

Received XX Month, XXXX; revised XX Month, XXXX; accepted XX Month, XXXX; Date of publication XX Month, XXXX; date of current version XX Month, XXXX.

Digital Object Identifier 10.1109/XXXX.2022.1234567

CTE-MLO: Continuous-time and Efficient Multi-LiDAR Odometry with Localizability-aware Point Cloud Sampling

Hongming Shen¹, Member, IEEE, Zhenyu Wu¹, Yulin Hui³, Wei Wang^{1,2}, Graduate Student Member, IEEE, Qiyang Lyu^{1,2}, Tianchen Deng², Yeqing Zhu², Bailing Tian³, Member, IEEE, and Danwei Wang^{1,2}, Life Fellow, IEEE

¹Centre for Advanced Robotics Technology Innovation (CARTIN), Nanyang Technological University, Singapore, 639798 Singapore

²School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798 Singapore

³School of Electrical and Information Engineering, Tianjin University, Tianjin, 300072 China

Corresponding author: Zhenyu Wu (email: ZHENYU002@e.ntu.edu.sg).

This research is supported by the National Research Foundation (NRF), Singapore, under the NRF Medium Sized Centre scheme (CARTIN), the Agency for Science, Technology and Research (A*STAR) under its National Robotics Programme with Grant No. M22NBK0109, and National Research Foundation, Singapore and Maritime and Port Authority of Singapore under its Maritime Transformation Programme (Project No. SMI-2022-MTP-04).

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Maritime and Port Authority of Singapore.

ABSTRACT In recent years, LiDAR-based localization and mapping methods have achieved significant progress thanks to their reliable and real-time localization capability. Considering single LiDAR odometry often faces hardware failures and degeneracy in practical scenarios, Multi-LiDAR Odometry (MLO), as an emerging technology, is studied to enhance the performance of LiDAR-based localization and mapping systems. However, MLO can suffer from high computational complexity introduced by dense point clouds that are fused from multiple LiDARs, and the continuous-time measurement characteristic is constantly neglected by existing LiDAR odometry. This motivates us to develop a Continuous-Time and Efficient MLO, namely CTE-MLO, which can achieve accurate and real-time estimation using multi-LiDAR measurements through a continuous-time perspective. In this paper, the Gaussian process estimation is naturally combined with the Kalman filter, which enables each LiDAR point in a point stream to query the corresponding continuous-time trajectory using its time instants. A decentralized multi-LiDAR synchronization scheme is also devised to combine points from separate LiDARs into a single point cloud without the primary LiDAR assignment. Moreover, with the aim of improving the real-time performance of MLO without sacrificing robustness, a point cloud sampling strategy is designed with the consideration of localizability. To this end, CTE-MLO integrates synchronization, localizability-aware sampling, continuous-time estimation, and voxel map management within a Kalman filter framework, which can achieve high accuracy and robust continuous-time estimation within only a few linear iterations. The effectiveness of the proposed method is demonstrated through various scenarios, including public datasets and real-world applications. Exhaustive benchmarks on public dataset show that CTE-MLO is demonstratively competitive compared to other State-of-the-art (SOTA) methods in accuracy, robustness, and efficiency. Furthermore, CTE-MLO has undergone comprehensive validation across various scenarios (such as port, campus, and forest) and on diverse platforms (a full-size truck, an autonomous sweeper, and an agile Micro Aerial Vehicle (MAV)), covering a total test area exceeding 3.6km^2 , which affirm the applicability of CTE-MLO in industrial and field scenarios. A demonstration video is available at <https://youtu.be/Q29PGPitHUI>, and the code is also available at <https://github.com/shenhm516/CTE-MLO> to benefit the community.

INDEX TERMS Autonomous automobiles, Autonomous aerial vehicles, LiDAR odometry, State estimation, Sensor fusion, Simultaneous localization and mapping (SLAM).

I. Introduction

LOCLIZATION is a fundamental task in developing autonomous mobile robots. LiDAR odometry has become one of the mainstream localization solutions in Global Navigation Satellite System (GNSS)-denied environments, which has the potential to enable extensive industrial applications, such as construction [1], [2], autonomous driving [3], [4], and underground mine exploration [5]–[7]. During the last decade, LiDAR odometry has achieved remarkable localization and mapping accuracy thanks to the direct depth measurement of LiDAR, which is immune to scene illumination and texture changes. However, many existing approaches [8]–[10] treat continuous LiDAR point streaming as an instantaneous measurement through the motion undistortion technique. The discrete-time estimation of LiDAR-only odometry suffers from the low frame rate (e.g., 10Hz) of the LiDAR, making it challenging to deploy on agile platforms that require high-frequency state estimation for control and planning feedback. To overcome this issue, many existing approaches fuse LiDAR measurements with the Inertial Measurement Unit (IMU) to provide high-frequency state estimation in a loosely coupled [11]–[13] or tightly coupled [14]–[16] manner. Unfortunately, recent research [17], [18] indicates that LiDAR-inertial odometry can not provide stable state estimation in certain extreme cases, such as vibration and aggressive motions. This motivates us to develop a continuous-time LiDAR odometry that treats the LiDAR point cloud as a continuous-time point stream and utilizes each LiDAR point in the stream to constrain a continuous-time trajectory corresponding to its time domain.

With further consideration of the degeneracy issues [19]–[22], extra onboard sensors (such as visual camera [23]–[27], GNSS [15], [28], [29], encoder [20], [30], [31], and Ultra-wide Bandwidth (UWB) anchors [32]–[34]) are generally adopted to provide constraints to the degeneracy direction of LiDAR odometry through the multi-sensor fusion technique. However, visual cameras can degenerate significantly or even become unobservable in conditions of poor illumination and low texture, and GNSS cannot work in an indoor or dense urban area due to the multi-path effect [35]. Besides, encoders can only be deployed on legged or wheel robots, and UWB anchors rely on installation and calibration. To improve the robustness of state estimation for field robots, a fraction of works [36], [37] focus on robust state estimation using multi-LiDAR measurements in defense of LiDAR odometry. With the continuous reduction in the cost, weight, and power consumption of LiDAR in recent years, multi-LiDAR localization systems can now be deployed not only on large-size robotic platforms such as autonomous driving vehicles [3], [38] and excavators [1], [2] but also on small-size robotic platforms (e.g., unmanned ground vehicles [36], [37], legged robots [39], even MAV [32], [40]) in such real-world industrial and field applications. However, the number of points provided by multiple LiDARs worsens the real-time performance of MLO, which significantly increases

the dimension of the estimation problem. The estimation of LiDAR odometry is a solve-associate-solve loop that implies the MLO can benefit more from the correct association than the nonlinear iterations. As an extreme example, the ego-motion between two LiDAR point clouds with exact point association can be estimated in a closed-form [41] without iteration. Based on this idea, the closed-form solution of the Extended Kalman Filter (EKF) is introduced as a prior of association in each iteration to replace the monotonic cost function descent in nonlinear optimization (e.g., Gaussian Newton, Levenberg-Marquardt), thereby improving the real-time performance of MLO. With further considering that the number of points provided by multiple LiDARs is a trade-off between robustness and real-time performance, we are motivated to develop a point cloud sampling strategy with the consideration of localizability. Hence, in this paper, an EKF-based Continuous-Time and Efficient MLO, so-called CTE-MLO, is developed with a localizability-aware point cloud sampling strategy.

The main contributions of this paper are listed as follows:

- Considering the inherent continuous-time characteristics of the LiDAR point stream, we extend the potential of LiDAR-only odometry and develop a continuous-time MLO that enables real-time and precise trajectory estimation using pure LiDAR measurements without any additional sensor fusion.
- A Gaussian process representation is adopted to describe the continuous-time trajectory, which enables a natural combination of the continuous-time LiDAR odometry with the Kalman filter. To the best of our knowledge, CTE-MLO is the first filter-based continuous-time MLO, which can solve the continuous-time trajectory estimation problem efficiently within a few linear iterations.
- A decentralized multi-LiDAR synchronization scheme is designed to combine multiple LiDAR measurements using the LiDAR splitting technique, which is robust to arbitrary LiDAR failure. Considering the large number of points provided by synchronized multi-LiDAR observation, which preserves a high level of geometric constraint redundancy. An efficient localizability-aware point cloud sampling method is proposed to improve the real-time performance of MLO, which quantifies the localizability contribution for each point, and ensures localizability using a minimal number of points with high localizability contribution.
- A full-fledged continuous-time MLO framework is proposed, which integrates decentralized synchronization, localizability-aware sampling, continuous-time estimation, and voxel map management. Exhaustive benchmark comparison on public datasets shows that CTE-MLO achieves competitive accuracy and efficiency when compared with SOTA MLO methods. Real-world industrial and field applications on various platforms, including a full-scale truck, an autonomous sweeper,

and an agile autonomous exploration MAV, are deployed to demonstrate the performance and industrial applicability of CTE-MLO.

II. Related Works

To develop a continuous-time and efficient MLO, related SOTA methods are reviewed with the subject of LiDAR odometry, continuous-time LiDAR odometry, and MLO.

