloud density as a function of the MAV's distance to the lidar and its speed, we then produce an expected point cloud. We validate the original point cloud given a threshold for the IoU measure with the generated estimate.

## IV. METHODOLOGY

### A. Experimental platforms

The experimental platforms consist on a single ground robot and a commercially available Ryze Tello MAV. The ground robot is an EAI Dashgo platform equipped with a Livox

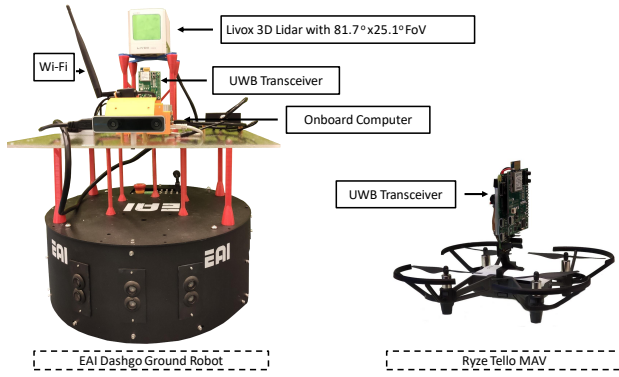


Fig. 4: Ground robot and MAV utilized in the experiments.

Horizon lidar ( $81.7^\circ \times 25.1^\circ$  FoV). The lidar is able to output scanned pointcloud up to 100 Hz, featuring a non-repetitive pattern. A pair of ultra-wideband (UWB) transceivers is used to obtain a single range between the robot and the MAV at frequencies ranging from 10 Hz to 100 Hz. The UWB ranging is only used in aiding the manual validation of the extracted trajectory in places where there was no external positioning system. In the future, it could be incorporated as part of the tracking algorithm as well, as is becoming increasingly adopted in multi-robot systems [32], [33].

### B. Software

The system has been implemented using ROS Melodic under Ubuntu 18.04. The algorithms are running in the main computer onboard the ground robot. The computer runs the robot's driver<sup>1</sup>, the Tello MAV driver<sup>2</sup>, the Livox lidar driver<sup>3</sup>, and our open-source MAV tracking package<sup>4</sup>. The latter is a multi-threaded node able to process the different point clouds in real time. The point cloud library (PCL) [34] is utilized to extract the position of the MAV from the lidar's point cloud.

### C. Metrics

Owing to the lack of an accurate external positioning system such as a motion capture system, our focus is instead on measuring the performance of the tracking at different scan integration rates and manually validating the overall trajectory. The experimental flights are carried out in large indoor halls with multiple columns and objects, as shown in Fig. 3. Another set of experiments is carried out in a small flying area where an external UWB positioning system was available and used to fly the MAV over a predefined trajectory. A characterization on the accuracy of such system can be found in [35].

## V. EXPERIMENTAL RESULTS

In this section we report on the experimental results. The experimental results consist mainly on flights in two different indoor environments and different conditions.

### A. Adaptive scan integration

The first objective of our experiments was to assess the tracking performance at different scan frequencies in order to better model the adaptiveness of our algorithm. In order to adapt the scanning frequency to optimize the tracking performance, key parameters are the point cloud density at different distances and the reliability of the detections at different speeds.

The point cloud density for different scanning frequencies as a function of the distance between the lidar and the MAV is shown in Fig. 5. This measure refers only to the density of the points representing the MAV and not the overall density including the rest of the scene. The darker lines represent the average point cloud density, while the band with higher transparency represents the values within the standard deviation. The size of the Tello MAV is about 500 cubic centimeters. Based on our experiments, reliable tracking at high speeds can be achieved with at least 4 points, while we require at least 20 points at medium scanning frequency. This, however, only applies in free space. As can be seen in Fig. 7, significant noise appears in the point cloud between the MAV and walls in the environment when flying nearby. We discuss further this issue at the end of this section.

In terms of the tracking performance based on the speed, we plot in Fig. 6 the distance between consecutive detections at different scanning frequencies. The results in this particular figure cannot be directly utilized to model the adaptive nature of our tracking algorithm. Nonetheless, they can be leveraged to better understand what are the speed limits under which given scanning frequencies do not provide the expected distance between detection that can be inferred from the MAV speed and the scan frequency.

