UAV Tracking with Lidar as a Camera Sensor in GNSS-Denied Environments

Ha Sier*†, Xianjia Yu†, Iacopo Catalano†, Jorge Peña Queralta†, Zhuo Zou*, Tomi Westerlund†

*School of Information Science and Technology, Fudan Universtiy, China
†Turku Intelligent Embedded and Robotic Systems (TIERS) Lab, University of Turku, Finland.
Emails: ^{1}{seha20, zhuo}@fudan.edu.cn, ^{2}{xianjia.yu, imcata, jopequ, tovewe}@utu.fi

Abstract*—Light detection and ranging (LiDAR) sensor has become one of the primary sensors in robotics and autonomous system for high-accuracy situational awareness. In recent years, multi-modal LiDAR systems emerged, and among them, LiDAR-as-a-camera sensors provide not only 3D point clouds but also fixed-resolution 360°panoramic images by encoding either depth, reflectivity, or near-infrared light in the image pixels. This potentially brings computer vision capabilities on top of the potential of LiDAR itself. In this paper, we are specifically interested in utilizing LiDARs and LiDAR-generated images for tracking Unmanned Aerial Vehicles (UAVs) in real-time which can benefit applications including docking, remote identification, or counter-UAV systems, among others. This is, to the best of our knowledge, the first work that explores the possibility of fusing the images and point cloud generated by a single LiDAR sensor to track a UAV without a priori known initialized position. We trained a custom YOLOv5 model for detecting UAVs based on the panoramic images collected in an indoor experiment arena with a motion capture (MOCAP) system. By integrating with the point cloud, we are able to continuously provide the position of the UAV. Our experiment demonstrated the effectiveness of the proposed UAV tracking approach compared with methods based only on point clouds or images. Additionally, we evaluated the real-time performance of our approach on the Nvidia Jetson Nano, a popular mobile computing platform.

***Index Terms*—UAV; LiDAR-as-a-camera; drone tracking; LiDAR; LiDAR detection; LiDAR tracking; YOLOV5;**

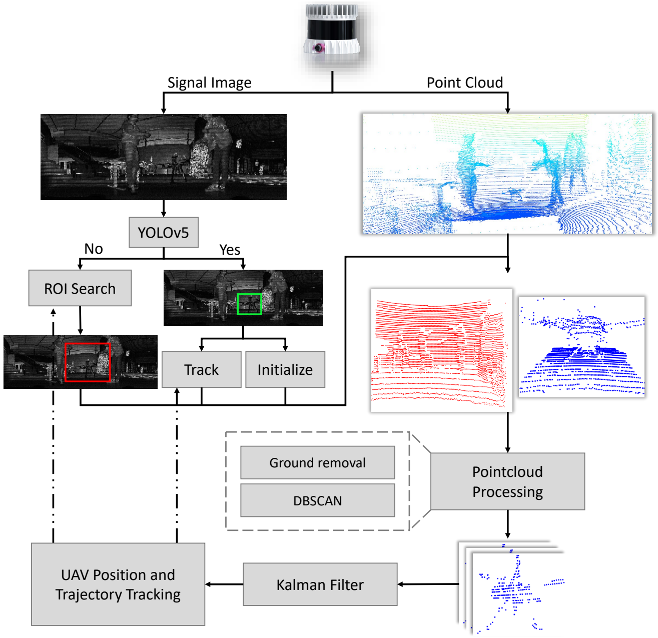


Fig. 1: Diagram of proposed UAV tracking system based on the image and point cloud generated by an Ouster LiDAR.

I. Introduction

Achieving high-level situational awareness is a vital but not trivial task in the domain of mobile robotics and autonomous systems in general, as it enables effective decision-making, navigation, and control in complex environments [1]. In recent years, researchers have exploited diverse sensor modalities to enhance robotics perception systems. Among these, LiDAR and cameras have emerged as the primary perception components [2]. In contrast to cameras, LiDARs offer key features, including long-range and precise geometry data, and generally greater resilience to adverse weather conditions, such as fog and rain. Currently, there are various modalities of LiDARs available [3]. Ouster LiDARs [4], [5], [6] stand out by offering dense point clouds and 360°field of view with low-resolution images, obtained by encoding depth, reflectivity, or near-infrared light in the image pixels. These are so-called lidar as a camera sensors [4], [6]. Other lidar manufacturers are expected to provide similar functionality in the near future.

This type of LiDAR as a camera sensor is inherently compatible with deep learning (DL) models for vision sensors, obviating the need for external camera mounting and calibration [6]. Moreover, the LiDAR-as-camera approach can augment conventional LiDAR-based methods for object detection and tracking, as the algorithms for the latter are more computationally intensive and less mature than vision sensors. Despite the low vertical resolution and lack of color, the key motivation for considering such an approach is simply to do more with already existing data and without additional sensors.