A. LiDAR Odometry:

LiDAR odometry is designed to estimate the robot's motion between LiDAR acquisitions using the point cloud registration technique represented by Iterative Closest Point (ICP) [41], Generalized Iterative Closest Point (GICP) [42], and Normal Distribution Transformation (NDT) [43]. One of the most popular LiDAR SLAM algorithms LOAM [8] provides a standard framework for LiDAR odometry, which achieves real-time ego-motion estimation through the scan-to-scan matching and low-drift registration through the scan-to-map matching. The odometry and mapping framework proposed in LOAM has been widely adopted in recent research. Several techniques used in LOAM, such as feature extraction, scan-to-scan matching, and motion estimation, are improved by current SOTA LiDAR SLAM methods. In DLO [9], a direct LiDAR odometry is developed to provide real-time pose estimation using the minimally preprocessed dense point cloud without feature extraction. A pruned GICP registration algorithm is designed in [9] to accelerate the scan-to-scan and scan-to-map process with an adaptive keyframe strategy. In FLOAM [10], the scan-to-scan matching in LOAM [8] is replaced by the assumption of a uniform motion model. LeGO-LOAM [44] improves the accuracy and real-time performance of LOAM by segmenting the point from the ground. To further improve the localization and mapping performance, MULLS [45] is proposed by segmenting the points in the scan with more detailed types (ground, facade, roof, pillar, beam, and vertex) when compared with LeGO-LOAM [44]. The aforementioned methods achieve LiDAR-based estimation using the GICP registration or variants (e.g. performing registration using the principal direction of the covariance in GICP). In recent years, a few methods attempting to return to the roots, which introduce the classic ICP registration to develop a LiDAR odometry in a concise formulation. Open3d-SLAM [46] provides a fully LiDAR SLAM solution, including the ICP-based estimation, loop closure, and map management, leveraging a highly performant point cloud library Open3D [47]. However, the performance of ICP registration highly relies on the accuracy of the initial guess, because it imposes point association in the nearest distance. KISS-ICP [48] solves this problem by combining the classical ICP registration with an adaptive matching strategy, which yields a lightweight LiDAR-only localization system in defense of point-to-point ICP. Besides LiDAR odometry based on ICP, GICP, and their variants, NDT provides a more lightweight estimation and map man-

agement for LiDAR odometry [49], [50]. Different from GICP registration, which performs the normal distribution approximation using an exact nearest neighbor search, NDT approximates normal distribution through lightweight voxelization clustering. However, NDT-based LiDAR odometry [49], [50] tends to cluster voxels in a pre-defined (fixed) size, which reduces the adaptability to variational scenarios. To overcome this, in BALM [51], a novel adaptive voxelization mapping structure is developed using the hash data structure, and a theoretical framework for performing LiDAR Bundle Adjustment (BA) is also proposed. Although BA has achieved good performance in the visual odometry field [23], [52]–[55], for LiDAR odometry which has 10–100 times of points than visual odometry, registration-based LiDAR odometry is still the mainstream, while LiDAR BA is commonly applied in the fields of non-real-time estimation and mapping, such as fine pose graph optimization [56] and consistent map building [57]. For tasks with high-frequency estimation requirements, e.g. feedback control for agile platform, current mainstream LiDAR odometry achieves real-time estimation through fusion with IMU which neglects the high-bandwidth characteristic of the LiDAR point streaming.

B. Continuous-time LiDAR Odometry

The aforementioned LiDAR odometry treats LiDAR scans as instantaneous measurements by employing motion undistortion techniques and utilizes the undistorted point cloud for the registration process. However, the motion undistortion relies on the scan-to-scan matching results or uniform motion assumption, which introduces the additional measurement noise into the undistorted point cloud. To solve this problem, a series of approaches investigate continuous-time LiDAR odometry with the idea of taking into account the motion during scanning by defining a continuous-time trajectory. In MARS [58], a continuous-time LiDAR odometry is proposed to estimate the B-spline trajectory and remove the motion distortion of the LiDAR scan with a non-rigid registration. Thanks to the continuous-time B-spline representation, in [59]–[61], a serial continuous-time trajectory estimation system is developed by combining asynchronous sensor measurements (e.g., camera, IMU) with the continuous-time LiDAR scan. However, according to the definition of B-spline [62], the formulation of the B-spline trajectory is determined by a set of control points, which includes control points not in the current time domain. Estimation of future control points leads to redundant computations for B-spline-based continuous-time SLAM systems. To solve this issue, linear interpolation is employed to model the continuous movement straightforwardly. CT-ICP [63] and KISS-ICP [48] show great trajectory estimation results by incorporating the motion undistortion into the registration using linear interpolation. The uniform motion assumption implicated in linear interpolation leads CT-ICP [63] and KISS-ICP [48] not robust enough to sudden changes of orientation or fast acceleration motion. In Traj-LO [18], a piecewise

linear interpolation is adopted to minimize the gap between uniform motion assumption and aggressive motion. In [64], the Gaussian Process Gauss-Newton (GPGN) optimization method is derived to enable continuous-time estimation, while [65] extends the GPGN to the SE(3) space. Currently, there only exists limited research focus on the Gaussian process continuous-time LiDAR-only odometry. In [66], a continuous-time LiDAR odometry is designed to estimate the Gaussian process trajectory for a novel LiDAR type, called Frequency-Modulated Continuous-Wave (FMCW) LiDAR which can measure the relative velocity of each measured point via the Doppler effect. Unfortunately, the application of FMCW LiDAR is still relatively limited in the industry platforms.

C. Multi-LiDAR Odometry

Considering that single LiDAR odometry frequently suffers from hardware failure and degeneracy, MLO has become an emerging research topic in SLAM. MLOAM [36] is a representative MLO, which has been proven to have better performance compared to single-LiDAR odometry methods. In MILIOM [67], a tightly coupled multi-LiDAR-inertial odometry is proposed by combining multiple LiDAR measurements through a general synchronization scheme. In SLICT [68] and SLICT2 [60], the authors extend MILIOM [67] to continuous-time multi-LiDAR-inertial odometry with linear interpolation and B-spline, respectively. However, the synchronization strategy adopted in [36], [60], [67], [68] is centralized, which can not provide stable estimation in the situation of primary LiDAR failure. An intuitive solution to this problem is to develop a decentralized MLO. In [37], a decentralized MLO framework is developed to realize robust state estimation exploits information from multiple Field of View (FOV)-limited LiDARs. However, decentralized MLO has to estimate the state or extrinsic for each LiDAR, which significantly increases the computational complexity of the state estimation process. On the mapping side, the dense point cloud acquired from multiple LiDARs poses a challenge to the efficiency of the map management process. In most conventional MLO [36], [37], [67], KD-Tree is introduced as the mapping structure to accelerate the K-NN querying process. However, conventional KD-Tree does not support incremental updates, which means a compulsory rebuild process is required to maintain the global map. To solve this problem, incremental KD-Tree (iKD-Tree) [69] and incremental voxel (iVox) [70] are developed to accelerate the mapping process, which maintains the global map incrementally. Although iKD-Tree and iVox achieve efficient data insertion, the time complexity of the K-NN search is still related to the size of the KD-Tree and the number of points in the voxel, respectively. In [71], an efficient adaptive voxel mapping algorithm is proposed by combining the octree with the hash data structure, which achieves efficient data association by clustering points in each voxel into normal distributions. To future improve the mapping efficiency,

SLICT [68] and SLICT2 [60] perform incremental updates on each voxel in the voxel map, enabling incremental updates and efficient query of the global map for MLO. In view of the aforementioned analysis, only a few MLO methods [60], [68] considered the continuous measurement characteristics of LiDAR, which all heavily rely on centralized multi-LiDAR synchronization and IMU measurements. For pure MLO [36], [37], the real-time performance is always worsened by the dense point cloud acquired from multiple LiDARs, the high-dimension state to be estimated, or inefficient map management. To overcome these problems, in this paper, CTE-MLO is developed to achieve a continuous-time and efficient MLO in a tightly coupled manner by integrating decentralized synchronization, continuous-time estimation, localizability-aware point cloud sampling, and efficient voxel map management.

III. Gaussian Process Continuous-Time LiDAR Odometry

A continuous-time LiDAR-only odometry is in charge of estimating a trajectory $\mathbf{x}(t), t \in [t_k, t_{k+1})$ with the observation from the LiDAR point cloud, where an arbitrary state vector $\mathbf{x} \in \mathbf{x}(t)$ is assumed to follow a Gaussian distribution. A Gaussian process is intuitively adopted to describe the continuous-time trajectory $\mathbf{x}(t)$ with a mean function $\bar{\mathbf{x}}(t)$ and a covariance function $\sigma_p(t, t')$.

$$\mathbf{x}(t) \sim \mathcal{GP}(\bar{\mathbf{x}}(t), \sigma_p(t, t')) \quad (1)$$

where the covariance function involves two time variables, t and t' , to account for cross-temporal relations.

The continuous-time LiDAR-based localization and mapping problem can be formulated as maximizing the posterior probability of the trajectory $\mathbf{x}(t)$ with LiDAR measurements \mathbf{z}_{t_k} plus an initial state $\check{\mathbf{x}}_{t_k}$.

$$\hat{\mathbf{x}}(t) = \arg \max_{\mathbf{x}(t)} p(\mathbf{x}(t) | \mathbf{z}_{t_k}, \check{\mathbf{x}}_{t_k}) \quad (2)$$

where $p(\cdot)$ denotes the probability function. For a measurement obtained at time $t_i \in [t_k, t_{k+1})$, the measurement model is defined to be

$$\mathbf{z}_{t_i} = h(\mathbf{x}(t_i)) + \mathbf{w}_{t_i}, \mathbf{w}_{t_i} \sim \mathcal{N}(\mathbf{0}, \sigma_{R,t_i}) \quad (3)$$

where $h(\cdot)$ denotes the observation function, and σ_{R,t_i} is the covariance of the measurement noise \mathbf{w}_{t_i} .

With the Gaussian approximation, the Maximum a Posteriori (MAP) problem defined in (2) can be simplified as a least square optimization problem.

$$\min_{\mathbf{x}(t)} \underbrace{\|\bar{\mathbf{x}}(t_k) - \check{\mathbf{x}}_{t_k}\|_{\sigma_{P,t_k}^{-1}}^2 + \sum_{i=1}^M \|\mathbf{z}_{t_i} - h(\bar{\mathbf{x}}(t_i))\|_{\sigma_{R,t_i}^{-1}}^2}_{V} \quad (4)$$

where M is the number of LiDAR measurements, $\bar{\mathbf{x}}(t_k)$ is the mean function corresponding to the trajectory $\mathbf{x}(t)$ at time $t = t_k$, and $\mathbf{x}(t_k)$ is subject to a Gaussian distribution $\mathbf{x}(t_k) \sim \mathcal{N}(\bar{\mathbf{x}}(t_k), \sigma_{P,t_k})$.