The results included in Fig. 5 and Fig. 6 have been obtained flying the MAV in a long, straight corridor with a length of about 35 m. The MAV was flying mostly in straight lines and the speed was estimated using both visual odometry and the position history extracted from the lidar data in a partially manual manner.

### B. Qualitative trajectory validation

In order to validate the performance of the tracking algorithm and better understand the limitations of our tracking approach at different scanning frequencies, we compare two different types of trajectories. Owing to the lack of a system to obtain ground truth (e.g., a motion capture system), we provide qualitative analysis for one of the trajectories and compare it with a UWB positioning system in the other one.

First, we test the tracking algorithm through a trajectory where the MAV flies in a large open area at distances from 2 m to over 17 m far from the lidar scanner and variable speeds. In this scenario, the analysis is mostly qualitative, with the trajectories shown in Fig. 8. However, the UWB ranging data and the lidar data has been both manually confirmed, so the maximum positioning error along the track is at worst around 20 cm. Qualitatively, the main results from this experiment are the ability of the tracking algorithm to keep track of the

<sup>1</sup><https://github.com/TIERS/dashgo-d1-ros>

<sup>2</sup><https://github.com/TIERS/tello-driver-ros>

<sup>3</sup>[https://github.com/Livox-SDK/livox\\_ros\\_driver](https://github.com/Livox-SDK/livox_ros_driver)

<sup>4</sup><https://github.com/TIERS/adaptive-lidar-tracking>



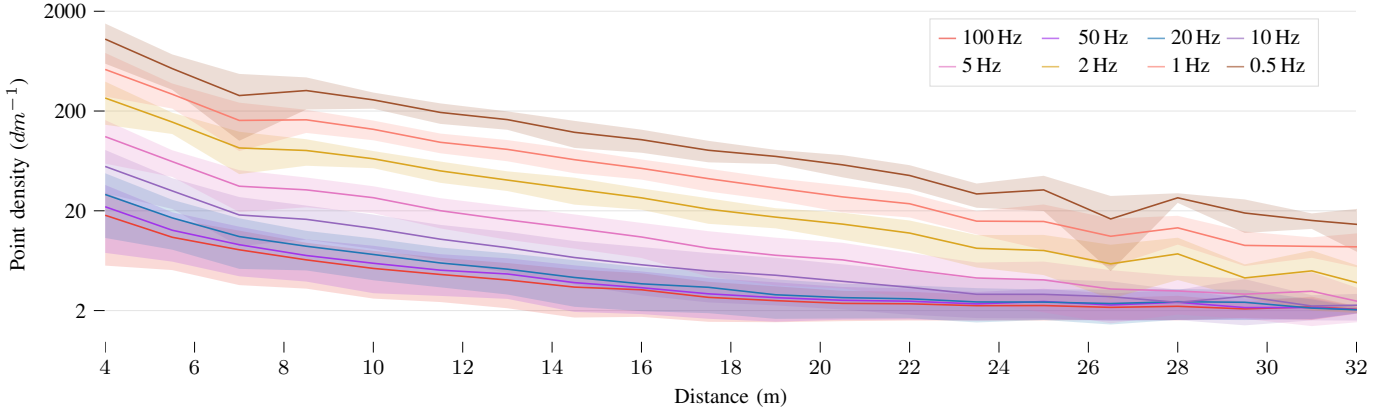


Fig. 5: Density of the point cloud representing the MAV based on the distance to the lidar scanner and the scanning frequency.