We are interested in studying the potential of these sensors for tracking unmanned aerial vehicles (UAVs), fusing both lidar point cloud data and signal images. Given that the images generated by the LiDAR as a camera sensors have low resolution (see for example in figure 4), we are particularly interested in short-range tracking of UAVs, which

979-8-3503-2308-5/23/$31.00

finds potential applications in UAV docking, or collaboration, among others. The deployment of UAVs requires accurate localization, especially in cases where GNSS signals suffer from degradation or are not available [7].

In this paper, we propose a UAV tracking approach based on the integration of images and 3D point clouds generated by an Ouster LiDAR sensor. The diagram of the approach we follow is illustrated in figure 1. The UAV can be detected in signal images instead of manually giving its initial position as it is needed in other point-cloud-only approaches. The detection also yields an approximate region of interest (ROI) in the point cloud, which will be expanded if no detection occurs. This approach reduces computation overhead by avoiding the need for an overall point cloud search. UAV identification is achieved by clustering points within the ROI, followed by continuous position estimation using the Kalman Filter (KF).

The remainder of this document is organized as follows. Section II introduces the current state-of-the-art UAV tracking methods relevant to the presented approach. Section III introduces the methodology and the experimental setup with results presented in Section IV. Section V concludes the work and outlines future research directions.

II. Related Work

This section reviews the literature in the field of UAV detection and tracking. Due to the limited research on tracking small objects such as UAVs based on LiDAR, we focus on: (i) vision-based and (ii) LiDAR-based UAV tracking; (iii) research into the potential of LiDAR-as-a-camera sensors; and (iv) applications of UAV tracking.

A. UAV tracking with cameras

Vision-based methods are widely used to track small objects and UAVs [8], [9], [10]. They can be divided into two categories: 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. For example, [9] introduces a trinocular system with ground-based cameras to control a rotary-wing UAV in real time based on its key features. Alternatively, [11] presents an infrared binocular vision system with PTU and exosensors to track drones cheaply under any weather and time conditions based on their infrared spectra. Recent developments in deep convolutional neural networks (CNNs) have boosted adoption in the field of object detection and tracking. Arguably, a large part of the state-of-the-art in tracking is based on deep learning methods. Recent advances in deep CNNs have improved object detection and tracking performance [12]. For instance, [10] proposes a CNN-based markerless UAV relative positioning system that allows the stable formation and autonomous interception of multiple UAVs.

Depth cameras can also detect UAVs and help them avoid obstacles using deep learning models that process depth maps [13]. While depth cameras can provide accurate position and size measurements, and vision sensors are generally capable of robust tracking and relative localization, our focus in this paper is on the use of Ouster LiDARs because of their flexibility with respect to environmental conditions and their ability to provides more accurate and informative signal images than depth cameras.

B. UAV tracking with LiDARs

While LiDAR systems are often employed for detecting and tracking objects, they pose unique challenges in detecting and tracking UAVs due to their small size, varied shapes and materials, high speed, and unpredictable movements.

When deployed from a ground robot, a crucial parameter is relative localization between different devices. Li et al. [14] suggest a new approach for tracking UAVs using LiDAR point clouds. They take into account the UAV speed and distance to adjust the LiDAR frame integration time, which affects the density and size of the point cloud to be processed.

By conducting a probabilistic analysis of detection and ensuring proper setup, as shown in [15], it is possible to achieve detection using fewer LiDAR beams, while performing continuous tracking only on a small number of hits. The limitations in the 3D LiDAR technology can be overcome by moving the sensor to increase the field of view and improve the coverage ratio. Additionally, combining a segmentation approach and a simple object model while leveraging temporal information in [16] has been shown to reduce parametrization effort and generalize to different settings.

Another approach, departing from the typical sequence of track-after-detect, is to leverage motion information by searching for minor 3D details in the 360° LiDAR scans of the scene. If these clues persist in consecutive scans, the probability of detecting a UAV increases. Furthermore, analyzing the trajectory of the tracked object enables the classification of UAVs and non-UAV objects by identifying typical movement patterns [17], [18].

C. LiDAR as a camera

The Ouster LiDAR blurs the traditional boundaries between cameras and LiDARs by providing 360°panoramic images, including depth, signal, and ambient images in the near-infrared spectrum, in addition to the point cloud output. These signal and ambient images capture the strength of laser light reflected back to the sensor and ambient light, respectively. Ouster has proven that training models based on these images are effective for various computer vision applications, such as object detection, segmentation, feature extraction, and optical flow [4], [5]. Several existing DL models have also been demonstrated to be effective in analyzing these images [6].

D. Applications of UAV tracking

Recently, researchers have shown interest in tracking and detecting UAVs due to two primary reasons: the rising demand for identifying and detecting foreign objects or drones in areas with controlled airspace, like airports [19], [20], and the potential for optimizing the utilization of UAVs as versatile mobile sensing platforms through tracking and detection [21].