Inspired by [72], the Gaussian Process is assumed to be generated by Linear, Time-varying (LTV) Stochastic Differential Equations (SDE).

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \quad (5)$$

where $\mathbf{F}(t)$ and $\mathbf{G}(t)$ are time-varying system matrices, $\mathbf{w}(t) \sim \mathcal{GP}(\mathbf{0}, \sigma_Q \delta(t-t'))$ is a stationary, zero-mean, white-noise Gaussian process with power-spectral density matrix σ_Q , and $\delta(\cdot)$ denotes the Dirac delta function.

Under the LTV SDE assumption, the mean function of the Gaussian process $\mathbf{x}(t)$ can be expressed by adopting a basis function representation [73]:

$$\begin{aligned} \bar{\mathbf{x}}(t) &= \Phi(t, t_k)\bar{\mathbf{x}}(t_k) \\ \sigma_P(t, t') &= \Phi(t, t_k)\sigma_P(t_k, t_k)\Phi^\top(t', t_k) \\ &\quad + \int_{t_k}^{\min(t, t')} \Phi(t, s)\mathbf{G}(s)\sigma_Q\mathbf{G}^\top(s)\Phi^\top(t', s)ds \end{aligned} \quad (6)$$

where $\Phi(t, t_k)$ is the transition matrix of the LTV system. Substitute (6) into (4), the cost function V of the localization and mapping problem (4) can be rewritten as:

$$\begin{aligned} V &= \|\bar{\mathbf{x}}(t_k) - \tilde{\mathbf{x}}_{t_k}\|_{\sigma_{P,t_k}^{-1}}^2 + \sum_{i=1}^M \|\mathbf{z}_{t_i} - h(\Phi(t_i, t_k)\bar{\mathbf{x}}(t_k))\|_{\sigma_{R,t_i}^{-1}}^2 \\ &= \|\bar{\mathbf{x}}(t_k) - \tilde{\mathbf{x}}_{t_k}\|_{\sigma_{P,t_k}^{-1}}^2 + \|\mathbf{z} - \mathbf{h}\|_{\sigma_R^{-1}}^2 \\ \mathbf{z} &= [\dots, \mathbf{z}_{t_i}^\top, \dots]^\top, \mathbf{h} = [\dots, h^\top(\Phi(t_i, t_k)\bar{\mathbf{x}}(t_k)), \dots]^\top \\ \sigma_R &= \text{Diag}([\dots, \sigma_{R,t_i}, \dots]) \end{aligned} \quad (7)$$

where $\text{Diag}(\cdot)$ denotes the diagonal operation.

To minimize the cost function defined in (7) with an initial guess $\tilde{\mathbf{x}}_{t_k}$, a first-order approximate is adopted to update $\tilde{\mathbf{x}}_{t_k}$ by a small perturbation $\tilde{\mathbf{x}}_{t_k}$.

$$\bar{\mathbf{x}}(t_k) = \tilde{\mathbf{x}}_{t_k} + \tilde{\mathbf{x}}_{t_k} \quad (8)$$

$$\begin{aligned} \tilde{\mathbf{x}}_{t_k} &= \arg \min_{\tilde{\mathbf{x}}_{t_k}} \|\tilde{\mathbf{x}}_{t_k}\|_{\sigma_{P,t_k}^{-1}}^2 + \left\| \mathbf{z} - \mathbf{h}_{\tilde{\mathbf{x}}} - \mathbf{H}|_{\tilde{\mathbf{x}}_{t_k}} \Phi \tilde{\mathbf{x}}_{t_k} \right\|_{\sigma_R^{-1}}^2 \\ \mathbf{h}_{\tilde{\mathbf{x}}} &= [\dots, h^\top(\Phi(t_i, t_k)\tilde{\mathbf{x}}(t_k)), \dots]^\top \\ \Phi &= [\dots, \Phi^\top(t_i, t_k), \dots]^\top \end{aligned} \quad (9)$$

where \mathbf{H} is the Jacobian matrix of \mathbf{h} with respect to $[\dots, \bar{\mathbf{x}}^\top(t_i), \dots]^\top$. The optimization problem defined in (9) is a Standard Quadratic Programming (SQP) problem, which has a closed-form solution by taking the derivative with respect to $\tilde{\mathbf{x}}_{t_k}$.

$$\underbrace{\left(\Phi^\top \mathbf{H}^\top \sigma_R^{-1} \mathbf{H} \Phi + \sigma_{P,t_k}^{-1} \right)}_{\text{Hessian Matrix}} \tilde{\mathbf{x}}_{t_k} = \Phi^\top \mathbf{H}^\top \sigma_R^{-1} (\mathbf{z} - \mathbf{h}) \quad (10)$$

Since the derivation follows the Gauss-Newton approach [64], the covariance can be expressed as the inverse of the Hessian matrix in (10).

$$\sigma_P(t_k, t_k) = \left(\Phi^\top \mathbf{H}^\top \sigma_R^{-1} \mathbf{H} \Phi + \sigma_{P,t_k}^{-1} \right)^{-1} \quad (11)$$

The Gaussian process $\mathbf{x}(t) \sim \mathcal{GP}(\bar{\mathbf{x}}(t), \sigma_P(t_k, t))$ can be estimated by substitute $\bar{\mathbf{x}}(t_k)$ (updated in (8) using the closed-form solution $\tilde{\mathbf{x}}(t_k)$ of SQP (10)) and the updated covariance $\sigma_P(t_k, t_k)$ (11) into (6).

A. Combining Gaussian Process with Kalman Filter

With the derivation of (1)-(11), the key to estimating the Gaussian process $\mathbf{x}(t)$ is to solve the perturbation $\tilde{\mathbf{x}}_{t_k}$ and the covariance matrix $\sigma_P(t_k, t_k)$ with the initial guess $\tilde{\mathbf{x}}_{t_k}$ and σ_{P,t_k} , which process can be realized using the Kalman filter. To facilitate the derivation, an intermediate matrix $\mathbf{H}' = \mathbf{H}\Phi$ is defined, which is the Jacobian matrix of \mathbf{h} with respect to $\bar{\mathbf{x}}(t_k)$. The Gauss-Newton equations (10) can be rewritten as:

$$\tilde{\mathbf{x}}_{t_k} = \underbrace{\left(\mathbf{H}'^\top \sigma_R^{-1} \mathbf{H}' + \sigma_{P,t_k}^{-1} \right)^{-1} \mathbf{H}'^\top \sigma_R^{-1} (\mathbf{z} - \mathbf{h})}_{\mathbf{K}_{t_k}} \quad (12)$$

where \mathbf{K}_{t_k} is equivalent to the Kalman gain matrix with the Sherman-Morrison-Woodbury identity [74], [75].

Hence, Gaussian process estimation using the Kalman filter is summarized as follows:

- 1) Get the initial guess $\tilde{\mathbf{x}}_{t_k}$ and σ_{P,t_k} through (6) which is equivalent to the state prediction process of the Kalman filter.
- 2) Update the covariance matrix $\sigma_P(t_k, t_k)$ and get the Kalman gain matrix \mathbf{K}_{t_k} :

$$\begin{aligned} \sigma_P(t_k, t_k) &= \left(\mathbf{H}'^\top \sigma_R^{-1} \mathbf{H}' + \sigma_{P,t_k}^{-1} \right)^{-1} \\ \mathbf{K}_{t_k} &= \sigma_P(t_k, t_k) \mathbf{H}'^\top \sigma_R^{-1} \end{aligned} \quad (13)$$

- 3) Update the mean vector $\bar{\mathbf{x}}(t_k)$ by substituting (12) and (13) into (8):

$$\bar{\mathbf{x}}(t_k) = \tilde{\mathbf{x}}_{t_k} + \mathbf{K}_{t_k}(\mathbf{z} - \mathbf{h}) \quad (14)$$

- 4) Step 2-3 iterates until convergence or exceeding the maximum number of iterations. At each iteration, the initial guess is updated according to $\tilde{\mathbf{x}}_{t_k} = \bar{\mathbf{x}}(t_k) + \tilde{\mathbf{x}}_{t_k}$, with the covariance given by $\sigma_{P,t_k} = \sigma_P(t_k, t_k)$.
- 5) Estimate the Gaussian process $\mathbf{x}(t)$ by substituting $\bar{\mathbf{x}}(t_k)$ and $\sigma_P(t_k, t_k)$ into (6).

B. Specialize in Continuous-time LiDAR Odometry

In this section, a continuous-time LiDAR-only odometry is developed through the state prediction and state update process of the Kalman filter, which are described in more detail below. Consider a continuous-time LiDAR odometry, the state of the Gaussian process is defined as:

$$\mathbf{x}(t) = [\mathbf{t}^\top(t), \boldsymbol{\phi}^\top(t), \mathbf{v}^\top(t), \boldsymbol{\omega}^\top(t), \mathbf{a}^\top(t)]^\top \quad (15)$$

where $\mathbf{t}(t)$ is the trajectory of translation, $\boldsymbol{\phi}(t)$ is the axis-angle representation of the rotation matrix trajectory $\mathbf{R}(t) \in \text{SO}(3)$, $\mathbf{v}(t)$ is the velocity trajectory, $\boldsymbol{\omega}(t)$ is the angular speed trajectory, and $\mathbf{a}(t)$ is the linear acceleration trajectory.