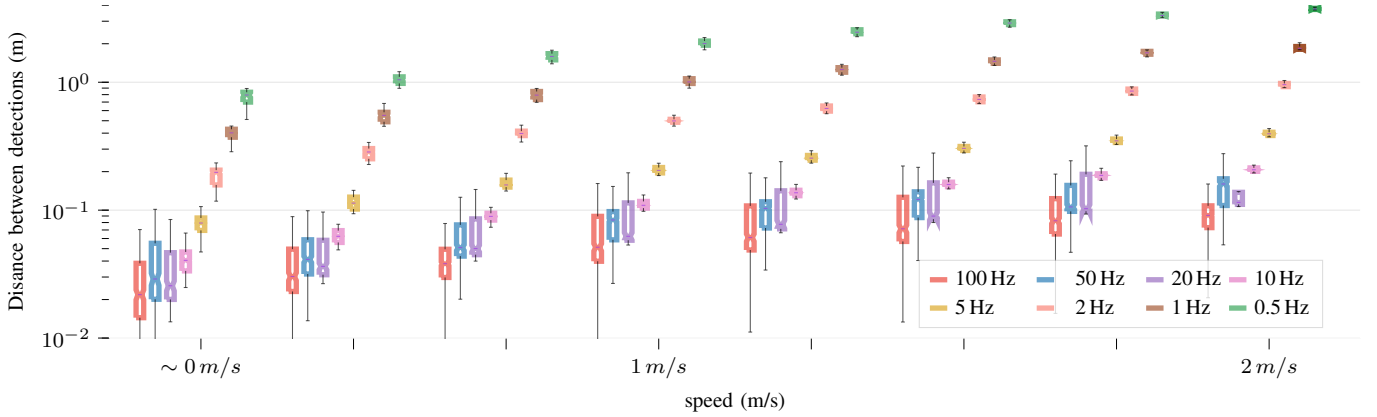


Fig. 6: Distance between consecutive MAV detections based on its speed and the lidar's scanning frequency.

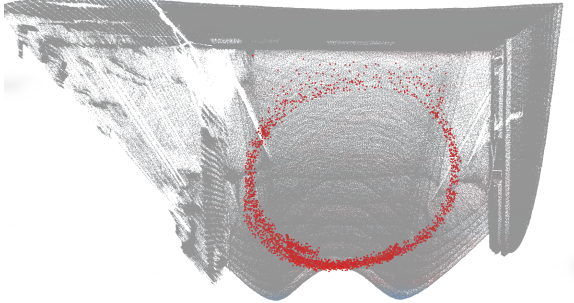


Fig. 7: Accumulated point cloud for the circular trajectory.

MAV over changes in speed, direction, and at longer distances. The figure only shows frequencies equal to or above 5 Hz because at lower scanning frequencies the speed estimation was highly inaccurate during the early stages of the flight. We can see that only at the highest frequency we are able to track the MAV along the completed trajectory, while the trajectory itself is noisier. The higher level of error when estimating the MAV position is due to a lower number of points being detected, which can correspond to different parts of the MAV in consecutive scans. The last subplot shows the overall estimated trajectory where our algorithm has combined the different scanning frequencies to obtain the smoothness of the medium frequencies and the performance of the higher

frequencies. The trajectory also employs the cubic spline interpolation from the validation algorithm.

Second, we perform a continuous flight with a predefined circular trajectory in a small flying arena where the UWB positioning system is available. The results for this flight are shown in Fig. 9. The leftmost plot shows the reference position. However, it is worth noticing that the accuracy of the lidar, of around 2 cm for distances smaller than 20 m, is higher than the average accuracy of 10 to 15 cm in the UWB positioning system. Therefore, the trajectory is mere as a reference and only a qualitative discussion is possible with these results. In any case, owing to the continuous change in the speed of the MAV, which is a prior unknown to the tracking algorithm, again only at frequencies equal or over 5 Hz are we able to track the MAV. Nonetheless, at 5 Hz the tracking stops before the fourth revolution is completed, and persistent tracking is only possible when higher frequencies are taken into account.

### C. Discussion

We have shown in this section qualitative results that show the performance of the adaptive tracking algorithm and the same approach applied only to specific scanning frequencies. From both sets of experiments, the main conclusion is that the adaptive approach is able to accommodate a wider variety of