The ability to track UAVs from unmanned ground vehicles (UGVs) allows for miniaturization and greater flexibility in

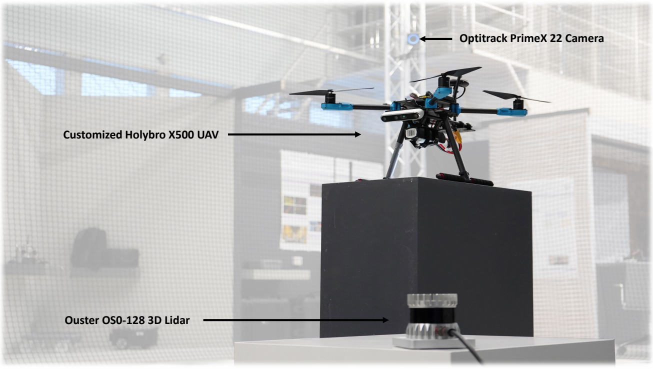


Fig. 2: Experimental hardware and site.

multi-robot systems, reducing the need for high-accuracy onboard localization. This was demonstrated in the DARPA Subterranean challenge [22], [23], where UAVs were dynamically deployed from UGVs in GNSS-denied environments. Localization and collaborative sensing were key challenges, with reports indicating that LiDAR-based tracking was useful in domains where visual-inertial odometry (VIO) has limitations, such as low-visibility situations [24], [25].

Similarly, tracking UAVs is crucial in the landing phase of the aerial system. Different methods using a ground-based stereo camera [26] or having the UAV carry an infrared camera to detect signals from the destination [27] have been proposed. As these works employ cameras as their main sensory system, they can be easily affected by background lighting conditions while in our approach we prefer a LiDAR which is more resilient in these environmental conditions.

III. Methodology

A. Hardware Information

The experimental setup consists of an Ouster OS0-128 LiDAR, an Intel computer, and a Holybro X500 V2 drone shown in Fig. 2. The Ouster OS0-128 boasts a wide field of view (360° × 90°) and is capable of producing both dense point cloud data and signal images at a frequency of 10 Hz. The drone is equipped with OptiTrack markers, allowing the acquisition of the drone's actual position at 100 Hz in the motion capture (MOCAP) system, which is also partially visible in Fig. 2.

B. Software Information

The system was implemented based on the ROS Noetic framework on Ubuntu 20.04 operating system. The tracking package, Ouster drivers, and OptiTrack mocap program were executed on the laptop computer connected to the Ouster LiDAR. Our tracking approach requires YOLOv5 [1] for UAV detection and Open3D [2] for point cloud data processing. The algorithms and code designed and developed for these experiments are written in Python and are publicly available in a GitHub repository [3].

C. Data collection

Regarding the data consisting of UAV detections with the Ouster LiDAR, we collected three different data sequences ($Seq \quad i, i \in (1 \sim 3)$). in an indoor area of $10.0\times10.0m^2$, with distances ranging 0.5 m to 8 m between the LiDAR and the UAV. The details of the collected data can be seen in Table I. *Seq 1* and *Seq 3* represent a helical ascension trajectory, while *Seq 2* represents an elliptical trajectory.

TABLE I: Data sequences collected in our experiment.

Sequences	Time (s)	Ground Truth	Trajectory	Distance (m)
Seq 1	35.8	Mocap	elliptical trajectory	7.0
Seq 2	26.9	Mocap	spiral trajectory	6.3
Seq 3	32.7	Mocap	spiral trajectory	8.0

D. UAV tracking fusing signal images and point clouds

This manuscript introduces an approach to tracking a UAV by fusing LiDAR signal images and point cloud data, both generated within the same sensor. The proposed methodology's overarching framework is illustrated in Fig. 1. It is worth noting that the Ouster OS0-128 LiDAR serves as the sole input source for the entire system, providing real-time UAV position outputs. The tracking procedure comprises two primary stages, namely:

1) Initialization of UAV position: Fig 3 shows the signal image and point cloud data of a UAV at its initial position. Notably, when the UAV approaches the Ouster LiDAR, the signal image of the UAV appears clearer. To detect the position of the UAV in the signal image and obtain the ROI in the image, the state-of-the-art object detection algorithm YOLOV5 is utilized. Given that the Ouster LiDAR signal image and the point cloud data are spatially linked, the corresponding point cloud ROI can be extracted. Subsequently, by employing ground removal and point cloud clustering techniques, the UAV point cloud can be extracted and, as such, the initial position of the UAV can be estimated.

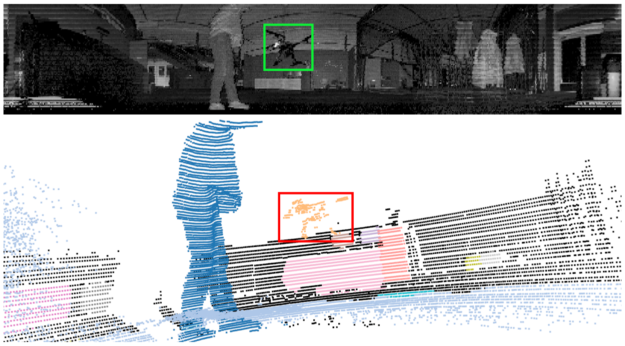


Fig. 3: Example of a signal image (top) and its corresponding point cloud with background removed (bottom).