1) State Prediction

With the definition of the Gaussian process $\mathbf{x}(t)$ in (15), the LTV SDE (5) can be specified as:

$$\begin{aligned}\dot{\mathbf{t}}(t) &= \mathbf{v}(t), \dot{\mathbf{v}}(t) = \mathbf{R}(t)\mathbf{a}(t), \dot{\phi}(t) = \omega(t) \\ \dot{\omega}(t) &= \mathbf{w}_\omega(t), \dot{\mathbf{a}}(t) = \mathbf{w}_a(t)\end{aligned}\quad (16)$$

where $\mathbf{w}_a(t)$ and $\mathbf{w}_\omega(t)$ are the white noise Gaussian processes of acceleration and angular speed. For a LiDAR-only odometry, the kinematic model (16) is adopted instead of the IMU prediction formulation in the state prediction process, and the initial guess $\check{\mathbf{x}}_{t_k}$, σ_{P,t_k} can be predicted by introducing (16) into the general solution of LTV SDE (6).

2) State Update

The state update process is in charge of solving the Gauss-Newton equation (12) using the continuous-time LiDAR measurements. For point \mathbf{p}_i ($t_i \in [t_k, t_{k+1}]$), the objective of the LiDAR matching process is to construct the state-dependent observation function through geometric constraints between the point \mathbf{p}_i and the voxel map \mathbb{M} (details can be found in Section V). The point-to-voxel matching problem can be formulated as a Maximum Likelihood Estimation (MLE) problem.

$$\max L(\mathbf{W}\mathbf{p}_i|\mathbb{M}) \stackrel{\Delta}{=} \min -\log(L(\mathbf{W}\mathbf{p}_i|\mathbb{M})) \quad (17)$$

where $L(\mathbf{W}\mathbf{p}_i|\mathbb{M})$ represents the likelihood of observing a point $\mathbf{W}\mathbf{p}_i$ belonging to the voxel map \mathbb{M} , and $\mathbf{W}\mathbf{p}_i$ is the transformed point defined as

$$\mathbf{W}\mathbf{p}_i = \bar{\mathbf{R}}(t_i)\mathbf{p}_i + \bar{\mathbf{t}}(t_i) \quad (18)$$

where $\{\bar{\mathbf{R}}(t_i), \bar{\mathbf{t}}(t_i)\} \in \bar{\mathbf{x}}(t_i)$, and $\bar{\mathbf{x}}(t_i) = \Phi(t_i, t_k)\bar{\mathbf{x}}(t_k)$.

Following our previous work [76], [77], the likelihood $L(\mathbf{W}\mathbf{p}_i|\mathbb{M})$ can be simplified through the Gaussian Mixture Model (GMM).

$$\begin{aligned}L(\mathbf{W}\mathbf{p}_i|\mathbb{M}) &= L(\mathbf{W}\mathbf{p}_i|\mathbf{m}_j) = \frac{1}{[(2\pi)^3 |\sigma_j^m|]^{\frac{1}{2}}} \exp(-D) \\ -\log(L(\mathbf{W}\mathbf{p}_i|\mathbf{m}_j)) &= -\log \underbrace{\left(\frac{1}{[(2\pi)^3 |\sigma_j^m|]^{\frac{1}{2}}} \right)}_{\text{independent of } \mathbf{x}(t)} + D\end{aligned}\quad (19)$$

where $D = \frac{1}{2} (\mathbf{W}\mathbf{p}_i - \mu_j^m)^\top \sigma_j^{m,-1} (\mathbf{W}\mathbf{p}_i - \mu_j^m)$, μ_j^m is the mean vector of the voxel $\mathbf{m}_j \sim \mathcal{N}(\mu_j^m, \sigma_j^m)$, and \mathbf{m}_j is the voxel corresponding to $\mathbf{W}\mathbf{p}_i$.

According to (19), the MLE problem (17) can be simplified as a least-square optimization problem:

$$\min (\mathbf{W}\mathbf{p}_i - \mu_j^m)^\top \sigma_j^{m,-1} (\mathbf{W}\mathbf{p}_i - \mu_j^m) \quad (20)$$

Adopting local geometry approximated by GMM will affect the stability of MLE if the covariance matrix σ_j^m does not have an inverse. A stabilization technique is adopted in this work, which simplifies (20) through the principal component analysis.

$$\min (\mathbf{W}\mathbf{p}_i - \mu_j^m)^\top \mathbf{n}_j^m \mathbf{n}_j^{m,\top} (\mathbf{W}\mathbf{p}_i - \mu_j^m) \quad (21)$$

where \mathbf{n}_j^m is the normal vector of the voxel $\mathbf{m}_j \sim \mathcal{N}(\mu_j^m, \sigma_j^m)$.

Hence, the observation model of the i -th continuous-time LiDAR matching factor can be defined as

$$z_{t_i} = 0, h(\Phi(t_i, t_k)\bar{\mathbf{x}}(t_k)) = \mathbf{n}_j^{m,\top} (\mathbf{W}\mathbf{p}_i - \mu_j^m) \quad (22)$$

From the definition of the observation model (22), the Jacobian of $h(\Phi(t_i, t_k)\bar{\mathbf{x}}(t_k))$ over the mean function of the Gaussian process $\bar{\mathbf{x}}(t_k)$ can be calculated as follows:

$$\begin{aligned}\mathbf{H}'_i &= \frac{\partial h(\Phi(t_i, t_k)\bar{\mathbf{x}}(t_k))}{\partial \bar{\mathbf{x}}(t_k)} \Big|_{\bar{\mathbf{x}}_{t_k}} = [\mathbf{H}'_i^t, \mathbf{H}'_i^R, \mathbf{H}'_i^V, \mathbf{H}'_i^\omega, \mathbf{H}'_i^a] \\ \mathbf{H}'_i^t &= \mathbf{n}_j^{m,\top}, \mathbf{H}'_i^V = \mathbf{n}_j^{m,\top} \Delta t, \mathbf{H}'_i^a = \frac{1}{2} \eta_j \Delta t^2 \\ \mathbf{H}'_i^R &= -\eta_j \left([\exp([\tilde{\omega}_{t_k} \Delta t]_\times) \mathbf{p}_i]_\times + \frac{1}{2} [\tilde{\mathbf{a}}_{t_k}]_\times \Delta t^2 \right) \\ \mathbf{H}'_i^\omega &= -\eta_j \exp([\tilde{\omega}_{t_k} \Delta t]_\times) [\mathbf{p}_i]_\times \Delta t\end{aligned}\quad (23)$$

where $\eta_j = \mathbf{n}_j^{m,\top} \tilde{\mathbf{R}}_{t_k}$, $[\cdot]_\times$ returns the skew-symmetric matrix of a vector, $\Delta t_i = t_i - t_k$, and \mathbf{H}'_i is the i -th row of \mathbf{H}' . The right Lie derivative model [78] is introduced to derive the Jacobian in (23).

Substitute the continuous-time LiDAR observation model (22) and the analytic Jacobians (23) into (13) and (14), the mean vector and covariance matrix of the state vector $\mathbf{x}(t_k) \in \mathbf{x}(t)$ can be estimated in the state update process of Kalman filter, and the Gaussian process $\mathbf{x}(t)$ can also be estimated by substitute $\mathbf{x}(t_k) \sim \mathcal{N}(\bar{\mathbf{x}}(t_k), \sigma(t_k, t_k))$ into (6).

Remark 1:

(Continuous-time LiDAR odometry) With the derivation in Section III-A, the Gaussian process estimation problem (4) can be solved by combining the continuous-time LTV SDE solution (6) with the Kalman filter. Conventional LiDAR odometry [8]–[10], [36], [37], [44], [45] generally performs state estimation with Kalman filter or optimization in a discrete formulation which treats LiDAR point cloud as an instantaneous measurement through the deskeew technique. However, different from instantaneous sensor suites, such as camera and IMU, LiDAR is naturally a continuous-time sampling sensor that captures LiDAR points continuously and accumulates LiDAR points as a point cloud \mathbb{P} before publishing. In this paper, a continuous-time LiDAR odometry is developed by estimating the Gaussian process represented continuous-time trajectory in a Kalman filter formulation, which can handle the distortion of the LiDAR scan during the estimation process.

IV. Point Cloud Synchronization and Sampling

This section gives a detailed description of the multi-LiDAR point cloud preprocessing, which is in charge of providing observations for the continuous-time LiDAR odometry.

A. Decentralized Multi-LiDAR Synchronization

To perform joint estimation across multiple LiDARs in a tightly coupled manner, data streams have to be synchronized. In practice, all LiDARs are connected to the onboard computer through high-speed Ethernet cables. A popular robotics communication middleware, ROS [79], is adopted to synchronize all LiDAR topics into the Coordinated Universal Time (UTC) with the LiDAR drivers. Hence, in this section, synchronization is considered in the data stream aspect. For most existing multi-LiDAR synchronization schemes [36], [60], [67], [68], multi-LiDAR measurements are synchronized to the end time of the primary LiDAR (so-called centralized synchronization), which is not robust to the primary LiDAR data loss. As shown in Fig. 1, when the primary LiDAR fails (in period $[t'_k, t'_{k+1}]$), the centralized synchronization scheme is unable to synchronize other LiDARs' points to the primary LiDAR. Considering the unstable LiDAR data stream in practical applications, commonly caused by sensor overheating or unreliable wiring, a decentralized multi-LiDAR synchronization strategy is proposed, which treats LiDAR messages as a continuous-time point stream. For each point in the LiDAR point cloud message, we can simultaneously obtain its coordinate p_i (in LiDAR frame) and time stamp t_i (UTC) through the LiDAR driver. At time t_k , all points obtained from each LiDAR in a period $[t_k, t_k + \Delta t]$ are merged into a point cloud \mathbb{P} . The proposed method achieves multi-LiDAR synchronization with a fixed time interval Δt using the LiDAR scan splitting technique, which implies the proposed synchronization strategy is totally decentralized and robust to arbitrary LiDAR failure. To illustrate the effectiveness, as shown in Fig. 2, the proposed multi-LiDAR synchronization strategy is evaluated in a real-world autonomous driving scenario (sequence Port 01 in Table 3 is adopted for testing). The conventional centralized synchronization strategy [60], shown in Fig. 2(b), can lead to point cloud synchronization failure and result in localization failure when the pre-defined primary LiDAR failure (LiDAR #1 experienced data loss at 591 – 654s, 675 – 801s, 1269 – 1597s, and 1609 – 1664s due to overly high temperatures, and the data of LiDAR #7 is totally lost due to the wiring problem). The proposed synchronization strategy is completely decentralized, which splits the point stream from each LiDAR with a fixed time interval $\Delta t = 0.01s$ and merges them into a point cloud \mathbb{P} . As illustrated in Fig. 2(c), the fusion of multiple LiDAR point streams provides denser observations compared to a single split LiDAR point stream (represented by different colors in Fig. 2(c)). The proposed decentralized synchronization strategy combines multiple point streams, each with a limited FOV, into a single dense LiDAR point cloud with an almost panoramic horizontal FOV at a frequency of 100 Hz.