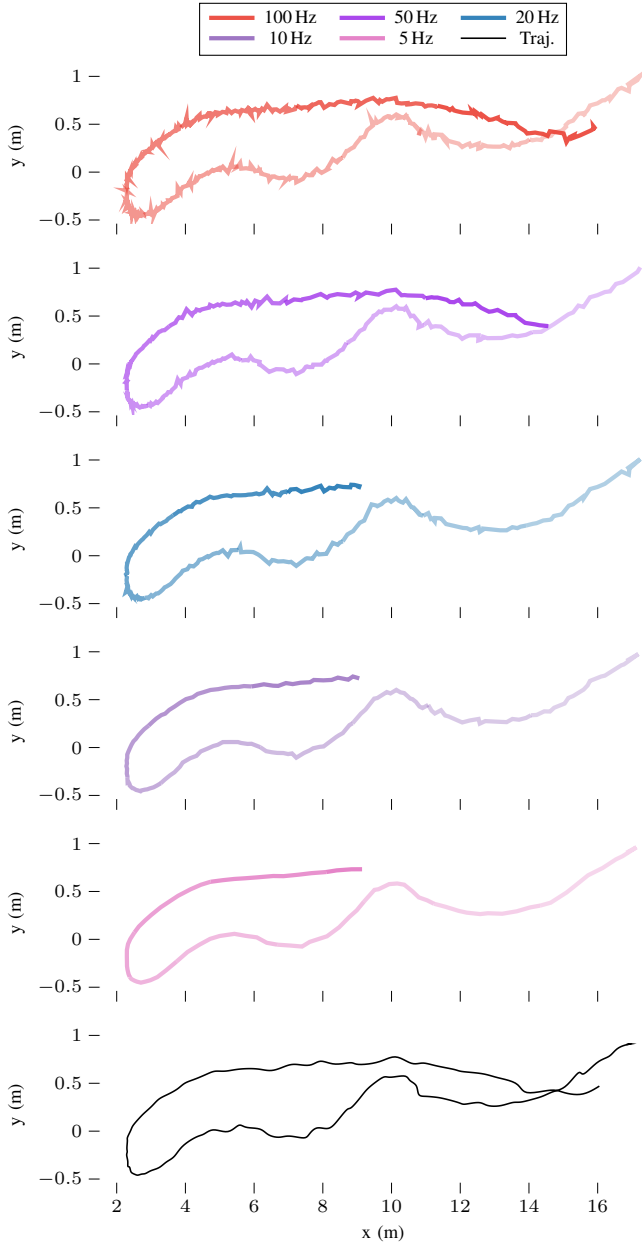


Fig. 8: Estimated trajectories at different frequencies and with adaptive approach (top five plots), and trajectory estimated from our algorithm (bottom plot).

scenarios. We have been able to put together the flexibility of high-speed tracking with the robustness of medium frequencies, avoiding the frequent errors of the former, and the lower tracking capacity of the latter is more challenging conditions.

One key limitation when tracking MAVs, as visualized in the circular trajectory experiments, is the low density of the point cloud and the inability to tell the difference between the MAV's points and lidar noise. This is also due to the low reflectivity of the MAV, and there is thus the potential for mitigation with more reflective surfaces that could aid in separating the sparse MAV point cloud from the lidar noise originated due to near objects. As we can see in Fig. 7, the point cloud density near the rear wall is very sparse in some

areas, therefore being unable to reconstruct a robust trajectory as there are multiple options available that would meet the dynamics and dimensional constraints of the MAV.

## VI. CONCLUSION

We have presented a set of methods for detecting and tracking MAVs that are deployed from ground robots, assuming that the initial position is known. The focus has been on the introduction of a novel adaptive lidar scan integration method that enables more accurate MAV localization with high-frequency scans, robust and persistent tracking with longer frame integration times, and trajectory validation with low-frequency analysis. Experimental results from different settings confirm the better suitability of the different integration times for different scenarios or MAV behaviour, with our adaptive tracking being able to consistently track a MAV in places where a constant lidar scan frequency cannot. Finally, with an additional method to validate the trajectory based on the known shape and size of the MAV, we are able to confirm that the object being tracked meets the dimensional constraints.

In future works, we will explore the integration of lidar-based tracking into the navigation of the MAV, and the integration of onboard state estimation at the MAV into the tracking algorithm.

## ACKNOWLEDGMENT

This research work is supported by the Academy of Finland's AutoSOS project (Grant No. 328755) and Finnish Foundation for Technology Promotion.