[1] https://github.com/ultralytics/yolov5/releases

[2] http://www.open3d.org/

[3] https://github.com/TIERS/UAV-tracking-based-on-LiDAR-as-a-camera

Algorithm 1: UAV tracking fusing signal images and point clouds

```
Input:
  Raw pointcloud: $\mathcal{P}^t_{raw}$
  Signal image: $\mathcal{S}^t$
  Target UAV point cloud: $\mathcal{P}^{t-1}_{UAV}$
Output:
  Drone pose: $\mathbf{P}^t_{UAV}$;
Function object_extraction($\mathcal{P}^t_{raw}$, $\mathcal{P}^{t-1}_{UAV}$, $\mathcal{ROI}^t_{YOLO}$):
    if $\mathcal{ROI}^t_{YOLO}$ then
        $\mathcal{P}^t_{roi}$ = $\mathcal{P}^t_{raw}$ ($\mathcal{ROI}^t_{YOLO}$);
    else
        $\mathcal{ROI}^t \longleftarrow KF\left(get\_center\left(\mathcal{P}^{t-1}_{UAV}\right)\right)$;
    $\mathcal{P}^t \longleftarrow ground\_removal\left(\mathcal{P}^t_{ROI}\right)$ ;
    $\mathcal{P}^t_i \longleftarrow DBSCAN\left(\mathcal{P}^t\right),\ i \in (0, R)$;
    foreach $\mathcal{P} \in \mathcal{P}^t_i$ do
        if Min (num($\mathcal{P}$ ) - num($\mathcal{P}^{t-1}_{UAV}$) ) then
            if Min (dis($\mathcal{P}$) - dis($\mathcal{P}^{t-1}_{UAV}$)) then
                $flag = 1$ ;
                $\mathcal{P}^t_{uav} \longleftarrow \mathcal{P}$ ;
            else
                $flag = 0$;
        else
            $flag = 0$;
    return $\mathcal{P}^t_{uav}$, flag ;
foreach new $\mathcal{S}^t$ do
    $\mathcal{ROI}^t_{YOLO} \longleftarrow YOLOv5\left(\mathcal{S}^t\right)$;
    if $\mathcal{ROI}^t_{YOLO} =$ None then
        $\mathcal{P}^t_{UAV}$ , flag = object_extraction ($\mathcal{P}^t_{raw}$, $\mathcal{P}^{t-1}_{uav}$) ;
        if flag = 0 then
            $\mathbf{P}^t_{UAV}$ = KF_predict (get_center($\mathcal{P}^{t-1}_{UAV}$)) ;
            KF_update ($\mathbf{P}^t_{UAV}$)
        else
            $\mathbf{P}^t_{UAV}$ = get_center($\mathcal{P}^t_{UAV}$) ;
            KF_update ($\mathbf{P}^t_{UAV}$) ;
    else
        $\mathcal{P}^t_{UAV} = object\_extraction\left(\mathcal{P}^t_{raw}, \mathcal{P}^{t-1}_{UAV}, \mathcal{ROI}^t\right)$ ;
        $\mathbf{P}^t_{UAV} = get\_center\left(\mathcal{P}^t_{UAV}\right)$;
        $KF\_update\left(\mathbf{P}^t_{UAV}\right)$;
```

2) Fusion of signal image and point cloud data: Our goal for fusing the signal image and point cloud data is to acquire an accurate ROI. Initially, we perform image detection on each signal image to identify the UAV. This allows us to obtain a more precise ROI. Additionally, we can use the UAV's initial position from the previous step as a reference to extract the UAV point cloud from the environment based on the number of UAV point clouds and their distance. However, if YOLOv5 fails to detect the UAV, we select the ROI predicted by the Kalman filter and separate the UAV point cloud from it. This process is explicated in Algorithm 1.

E. Evaluation

To validate the precision of the UAV estimated poses and velocities by our approach, we calculate the absolute pose error (APE) and velocity error based on the ground truth from the MOCAP system. We conducted a comparative analysis of our proposed method with a UAV tracking method that solely relies on either Ouster LiDAR images or point clouds. The point cloud tracking method uses only Ouster OS0-128 LiDAR point cloud data as input, with a frame rate of 10Hz. When tracking the UAV using point cloud data, the initial position of the UAV needs to be known as the point cloud of the UAV is sparser than that of larger objects, such as cars or humans, and distinguishing the point cloud of the UAV from the environment using features is challenging. On the other hand, the image tracking method uses only Ouster OS0-128 LiDAR signal images with a frame rate of 10 Hz. Firstly, the signal image undergoes target detection processing to obtain the UAV's bounding box in the signal image. Subsequently, the image in the bounding box is converted into point cloud data. The point cloud clustering algorithm is then utilized to separate the UAV's point cloud from the environment based on the number and distance features of the point cloud clusters. This approach allows us to obtain the trajectory of the UAV. Both of these methods are estimated by the Kalman filter method to obtain the UAV's trajectory.