B. Localizability-aware Point Cloud Sampling

Multi-LiDAR SLAM methods generally suffer from high latency due to a high level of point redundancy which

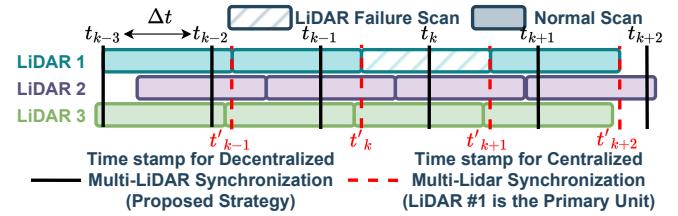


FIGURE 1. Decentralized multi-LiDAR synchronization scheme. For centralized synchronization, no point can merge in a period $[t'_k, t'_{k+1}]$ due to the primary LiDAR failure (LiDAR #1). The decentralized synchronization scheme can obtain synchronized point clouds even when only one LiDAR is functioning.

leads to expensive data association and estimation. The point cloud sampling strategy is a trade-off between the real-time performance and the stability of LiDAR odometry. Removing too many points from the point cloud may lead to accuracy reduction or even degeneration (the point-to-voxel registration problem [17] degenerated into a non-convex optimization problem). Hence, in this paper, a localizability-aware point cloud sampling method is proposed to remove point redundancy with the evaluation of LiDAR degeneracy. It is worth noting that the conventional point sampling methods [20], [80] require knowledge about point-to-map correspondence relationships. Getting point-to-map correspondence for MLO is expensive as the point-to-map matching has to be recomputed in every iteration. This motivates us to develop a point cloud sampling method without prior knowledge of the point-to-map matching information.

Remark 2:

(Definition of Localizability) In this paper, localizability is a metric to determine if the geometry observations can well constrain the LiDAR-based estimation. To illustrate the definition of localizability, a 2D example is provided in Fig. 3. Fig. 3(a) illustrates an environment consisting of two perpendicular planes that well constrain the LiDAR odometry. For example, if Plane 2 in Fig. 3(a) is removed, the LiDAR odometry will degenerate in the normal direction of Plane 2, which means \mathbf{p}_3 and \mathbf{p}_4 contribute to the localizability in the normal direction of their corresponding plane (Plane 2). For the rotation perspective, as shown in Fig. 3(b), \mathbf{p}_1 and \mathbf{p}_2 contribute localizability, and the rotation estimation will be dominated by observation noise (so-called rotation degeneracy) when \mathbf{p}_1 and \mathbf{p}_2 are removed.

To facilitate the LiDAR degeneracy analysis, we substitute the LiDAR observation model (22) into (4) and neglect the predictive constraints (irrelevant to LiDAR observations).

$$\min_{\mathbf{x}(t)} \|\mathbf{h}\|_{\sigma_R^{-1}}^2 \triangleq \min_{\mathbf{x}(t)} \sum_{i=1}^M \left\| \mathbf{n}_j^{m,\top} (\bar{\mathbf{R}}(t_i) \mathbf{p}_i + \bar{\mathbf{t}}(t_i) - \boldsymbol{\mu}_j^m) \right\|_{\sigma_{R,t_i}^{-1}}^2 \quad (24)$$

Referring to the derivations in Section III, the Gaussian process Kalman filter solves (24) by iteratively updating $\bar{\mathbf{x}}$

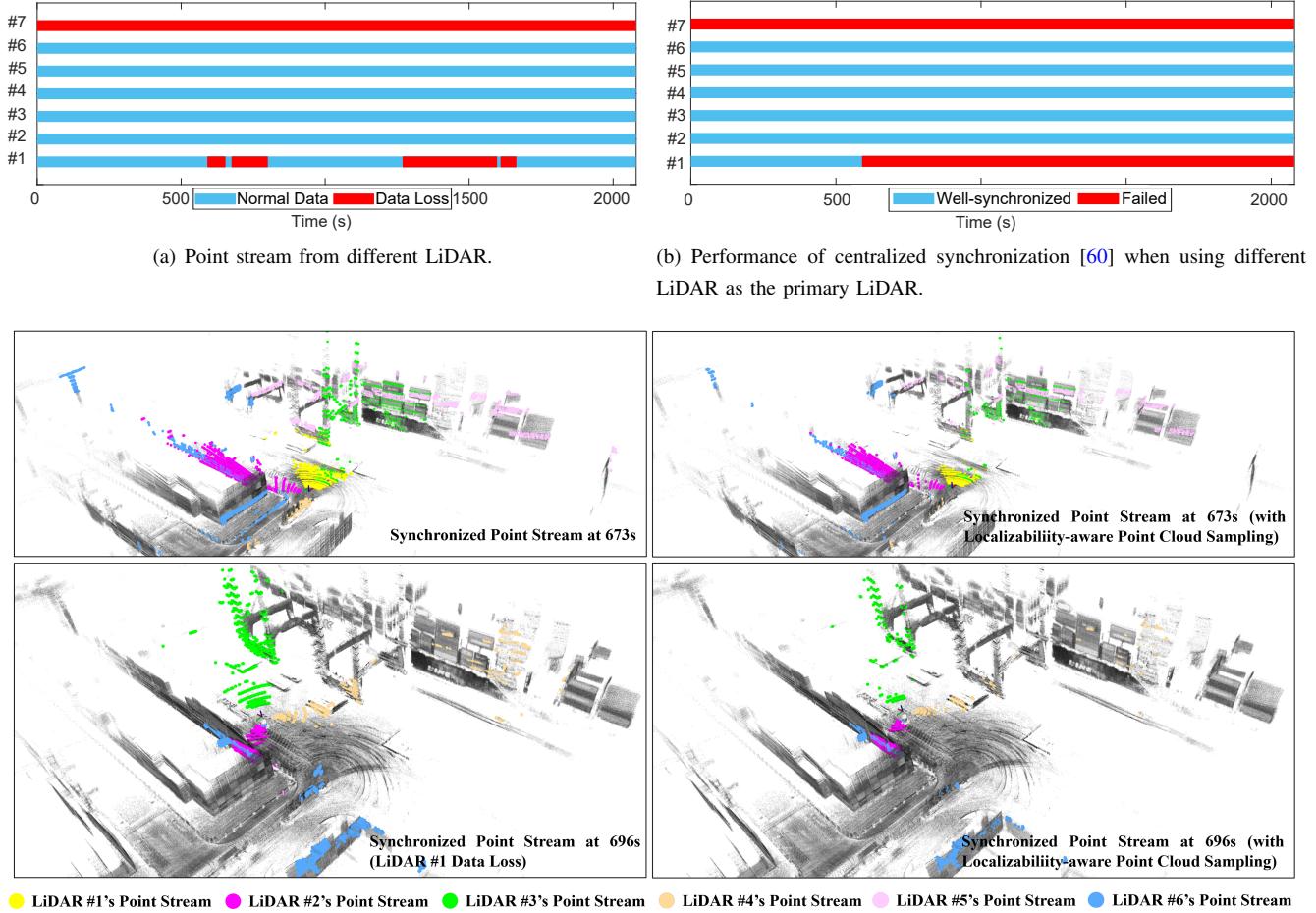


FIGURE 2. Demonstration of proposed multi-LiDAR synchronization and sampling strategy under sequence *Port 01*. LiDAR #1 experienced data loss at 591s – 654s, 675s – 801s, 1269 – 1597s, and 1609 – 1664s due to overly high temperatures, and the data of LiDAR #7 is totally lost due to hardware failure.

with the error state $\tilde{\mathbf{x}}$.

$$\begin{aligned} \min_{\tilde{\mathbf{x}}} \|\mathbf{h}_{\tilde{\mathbf{x}}} + \mathbf{H}'\tilde{\mathbf{x}}\|_{\sigma_R^{-1}}^2 &\triangleq \\ \min_{\tilde{\mathbf{x}}} \tilde{\mathbf{x}}^\top \mathbf{H}'^\top \sigma_R^{-1} \mathbf{H}' \tilde{\mathbf{x}} + 2\mathbf{h}_{\tilde{\mathbf{x}}}^\top \sigma_R^{-1} \mathbf{H}' \tilde{\mathbf{x}} + \mathbf{h}_{\tilde{\mathbf{x}}}^\top \sigma_R^{-1} \mathbf{h}_{\tilde{\mathbf{x}}} \end{aligned} \quad (25)$$

The optimal solution of (25) can thus be obtained by solving $\partial \|\mathbf{h}_{\tilde{\mathbf{x}}} + \mathbf{H}'\tilde{\mathbf{x}}\|_{\sigma_R^{-1}}^2 / \partial \tilde{\mathbf{x}} = 0$.