## REFERENCES

- [1] M. Blösch, S. Weiss, D. Scaramuzza, and R. Siegwart, "Vision based mav navigation in unknown and unstructured environments," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 21–28.
- [2] 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*, vol. 84, no. 1, pp. 199–216, 2016.
- [3] Y. Song, S. Nuske, and S. Scherer, "A multi-sensor fusion mav state estimation from long-range stereo, imu, gps and barometric sensors," *Sensors*, vol. 17, no. 1, p. 11, 2017.
- [4] H. Xu, L. Wang, Y. Zhang, K. Qiu, and S. Shen, "Decentralized visual-inertial-uwv fusion for relative state estimation of aerial swarm," *arXiv preprint arXiv:2003.05138*, 2020.
- [5] A. Gawel, Y. Lin, T. Koutros, R. Siegwart, and C. Cadena, "Aerial-ground collaborative sensing: Third-person view for teleoperation," in *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2018, pp. 1–7.
- [6] J. Peña Queralta, J. Taipalmaa, B. C. Pullinen, V. K. Sarker, T. N. Gia, H. Tenhunen, M. Gabbouj, J. Raitoharju, and T. Westerlund, "Collaborative multi-robot search and rescue: Planning, coordination, perception and active vision," *IEEE Access*, pp. 1–1, 2020.
- [7] T. Rouček, M. Pecka, P. Čížek, T. Petříček, J. Bayer, V. Šalanský, D. Heřt, M. Petrлік, T. Bába, V. Spurný *et al.*, "Darpa subterranean challenge: Multi-robotic exploration of underground environments," in *International Conference on Modelling and Simulation for Autonomous Systems*. Springer, 2019, pp. 274–290.
- [8] M. Petrлік, T. Bába, D. Heřt, M. Vrba, T. Krajník, and M. Saska, "A robust uav system for operations in a constrained environment," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 2, pp. 2169–2176, 2020.
- [9] T. Nguyen, K. Mohta, C. J. Taylor, and V. Kumar, "Vision-based multi-mav localization with anonymous relative measurements using coupled probabilistic data association filter," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 3349–3355.

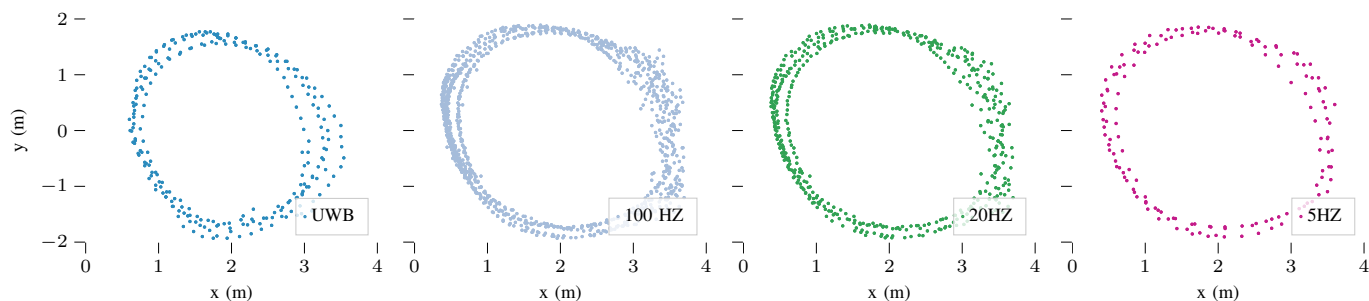


Fig. 9: Reference trajectory (UWB) and estimated positions at different fixed frequencies.