We conducted the experiments on two different platforms to assess real-time performance, the Lenovo Legion Y7000P equipped with 16GB RAM, 6-core Intel i5-9300H (2.40GHz) and Nvidia GTX 1660Ti (1536 CUDA cores, 6GB VRAM), as well as the commonly used embedded computing platform Jetson Nano with 4-core ARM A57 64-bit CPU (1.43GHz), 4GB RAM, and 128-core Maxwell GPU.

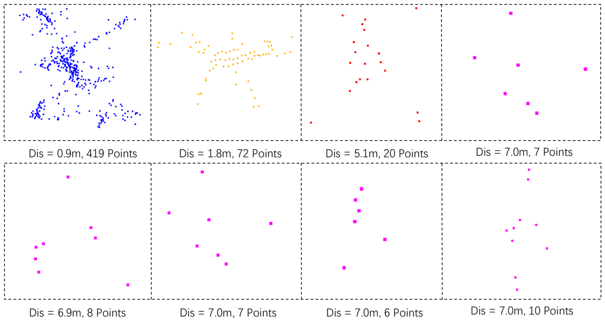


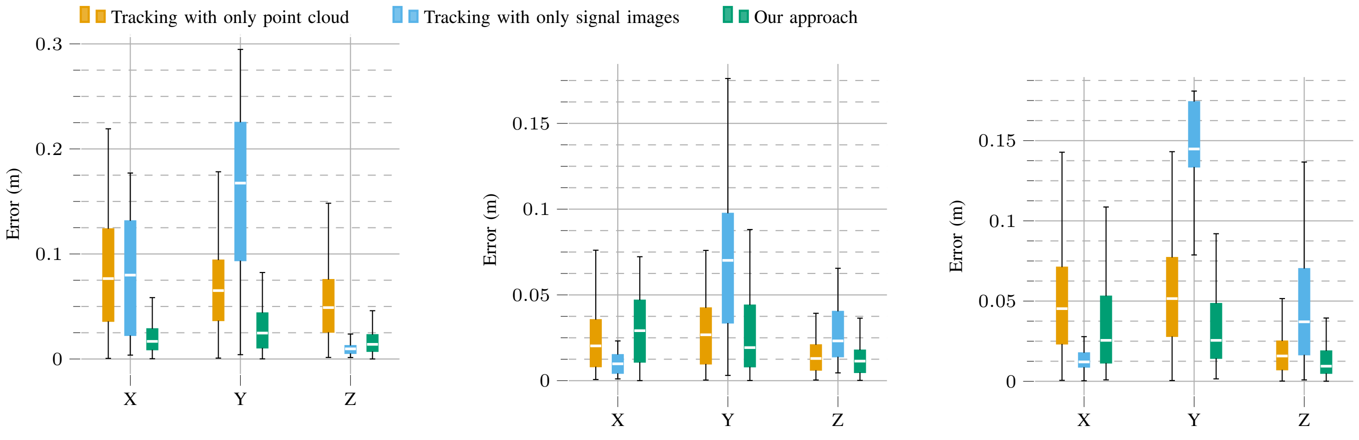
Fig. 4: UAV point cloud of drones at different distances, the bottom line shows UAV point cloud of four consecutive frames at the same long distance.

IV. Experimental Results

In this section, we report the experimental results, based on the three data sequences gathered in the indoor test environment.

A. UAV in the Ouster LiDAR point cloud

The first parameter to analyze is the number of points that reflect from the UAV at different distances. Our analysis, shown in Fig 4, reveals that the point cloud structure generated by the UAV is significantly influenced by the distance from the target. At short distances, the point cloud produced by LiDAR is abundant and presents comprehensive details. When the