$$\underbrace{\mathbf{H}'^\top \sigma_R^{-1} \mathbf{H}'}_{\mathbf{A}} \tilde{\mathbf{x}} = -\mathbf{H}'^\top \sigma_R^{-1} \mathbf{h}_{\tilde{\mathbf{x}}} \quad (26)$$

where \mathbf{A} corresponding to the Hessian matrix of (25). Intuitively, the inverse of σ_R provides weight for each LiDAR observation. As the trick used in [14], which set all LiDAR observations to have the same weight, the Hessian matrix

can be simplified as:

$$\mathbf{A}' = \mathbf{H}'^\top \mathbf{H}' = \begin{bmatrix} \mathbf{A}'_{tt} & \mathbf{A}'_{tR} & \mathbf{A}'_{tv} & \mathbf{A}'_{tw} & \mathbf{A}'_{ta} \\ \mathbf{A}'_{Rt} & \mathbf{A}'_{RR} & \mathbf{A}'_{Rv} & \mathbf{A}'_{Rw} & \mathbf{A}'_{Ra} \\ \mathbf{A}'_{vt} & \mathbf{A}'_{vR} & \mathbf{A}'_{vv} & \mathbf{A}'_{vw} & \mathbf{A}'_{va} \\ \mathbf{A}'_{wt} & \mathbf{A}'_{wR} & \mathbf{A}'_{ww} & \mathbf{A}'_{wv} & \mathbf{A}'_{wa} \\ \mathbf{A}'_{at} & \mathbf{A}'_{aR} & \mathbf{A}'_{av} & \mathbf{A}'_{aw} & \mathbf{A}'_{aa} \end{bmatrix} \quad (27)$$

According to the definition of the Hessian matrix for the Gaussian process Kalman filter (27), the LiDAR degeneracy metric [19] for optimization-based estimation can be adopted to the proposed filter-based estimation, which evaluates LiDAR degeneracy through the minimum eigenvalue of the Hessian matrix. Without loss of generality, only the \mathbf{A}'_{tt} and \mathbf{A}'_{RR} are used in degeneracy analysis, because all elements in the state can be expressed by derivatives (from 0 to 2 order) of translation and rotation. \mathbf{A}'_{tt} and \mathbf{A}'_{RR} can be

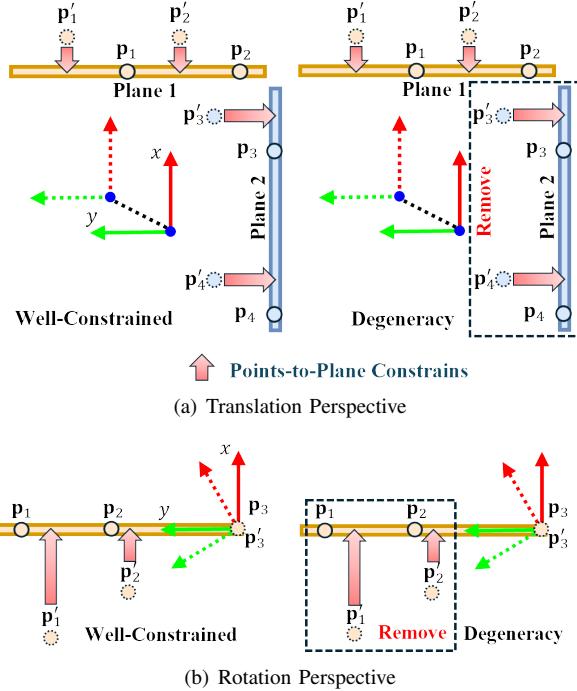


FIGURE 3. An illustration of the localizability concept in 2D.

obtained by substituting (23) into (27).

$$\begin{aligned} \mathbf{A}'_{tt} &= \sum_{i=1}^M \mathbf{n}_j^m \mathbf{n}_j^{m,\top} \\ \mathbf{A}'_{RR} &= - \sum_{i=1}^M [\mathbf{P}_i]_\times \check{\mathbf{R}}_{tk}^\top \mathbf{n}_j^m \mathbf{n}_j^{m,\top} \check{\mathbf{R}}_{tk} [\mathbf{P}_i]_\times \end{aligned} \quad (28)$$

where $\mathbf{P}_i = \exp([\check{\omega}_{tk} \Delta t]_\times) \mathbf{p}_i + \frac{1}{2} \check{\mathbf{a}}_{tk} \Delta t^2$. The translational and rotational contribution for each point \mathbf{p}_i is highly dependent on \mathbf{n}_j^m and $[\mathbf{P}_i]_\times \check{\mathbf{R}}_{tk}^\top \mathbf{n}_j^m$, respectively. Hence, the translational and rotational information matrices, $\mathbf{N}_{tt} \in \mathbb{R}^{M \times 3}$ and $\mathbf{N}_{RR} \in \mathbb{R}^{M \times 3}$, can be defined as:

$$\begin{aligned} \mathbf{N}_{tt} &= [\dots, \mathbf{n}_j^m, \dots]^\top \\ \mathbf{N}_{RR} &= \left[\dots, \frac{[\mathbf{P}_i]_\times \check{\mathbf{R}}_{tk}^\top \mathbf{n}_j^m}{\|[\mathbf{P}_i]_\times \check{\mathbf{R}}_{tk}^\top \mathbf{n}_j^m\|^2}, \dots \right]^\top \end{aligned} \quad (29)$$

Inspired by [20], the translational and rotational contribution for a point cloud \mathbb{P} can be quantified by projecting the information matrices onto the eigenspace of the translation Hessian \mathbf{A}'_{tt} and rotation Hessian \mathbf{A}'_{RR} .

$$\mathbf{C}_{tt} = |\mathbf{N}_{tt} \mathbf{V}_{tt}|, \mathbf{C}_{RR} = |\mathbf{N}_{RR} \mathbf{V}_{RR}| \quad (30)$$

where $|\cdot|$ indicates the element-wise absolute value of the matrix, $\mathbf{V}_{tt} \in \mathbb{R}^{3 \times 3}$ and $\mathbf{V}_{RR} \in \mathbb{R}^{3 \times 3}$ are obtained through the eigen-analysis over \mathbf{A}'_{tt} and \mathbf{A}'_{RR} , and $\mathbf{C}_{tt}, \mathbf{C}_{RR} \in \mathbb{R}^{M \times 3}$ are the translational and rotational contribution matrixes, respectively. The i -th row of $\mathbf{C}_{tt}, \mathbf{C}_{RR}$ corresponds to the localizability contribution provided by $\mathbf{p}_i \in \mathbb{P}$. To quantify the contribution of 6-DOF, six information pairs $\mathcal{L}_d(i) = \{i, \mathbf{C}(i, d)\}, d = 1, \dots, 6$ are defined for each point

\mathbf{p}_i , where $\mathbf{C} = [\mathbf{C}_{tt}, \mathbf{C}_{RR}]$. Hence, according to the sorting of the contributions for each point \mathbf{p}_i to each DOF d , points are added to the selected point set \mathbb{P}_s in descending order until the contributions of the points in the selected point set \mathbb{P}_s satisfy the constraints defined in (31) for all 6-DOF.

$$S_d > s, S_d = \sum_{\mathbf{p}_i \in \mathbb{P}_s} \mathbf{C}(i, d), d = 1, \dots, 6 \quad (31)$$

where s is the localizability threshold.

Considering the definition of the contribution matrix (30) relies on the point-to-voxel correspondence information, which process is expensive, especially when repeatedly computed within iterations. To tackle this problem, a normal estimation algorithm¹ is adopted to approximate the normal vector $\mathbf{n}_j^m \approx \check{\mathbf{R}}_{tk} \mathbf{n}_i$ used in (28) and (29). Hence, (28) and (29) can be rewritten as:

$$\begin{aligned} \mathbf{A}'_{tt} &\approx \check{\mathbf{R}}_{tk} \sum_{i=1}^M \mathbf{n}_i \mathbf{n}_i^\top \check{\mathbf{R}}_{tk}^\top \\ \mathbf{A}'_{RR} &\approx - \sum_{i=1}^M [\mathbf{P}_i]_\times \mathbf{n}_i \mathbf{n}_i^\top [\mathbf{P}_i]_\times \\ \mathbf{N}_{tt} &\approx [\dots, \check{\mathbf{R}}_{tk} \mathbf{n}_i, \dots]^\top, \mathbf{N}_{RR} \approx \left[\dots, \frac{[\mathbf{P}_i]_\times \mathbf{n}_i}{\|[\mathbf{P}_i]_\times \mathbf{n}_i\|^2}, \dots \right]^\top \end{aligned} \quad (32)$$

which process does not rely on the point-to-voxel correspondence information. The process of the localizability-aware point cloud sampling is given in Algorithm 1.

Remark 3:

(Localizability-aware sampling) Based on the derivation on the Hessian matrix (from (24) to (27)), we expand the LiDAR degeneracy metric [19], [20] for optimization-based estimation to the analysis of filter-based LiDAR odometry and quantify the localizability contribution for each point $\mathbf{p}_i \in \mathbb{P}$ in (28)-(30). Thanks to above derivations, the proposed point cloud sampling strategy has the capability to modify the point number online with the consideration of localizability, so-called localizability-aware point cloud sampling. In [20], [80], the point sampling is developed with the prior knowledge of the point-to-map matching information, which process is time-consuming due to the LiDAR matching having to be performed for each iteration. To tackle this problem, in this paper, a novel point sampling strategy is proposed without knowing the correspondences beforehand while considering the localizability.

V. Voxel Map Management

An adaptive voxelization method [51], [71] is adopted to maintain the voxel map \mathbb{M} for CTE-MLO with a hash data structure. Each voxel $\mathbf{m}_j \in \mathbb{M}$ is clustered from a set of points $\mathbf{p}_i (i = 1, 2, \dots, M_j)$ sampled at different times but belongs to the same plane in three-dimensional spatial space, which is indexed in a hash table. With the Gaussian

¹https://pcl.readthedocs.io/projects/tutorials/en/latest/normal_estimation.html

Algorithm 1: Localizability-aware sampling

Input : \mathbb{P} : Newly obtained point cloud,
 s : Localizability threshold.