- [10] J. Peña Queralta, L. Qingqing, F. Schiano, and T. Westerlund, "Vio-uw-based collaborative localization and dense scene reconstruction within heterogeneous multi-robot systems," *arXiv preprint arXiv:2011.00830*, 2020.
- [11] L. Qingqing *et al.*, "Offloading Monocular Visual Odometry with Edge Computing: Optimizing Image Compression Ratios in Multi-Robot Systems," in *The 5th ICSCC*. IEEE, 2019.
- [12] I. Guvenc, F. Koohifar, S. Singh, M. L. Sichertiu, and D. Matolak, "Detection, tracking, and interdiction for amateur drones," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 75–81, 2018.
- [13] S. Hengy, M. Laurenzis, S. Schertzer, A. Hommes, F. Kloeppel, A. Shoykhetbrod, T. Geibig, W. Johannes, O. Rassy, and F. Christnacher, "Multimodal uav detection: study of various intrusion scenarios," in *Electro-Optical Remote Sensing XI*, vol. 10434. International Society for Optics and Photonics, 2017, p. 104340P.
- [14] J. Peña Queralta, J. Raitoharju, T. N. Gia, N. Passalis, and T. Westerlund, "Autos: Towards multi-uav systems supporting maritime search and rescue with lightweight ai and edge computing," *arXiv preprint arXiv:2005.03409*, 2020.
- [15] M. Vrba and M. Saska, "Marker-less micro aerial vehicle detection and localization using convolutional neural networks," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 2, pp. 2459–2466, 2020.
- [16] J. Razlaw, J. Quenzel, and S. Behnke, "Detection and tracking of small objects in sparse 3d laser range data," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 2967–2973.
- [17] K. Li, M. Li, and U. D. Hanebeck, "Towards high-performance solid-state-lidar-inertial odometry and mapping," *arXiv preprint arXiv:2010.13150*, 2020.
- [18] J. Lin and F. Zhang, "Loam livox: A fast, robust, high-precision LiDAR odometry and mapping package for LiDARs of small FoV," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 3126–3131.
- [19] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for uav tracking," in *European conference on computer vision*. Springer, 2016, pp. 445–461.
- [20] P. H. Nguyen, K. W. Kim, Y. W. Lee, and K. R. Park, "Remote marker-based tracking for uav landing using visible-light camera sensor," *Sensors*, vol. 17, no. 9, p. 1987, 2017.
- [21] V. Walter, N. Staub, A. Franchi, and M. Saska, "Uvdar system for visual relative localization with application to leader-follower formations of multirotor uavs," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 3, pp. 2637–2644, 2019.
- [22] P. Li, D. Wang, L. Wang, and H. Lu, "Deep visual tracking: Review and experimental comparison," *Pattern Recognition*, vol. 76, pp. 323–338, 2018.
- [23] A. Carrio, S. Vemprala, A. Ripoll, S. Saripalli, and P. Campoy, "Drone detection using depth maps," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1034–1037.
- [24] A. Rangesh and M. M. Trivedi, "No blind spots: Full-surround multi-object tracking for autonomous vehicles using cameras and lidars," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 4, pp. 588–599, 2019.
- [25] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4490–4499.
- [26] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.
- [27] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *arXiv preprint arXiv:1706.02413*, 2017.
- [28] H. Qi, C. Feng, Z. Cao, F. Zhao, and Y. Xiao, "P2b: Point-to-box network for 3d object tracking in point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6329–6338.
- [29] G. Fang, J. Yi, X. Wan, Y. Liu, and H. Ke, "Experimental research of multistatic passive radar with a single antenna for drone detection," *IEEE Access*, vol. 6, pp. 33 542–33 551, 2018.
- [30] M. Hammer, M. Hebel, B. Borgmann, M. Laurenzis, and M. Arens, "Potential of lidar sensors for the detection of uavs," in *Laser Radar Technology and Applications XXIII*, vol. 10636. International Society for Optics and Photonics, 2018, p. 1063605.
- [31] F. Neuhaus, T. Koß, R. Kohnen, and D. Paulus, "Mc2slam: Real-time inertial lidar odometry using two-scan motion compensation," in *German Conference on Pattern Recognition*. Springer, 2018, pp. 60–72.
- [32] W. Shule, C. M. Almansa, J. Peña Queralta, Z. Zou, and T. Westerlund, "Uwb-based localization for multi-uav systems and collaborative heterogeneous multi-robot systems: a survey," *arXiv preprint arXiv:2004.08174*, 2020.
- [33] C. M. Almansa, W. Shule, J. Peña Queralta, and T. Westerlund, "Autocalibration of a mobile uwb localization system for ad-hoc multi-robot deployments in gnss-denied environments," *arXiv preprint arXiv:2004.06762*, 2020.
- [34] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 1–4.
- [35] J. Peña Queralta, C. M. Almansa, F. Schiano, D. Floreano, and T. Westerlund, "Uwb-based system for uav localization in gnss-denied environments: Characterization and dataset," *arXiv preprint arXiv:2003.04380*, 2020.