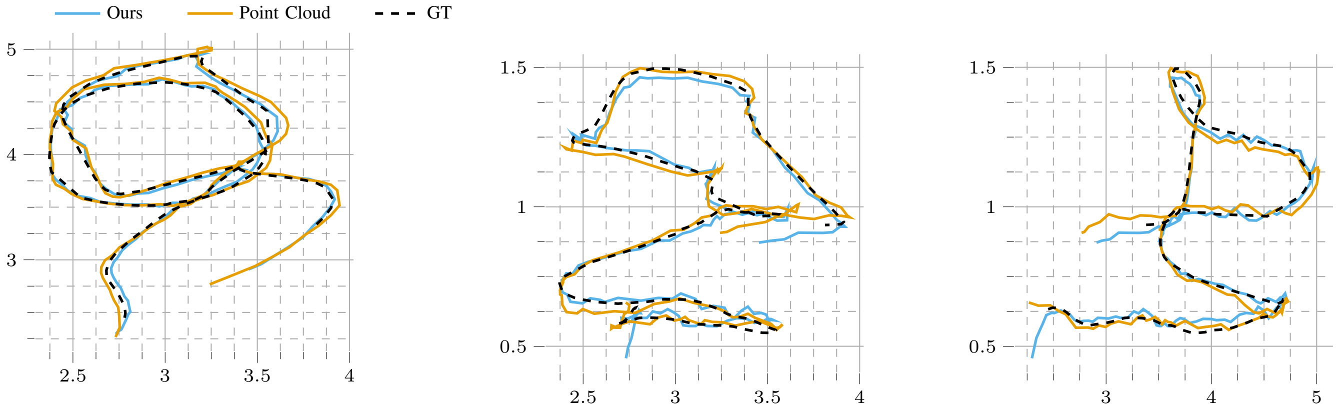


(a) Seq 1

(b) Seq 2

(c) Seq 3

Fig. 5: Absolute position error (APE) value of three data sequences.



(a) XY trajectory plot

(b) XZ trajectory plot

(c) YZ trajectory plot

Fig. 6: Comparison of estimated trajectories with the point cloud tracking method and our proposed method from three different projections.

distance extends to a medium range, the number of UAV point clouds decreases to less than 100, but the three-dimensional structure of the UAV remains discernible. However, when the distance is at a medium range of 7 m according to our results, the number of UAV point clouds reduces to single digits, and the point cloud structure becomes highly unpredictable and unstructured. It is worth noting that in a more realistic application, additional elements such as other sensor payloads or a cargo bay would potentially increase significantly the reflective surface of the UAV.

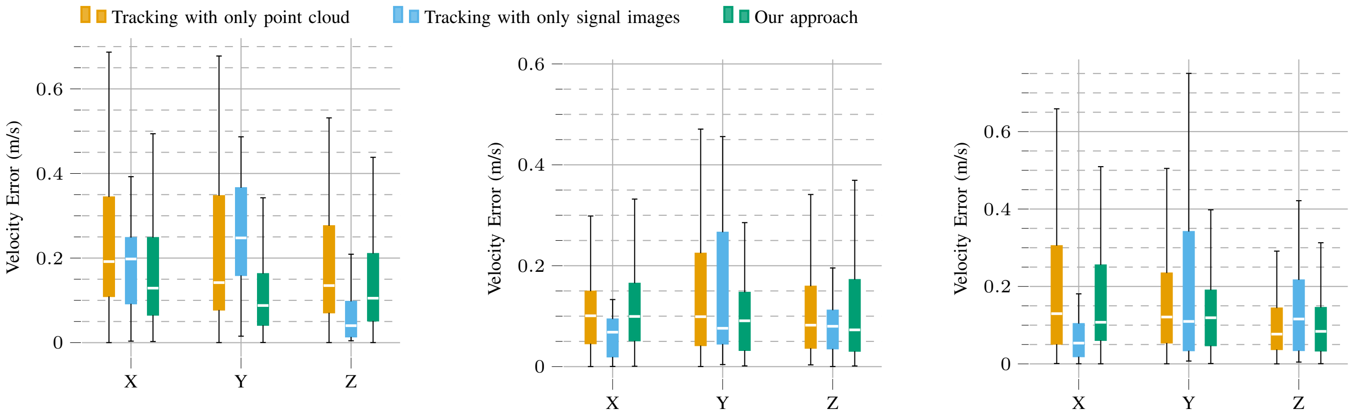
B. Trajectory Validation

We also show a quantitative analysis of the APE based on ground truth, with the main results summarized in Fig. 5. To ensure the trajectories are compared under the same coordinates, we utilize the coordinates of the ground truth as the reference coordinates and convert all trajectories generated by the three UAV tracking methods to these coordinates. Table II presents a comprehensive comparison of three different UAV tracking methods in terms of detectable distance, average APE, algorithm update frequency, and need for initial conditions.

TABLE II: Performance(Detectable distance, frame rate, and APE error) and initial conditions comparison of selected tracking methods.

	Distance *(m)*	**APE Error(Mean/RMSE)** (m)	**FPS** (Hz)	**Initialization**
Tracking with point cloud	**8.0**	0.104 / 0.142	8.3	Yes
Tracking with signal images	2.4	0.078 / 0.088	**10**	No
Fused(Ours)	**8.0**	**0.061 / 0.067**	**10**	No

The image-based UAV tracking method shows a relatively small average error; however, its overall error distribution is inconsistent, as the Y-axis error in the *Seq 1* sequence reaches up to 0.3 m. Conversely, the point cloud-based UAV tracking method has the largest average error, but its error distribution is more uniform. Our proposed method, on the other hand, achieves the smallest average error and minimal error fluctuation. Additionally, to supplement the quantitative trajectory analysis, we also provide a visualization of the trajectories based on the point cloud tracking method and our proposed method from three different viewpoints, as illustrated in Fig 6, with more consistent behavior.



(a) Seq 1

(b) Seq 2

(c) Seq 3

Fig. 7: Velocity estimation error for each linear component in the three data sequences.