Output: \mathbb{P}_s : The selected point cloud.

```

1 for each  $\mathbf{p}_i \in \mathbb{P}$  do
2    $\mathbf{n}_i \leftarrow \text{NormalEstimation}(\mathbf{p}_i)$ 
3   Update  $\mathbf{A}'_{tt}, \mathbf{A}'_{RR}, \mathbf{N}_{tt}, \mathbf{N}_{RR}$  by substituting  $\mathbf{n}_i$  into (32)
4 Singular Value Decomposition (SVD):
   $\mathbf{A}'_{tt} = \mathbf{V}_{tt}\Lambda_{tt}\mathbf{V}_{tt}^\top, \mathbf{A}'_{RR} = \mathbf{V}_{RR}\Lambda_{RR}\mathbf{V}_{RR}^\top$ 
5 Get the contribution matrix  $\mathbf{C} = [\mathbf{C}_{tt}, \mathbf{C}_{RR}]$  by
  substituting  $\mathbf{N}_{tt}, \mathbf{N}_{RR}, \mathbf{V}_{tt}, \mathbf{V}_{RR}$  into (30)
6 for  $d \leftarrow 1$  to 6 do
  //  $\text{Card}(\cdot)$  returns the cardinal
  number of a set
  7 for  $i \leftarrow 1$  to  $\text{Card}(\mathbb{P})$  do
    Build information pair  $\mathcal{L}_d(i) = \{i, \mathbf{C}(i, d)\}$ 
  8 Sort  $\mathcal{L}_d$  according to  $\mathbf{C}(\dots, d)$ 
  9 for  $i \leftarrow 1$  to  $\text{Card}(\mathbb{P})$  do
    10    $S_d = S_d + \mathcal{L}_d(i).\text{second}$ 
    11   if  $\mathbf{p}_i \notin \mathbb{P}_s$  then
      12     Add  $\mathbf{p}_i$  into  $\mathbb{P}_s$ 
    13   if  $S_d > s$  then
      14     Break
  15
16 return  $\mathbb{P}_s$ 
```

distribution assumption, a voxel $\mathbf{m}_j \sim \mathcal{N}(\boldsymbol{\mu}_j^m, \boldsymbol{\sigma}_j^m)$ can be described using a mean vector and a covariance matrix.

$$\begin{aligned} \boldsymbol{\mu}_j^m &= \frac{1}{M_j} \sum_{i=1}^{M_j} {}^W \mathbf{p}_i \\ \boldsymbol{\sigma}_j^m &= \frac{1}{M_j} \sum_{i=1}^{M_j} ({}^W \mathbf{p}_i - \boldsymbol{\mu}_j^m) ({}^W \mathbf{p}_i - \boldsymbol{\mu}_j^m)^\top \\ &= \frac{1}{M_j} \sum_{i=1}^{M_j} {}^W \mathbf{p}_i {}^W \mathbf{p}_i^\top - \boldsymbol{\mu}_j^m \boldsymbol{\mu}_j^{m,\top} \end{aligned} \quad (33)$$

Hence, each voxel in the voxel map is represented in a point-free formulation by a centroid $\boldsymbol{\mu}_j^m$, a covariance matrix $\boldsymbol{\sigma}_j^m$, and the number of points M_j that contribute to cluster the voxel \mathbf{m}_j . The voxel map stores all voxels in a hash table, and the hash key of each voxel is calculated as follows:

$$K_j = H\left(\left\lfloor \frac{\boldsymbol{\mu}_j^m}{r} \right\rfloor\right) \quad (34)$$

where $H(\cdot)$ is a function that calculates the hash index $K_j \in \mathbb{R}$ using the centroid of the voxel $\boldsymbol{\mu}_j^m \in \mathbb{R}^3$, r is the resolution of the voxel map, and $\lfloor \cdot \rfloor$ denotes the floor operation.

For a newly obtained point $\mathbf{p}_i(t_i \in [t_k, t_{k+1}])$, we first transform it into the inertial frame ${}^W \mathbf{p}_i = \bar{\mathbf{R}}(t_i) \mathbf{p}_i + \bar{\mathbf{t}}(t_i)$. The corresponding rotation $\mathbf{R}(t_i)$ and translation $\mathbf{t}(t_i)$ are

estimated through the Gaussian process continuous-time LiDAR odometry (Section III). Then, the corresponding voxel of \mathbf{p}_i can be associated with the hash key $K_i = H\left(\left\lfloor \frac{{}^W \mathbf{p}_i}{r} \right\rfloor\right)$. Inspired by [56], the corresponding voxel can be updated incrementally.

$$\begin{aligned} \boldsymbol{\mu}_{j,\text{new}}^m &= \frac{1}{M_{j,\text{new}}} \left(M_j \boldsymbol{\mu}_j^m + \sum_{\mathbf{p}_i \in \mathbb{P}_j} {}^W \mathbf{p}_i \right) \\ \boldsymbol{\sigma}_{j,\text{new}}^m &= \frac{1}{M_{j,\text{new}}} \left(M_j (\boldsymbol{\sigma}_j^m + \boldsymbol{\mu}_{j,\text{new}}^m \boldsymbol{\mu}_{j,\text{new}}^{m,\top}) + \sum_{\mathbf{p}_i \in \mathbb{P}_j} {}^W \mathbf{p}_i {}^W \mathbf{p}_i^\top \right. \\ &\quad \left. - \boldsymbol{\mu}_{j,\text{new}}^m \boldsymbol{\mu}_{j,\text{new}}^{m,\top} \right) \end{aligned} \quad (35)$$

where $M_{j,\text{new}} = M_j + \text{Card}(\mathbb{P}_j)$, $\mathbb{P}_j \in \mathbb{P}$ is the newly obtained point cloud that corresponds to the j -th voxel, and $\boldsymbol{\mu}_{j,\text{new}}^m$, $\boldsymbol{\sigma}_{j,\text{new}}^m$, and M_{new} denote the updated centroid, covariance matrix, and number of points belonging to the j -th voxel, respectively.

After the voxel map update, voxels can be divided into different shapes depending on the relationships between the eigenvalues of the covariance matrix $\boldsymbol{\sigma}_j^m$. The SVD (36) is performed to select the plane voxel that satisfies $\lambda_2 \gg \lambda_3$, and the normal vector of a plane voxel is defined as $\mathbf{n}_j^m = \mathbf{V}_3$.

$$\boldsymbol{\sigma}_j^m = [\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3] \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} [\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]^\top \quad (36)$$

where λ_1 , λ_2 , and λ_3 are eigenvalues of the covariance matrix $\boldsymbol{\sigma}_j^m$ in descending order. \mathbf{V}_1 , \mathbf{V}_2 , and \mathbf{V}_3 are eigenvectors that correspond to λ_1 , λ_2 , and λ_3 , respectively.

VI. Experimental Evaluations and Validations

In this section, the effectiveness of the proposed CTE-MLO is demonstrated through various scenarios, including public datasets and real-world experiments.

Remark 4:

(Extendability) The continuous-time trajectory formulation grants CTE-MLO exceptional extendability, allowing each point in a point stream (within the continuous-time trajectory period) to independently query the trajectory without synchronizing all points from different LiDARs to an instantaneous time. To demonstrate the extendability of CTE-MLO, in this section, various experiments are conducted with different LiDAR configurations, including spinning LiDAR (Ouster OS1, Robosense Helios 16/32), non-repetitive scan LiDAR (Livox Mid 360), and solid-state LiDAR (Robosense M1, Robosense Bpearl, Realsense L515).

A. Quantitative Evaluation on the NTU VIRAL Datasets

The proposed CTE-MLO is evaluated quantitatively on the NTU VIRAL dataset [81], which provides point clouds captured by two lightweight ouster OS1 LiDARs (deployed

in horizontal form and vertical form respectively), and ground truth from the Leica laser tracker. Each experiment is conducted on a computer equipped with an Intel Core i9-13900KF. The comparative methods include SOTA MLO MLOAM [36] and a continuous-time LiDAR odometry MARS [58]. For a fair comparison, we merge the LiDAR point from two OS1 LiDAR into a single LiDAR scan through the LiDAR synchronization package² provided by [60], which enables MARS [58] as a MLO. It is worth noting that all methods are implemented without loop closure. Moreover, an ablation study on the proposed method is performed to understand the influence of the number of LiDAR and localizability-aware point cloud sampling. We implement the proposed algorithm in various configurations, including 1) CT-LO: a continuous-time single LiDAR odometry developed through the Kalman filter-based Gaussian process estimation (Section III). Horizontal LiDAR is used for CT-LO; 2) CT-MLO: a continuous-time MLO that utilizes LiDAR points from multiple LiDARs using the decentralized multi-LiDAR synchronization scheme (Section IV-A); 3) CTE-MLO: a continuous-time and efficient MLO with the localizability-aware point cloud sampling illustrated in Section IV-B, which is the full algorithm proposed in this paper.