C. Velocity Validation

In addition to pose estimation, we conducted a quantitative analysis of the UAV velocities based on the ground truth data and compared them with different UAV tracking methods. Fig. 7 illustrates the velocity errors of each method along the X, Y, and Z axes. The experimental results reveal that all methods have similar mean values of the velocity errors, but different fluctuations. The image tracking method has a large fluctuation in the Y-axis velocity error, reaching up to *0.75 m/s* in *Seq 3*. The point cloud tracking method also has relatively large fluctuations in all dimensions. In contrast, our method achieves smaller overall velocity errors in both the mean value and fluctuation range.

D. Resource Consumption

Both the Intel laptop and the Jetson Nano run ROS Melodic on Ubuntu 18.04. The CPU and memory utilization is measured with a ROS resource monitor tool [4]. Additionally, for minimizing the difference of the operating environment, we unified the dependencies used in each method into same version. The results are summarized in Table III.

The memory utilization of each selected method was roughly equivalent in both processor architecture platforms. However, the same algorithm showed generally higher CPU utilization and achieved the highest publishing rate when running on the Intel processor. For the Intel processor, the point cloud tracking method had higher CPU utilization than other methods but the lowest publishing rate. The fusion method performed well on the laptop and had the smallest overall error. On the embedded computing platform, the CPU utilization of all methods did not differ significantly, and the point cloud tracking method had the lowest memory utilization but the lowest pose publication rate. The difference in CPU utilization is caused by the use of CUDA GPU acceleration in the Open3D binaries utilized for the Jetson Nano platform, while the Intel computer uses only the CPU for point cloud data processing. The image processing also leverages the embedded GPU in the Jetson Nano board. Because of the small ROI that we extract to process the point cloud, the fused method adds little overhead on top of the vision-only method.

TABLE III: Average run-time resource (CPU/RAM) utilization and performance (pose calculation speed) comparison of selected tracking methods across multiple platforms. CPU utilization of 100% equals one full processor core.

	Laptop	Jetson Nano
	(CPU (%), RAM (MB), Pose rate (Hz))	
Tracking with point cloud	(422.7, 296.1, 8.3)	(121.7, 179.7, 5.15)
Tracking with image	(209.0, 293.6, 10)	(113.8, 232.9, 6.07)
Fused (ours)	(195.5, 299.0, 10)	(114.5, 247.8, 6.04)

V. Conclusion

This paper has proposed a novel approach for tracking a UAV based on the fusion of signal images and point clouds from an Ouster LiDAR. Unlike conventional LiDAR and camera fusion, this approach does not need any calibration and preprocessing with external cameras and the LiDAR data is more resistant to harsh environments. We collected three different data sequences in an indoor environment with the OptiTrack mocap system providing ground truth positions. We compared the proposed approach with the approaches based on either only point clouds or signal images and the results showed the effectiveness of our proposed approach. Additionally, we found that our approach can be utilized in a popular mobile computing platform, Jetson Nano according to our evaluation.

Future work includes fusing the Ouster images (depth, signal, and ambient), point clouds, and conventional RGB images in various applications including simultaneous localization and mapping (SLAM), object detection, and tracking.

Acknowledgment

This research work is supported by the Academy of Finland's AeroPolis project (Grant No. 348480).

[4] https://github.com/alspitz/cpu_monitor

References

[1] Rui Fan, Jianhao Jiao, Haoyang Ye, Yang Yu, Ioannis Pitas, and Ming Liu. Key ingredients of self-driving cars. *arXiv preprint arXiv:1906.02939*, 2019.

[2] Shinpei Kato, Shota Tokunaga, Yuya Maruyama, Seiya Maeda, Manato Hirabayashi, Yuki Kitsukawa, Abraham Monrroy, Tomohito Ando, Yusuke Fujii, and Takuya Azumi. Autoware on board: Enabling autonomous vehicles with embedded systems. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, pages 287–296. IEEE, 2018.

[3] Li Qingqing, Yu Xianjia, Jorge Pena Queralta, and Tomi Westerlund. Multi-modal lidar dataset for benchmarking general-purpose localization and mapping algorithms. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3837–3844. IEEE, 2022.

[4] Ardi Tampuu, Romet Aidla, Jan Are van Gent, and Tambet Matiisen. Lidar-as-camera for end-to-end driving. *arXiv preprint arXiv:2206.15170*, 2022.

[5] Angus Pacala. Lidar as a camera – digital lidar's implications for computer vision. *Ouster Blog*, 2018.

[6] Yu Xianjia, Sahar Salimpour, Jorge Pena Queralta, and Tomi Westerlund. Analyzing general-purpose deep-learning detection and segmentation models with images from a lidar as a camera sensor. *arXiv preprint arXiv:2203.04064*, 2022.

[7] Tuan Li, Hongping Zhang, Zhouzheng Gao, Qijin Chen, and Xiaoji Niu. High-accuracy positioning in urban environments using single-frequency multi-gnss rtk/mems-imu integration. *Remote sensing*, 10(2):205, 2018.

[8] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 445–461. Springer, 2016.

[9] Carol Martínez, Pascual Campoy, Iván Mondragón, and Miguel A. Olivares-Méndez. Trinocular ground system to control uavs. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3361–3367, 2009.

[10] Matouš Vrba and Martin Saska. Marker-less micro aerial vehicle detection and localization using convolutional neural networks. *IEEE Robotics and Automation Letters*, 5(2):2459–2466, 2020.

[11] Weiwei Kong, Daibing Zhang, Xun Wang, Zhiwen Xian, and Jianwei Zhang. Autonomous landing of an uav with a ground-based actuated infrared stereo vision system. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2963–2970, 2013.

[12] Peixia Li, Dong Wang, Lijun Wang, and Huchuan Lu. Deep visual tracking: Review and experimental comparison. *Pattern Recognition*, 76:323–338, 2018.

[13] Adrian Carrio, Sai Vemprala, Andres Ripoll, Srikanth Saripalli, and Pascual Campoy. Drone detection using depth maps. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1034–1037. IEEE, 2018.

[14] Li Qingqing, Yu Xianjia, Jorge Peña Queralta, and Tomi Westerlund. Adaptive lidar scan frame integration: Tracking known mavs in 3d point clouds. *arXiv preprint arXiv:2103.04069*, 2021.

[15] Sedat Dogru and Lino Marques. Drone detection using sparse lidar measurements. *IEEE Robotics and Automation Letters*, 7(2):3062–3069, 2022.

[16] Jan Razlaw, Jan Quenzel, and Sven Behnke. Detection and tracking of small objects in sparse 3d laser range data. *2019 International Conference on Robotics and Automation (ICRA)*, pages 2967–2973, 2019.

[17] Marcus Hammer, Marcus Hebel, Björn Borgmann, Martin Laurenzis, and Michael Arens. Potential of lidar sensors for the detection of UAVs. In Monte D. Turner and Gary W. Kamerman, editors, *Laser Radar Technology and Applications XXIII*, volume 10636, page 1063605. International Society for Optics and Photonics, SPIE, 2018.

[18] Marcus Hammer, Marcus Hebel, Martin Laurenzis, and Michael Arens. Lidar-based detection and tracking of small UAVs. In Gerald S. Buller, Richard C. Hollins, Robert A. Lamb, and Markus Mueller, editors, *Emerging Imaging and Sensing Technologies for Security and Defence III; and Unmanned Sensors, Systems, and Countermeasures*, volume 10799, page 107990S. International Society for Optics and Photonics, SPIE, 2018.

[19] Ismail Guvenc, Farshad Koohifar, Simran Singh, Mihail L Sichitiu, and David Matolak. Detection, tracking, and interdiction for amateur drones. *IEEE Communications Magazine*, 56(4):75–81, 2018.

[20] S.Hengy *et al.* Multimodal uav detection: study of various intrusion scenarios. In *Electro-Optical Remote Sensing XI*, volume 10434, page 104340P. International Society for Optics and Photonics, 2017.

[21] J. Peña Queralta *et al.* Autosos: Towards multi-uav systems supporting maritime search and rescue with lightweight ai and edge computing. *arXiv preprint arXiv:2005.03409*, 2020.

[22] Tomáš Rouček, Martin Pecka, Petr Čížek, Tomáš Petříček, Jan Bayer, Vojtěch Šalanskỳ, Daniel Heřt, Matěj Petrlík, Tomáš Báča, Vojěch Spurnỳ, et al. Darpa subterranean challenge: Multi-robotic exploration of underground environments. In *International Conference on Modelling and Simulation for Autonomous Systesm*, pages 274–290. Springer, 2019.

[23] Matěj Petrlík, Tomáš Báča, Daniel Heřt, Matouš Vrba, Tomáš Krajník, and Martin Saska. A robust uav system for operations in a constrained environment. *IEEE Robotics and Automation Letters*, 5(2):2169–2176, 2020.

[24] Jorge Peña Queralta, Li Qingqing, Fabrizio Schiano, and Tomi Westerlund. Vio-uwb-based collaborative localization and dense scene reconstruction within heterogeneous multi-robot systems. *arXiv preprint arXiv:2011.00830*, 2020.

[25] Li Qingqing, Jorge Pena Queralta, Tuan Nguyen Gia, and Tomi Westerlund. Offloading monocular visual odometry with edge computing: Optimizing image quality in multi-robot systems. In *Proceedings of the 2019 5th International Conference on Systems, Control and Communications*, pages 22–26, 2019.

[26] Weiwei Kong, Daibing Zhang, Xun Wang, Zhiwen Xian, and Jianwei Zhang. Autonomous landing of an uav with a ground-based actuated infrared stereo vision system. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2963–2970, 2013.

[27] Yang Gui, Pengyu Guo, Hongliang Zhang, Zhihui Lei, Xiang Zhou, Jing Du, and Qifeng Yu. Airborne vision-based navigation method for uav accuracy landing using infrared lamps. *Journal of Intelligent & Robotic Systems*, 72, 11 2013.