1) Ablation Study

To illustrate the effectiveness of the multi-LiDAR fusion, an ablation study over single LiDAR odometry (CT-LO) and multi-LiDAR odometry (CT-MLO and CTE-MLO) is conducted on the NTU VIRAL dataset [81]. Localization for a MAV flying at a high altitude is challenging because the horizontal LiDAR can only provide observation around a MAV with very limited points belonging to the ground or ceiling. As shown in Fig. 4(a), the scarcity of geometric constraint over the z -axis leads CT-LO can not provide accurate estimation for MAV when compared with the multi-LiDAR version of the proposed method called CT-MLO and CTE-MLO. Thanks to the decentralized multi-LiDAR synchronization scheme introduced in Section IV-A, CT-MLO outperformed CT-LO in accuracy especially along the z -axis by fully leveraging the geometric information observed from both horizontal LiDAR and vertical LiDAR. To quantitatively evaluate the accuracy of MLO, the Absolute Trajectory Error (ATE) results of each method over NTU VIRAL datasets are shown in Table 1. CT-MLO exhibits excellent accuracy on 17 out of 18 sequences of the NTU VIRAL dataset. However, the large-size point cloud generated by multiple LiDARs worsens the computational complexity. As shown in Fig. 4(b), despite the average time consumption of CT-MLO being less than the time interval $\Delta t = 10ms$, the total time consumption of CT-MLO has a large number of outliers greater than Δt . This leads to the failure of CT-MLO in sequence *spms02* due to CT-

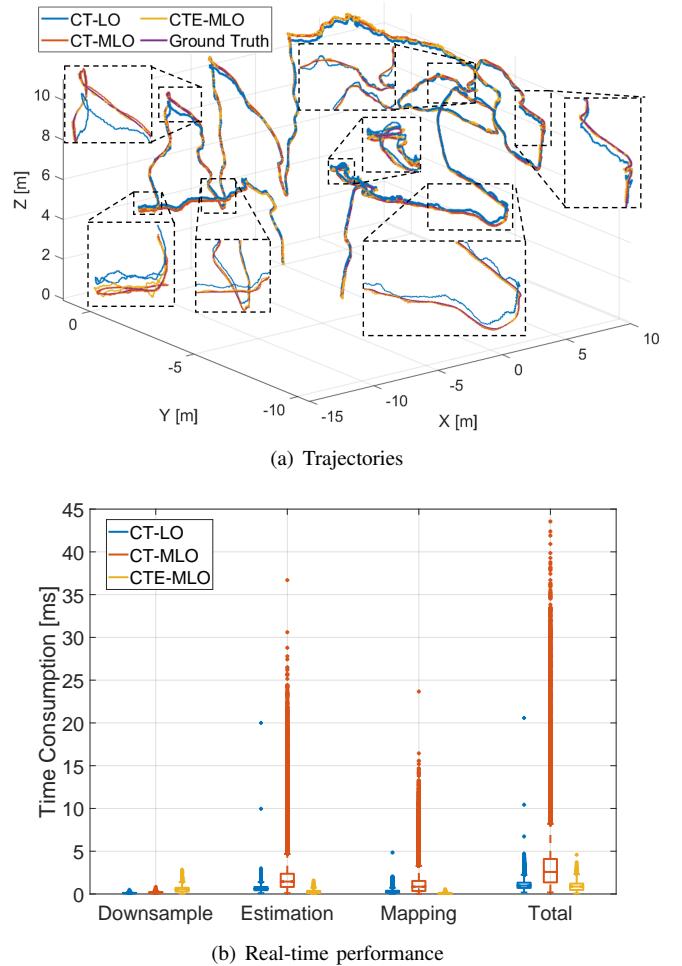


FIGURE 4. Benchmark results of CT-LO, CT-MLO, and CTE-MLO under the *tnp02* sequence of the NTU VIRAL dataset.

MLO has to drop a lot of LiDAR scans to achieve real-time performance. Hence, motivated by developing a real-time and accurate MLO, CTE-MLO is proposed by introducing the localizability-aware point cloud sampling strategy into CT-MLO. As shown in Fig. 4(b), the localizability-aware point cloud sampling strategy can significantly improve the estimation efficiency, and the time consumption of CTE-MLO is strictly less than the time interval Δt , which means CTE-MLO can achieve real-time without dropping any scan.

To justify the effectiveness of the localizability-aware point cloud sampling, as shown in Fig. 5, the localizability and used points of CT-LO, CT-MLO, and CTE-MLO are compared under one of the degenerate sequences over the NTU-VIRAL dataset. For visualization, the point stream shown in Fig. 5(c) comprises points accumulated within 0.1s. From the top row of Fig. 5(c), there are almost no points belonging to the ground or ceiling to provide z -axis constraints for CT-LO when the MAV is flying at a relatively high altitude, which is attributed to the limited vertical FOV of the horizontal LiDAR. CT-MLO prevents the degeneration of

²<https://github.com/brytsknguyen/slict/blob/master/src/MergeLidar.cpp>

TABLE 1. Absolute Trajectory Error (ATE, Meters) over NTU VIRAL dataset.

Method	eee01	eee02	eee03	nya01	nya02	nya03	sbs01	sbs02	sbs03	rtp01	rtp02	rtp03	spms01	spms02	spms03	tpp01	tpp02	tpp03	Average
CT-LO	0.08	0.07	0.11	0.05	0.09	0.11	0.09	0.08	0.08	-	0.13	0.15	0.22	0.31	-	-	0.12	0.10	0.12
CT-MLO	0.07	0.07	0.11	0.05	0.08	0.10	0.09	0.08	0.08	0.13	0.13	0.14	0.21	-	0.20	0.08	0.07	0.08	0.10
CTE-MLO	0.08	0.07	0.12	0.06	0.09	0.10	0.09	0.08	0.09	0.13	0.14	0.14	0.21	0.33	0.20	0.09	0.09	0.10	0.12
MLOAM	0.23	0.16	0.23	0.13	0.19	0.22	0.18	0.16	0.17	0.41	0.34	1.04	0.79	11.72	6.11	0.15	0.11	0.15	1.25
MARS	0.19	0.08	0.10	0.06	0.06	0.08	0.13	0.14	0.20	0.15	0.08	0.14	0.94	-	-	0.08	0.09	0.11	0.16

¹ - denotes the system failure, and the best result is highlighted in Blue.

CT-LO by leveraging the merged point stream observations from horizontal LiDAR and vertical LiDAR. As shown in Fig. 5(a), although CT-MLO ensures the localizability using the observation from multiple LiDAR, the dense point cloud preserves a high level of geometric constraint redundancy (the localizability of CT-MLO is much greater than the empirical localizability threshold $s = 500$). This redundancy unnecessarily slows down the CT-MLO and may cause estimation failure under real-time constraints. Thanks to the localizability-aware point cloud sampling strategy, which quantifies the localizability contribution for each point, CTE-MLO can ensure localizability using a minimal number of points with high localizability contributions. This enables CTE-MLO to avoid degeneration by using fewer points compared to CT-LO and CT-MLO (Fig. 5(b)). As shown in Fig. 5(a) and Fig. 5(c), the localizability-aware point cloud sampling strategy defined in Section IV-B can effectively evaluate the localizability of LiDAR odometry. When the MAV is flying at a relatively high altitude, the proposed localizability-aware point cloud sampling strategy can detect the degeneration of CT-LO. When the MAV flies at low altitudes, CT-LO and CT-MLO exhibit similar localizability because the horizontal LiDAR can provide sufficient ground observations (as shown in the bottom row of Fig. 5(c)). Thanks to the integration of tightly coupled multi-LiDAR estimation and the efficient localizability-aware point cloud sampling strategy, CTE-MLO exhibits excellent localization and mapping consistency across the 18 data sequences and reaches a 54% estimation efficiency improvement when compared with the CT-MLO (as shown in Table 2).

2) Accuracy evaluation

Table 1 summarized the ATE of the proposed method and SOTA LiDAR odometry, including the SOTA MLO MLOAM [36] and multi-LiDAR-augmented continuous-time LiDAR odometry MARS [58], over the NTU VIRAL dataset [81]. The average accuracy of CTE-MLO outperforms both MLOAM [36] and MARS [58]. Thanks to the Gaussian process Kalman filter derived in Section III, which considers the continuous-time sampling characteristics of LiDAR in a Kalman filter, CTE-MLO can estimate the trajectory using point stream from multiple LiDARs without estimating the state of prime LiDAR using the point cloud undistortion technique like MLOAM [36]. When compared with

MARS [58], CTE-MLO achieves continuous-time estimation using Gaussian process regression, which dispenses with the estimation of control points not belong to the current time domain. Due to future control points estimation being inevitable for the B-spline-based continuous-time estimator MARS [58], a keyframe strategy is adopted in MARS [58] to enable continuous-time trajectory estimation in real time. Nonetheless, the keyframe strategy is a trade-off between accuracy and real-time performance, which also leads to failure during aggressive rotational and translational motions in *spms02* and *spms03*. Benefited by Gaussian process Kalman filter and localizability-aware point cloud sampling scheme, CTE-MLO can provide consistent estimation over 18 sequences of NTU VIRAL dataset, and the average ATE of CTE-MLO achieves 0.122m which reaches a 25% – 90% accuracy improvement over MARS [58] and MLOAM [36]. The real-time reconstructed 3D map is shown in Fig. 6. The high-quality point cloud demonstrates that the CTE-MLO is highly accurate and is able to provide high-precision reconstruction using LiDAR point cloud without any extra sensors, such as IMU, camera, and GNSS.

3) Running time evaluation

Directly comparing the processing time of the proposed method with SOTA methods would be unfair due to the adoption of the LiDAR splitting technique in the multi-LiDAR synchronization, which technique significantly reduces the dimension of the Jacobian matrix used for Hessian construction. Therefore, the decentralized multi-LiDAR synchronization scheme implicit a high-level compression of computational complexity in the state update process (13)-(14). For a fair comparison, the efficiency is measured by the ratio between processing time t_{process} and the time interval ($\Delta t = 10\text{ms}$ for CT-LO, CT-MLO, and CTE-MLO; $\Delta t = 100\text{ms}$ for MLOAM [36] and MARS [58]).

$$\text{Efficiency} = t_{\text{process}} / \Delta t \quad (37)$$

Table 2 shows the average efficiency for processing one scan of all the benchmark methods. The efficiency of each component, including sampling (“DS” in Table 2), estimation (“ES” in Table 2), and mapping (“Map” in Table 2), are compared across all 18 sequences. Moreover, the total efficiency (“Total” in Table 2) is also measured by the ratio between total processing time (gathered all the

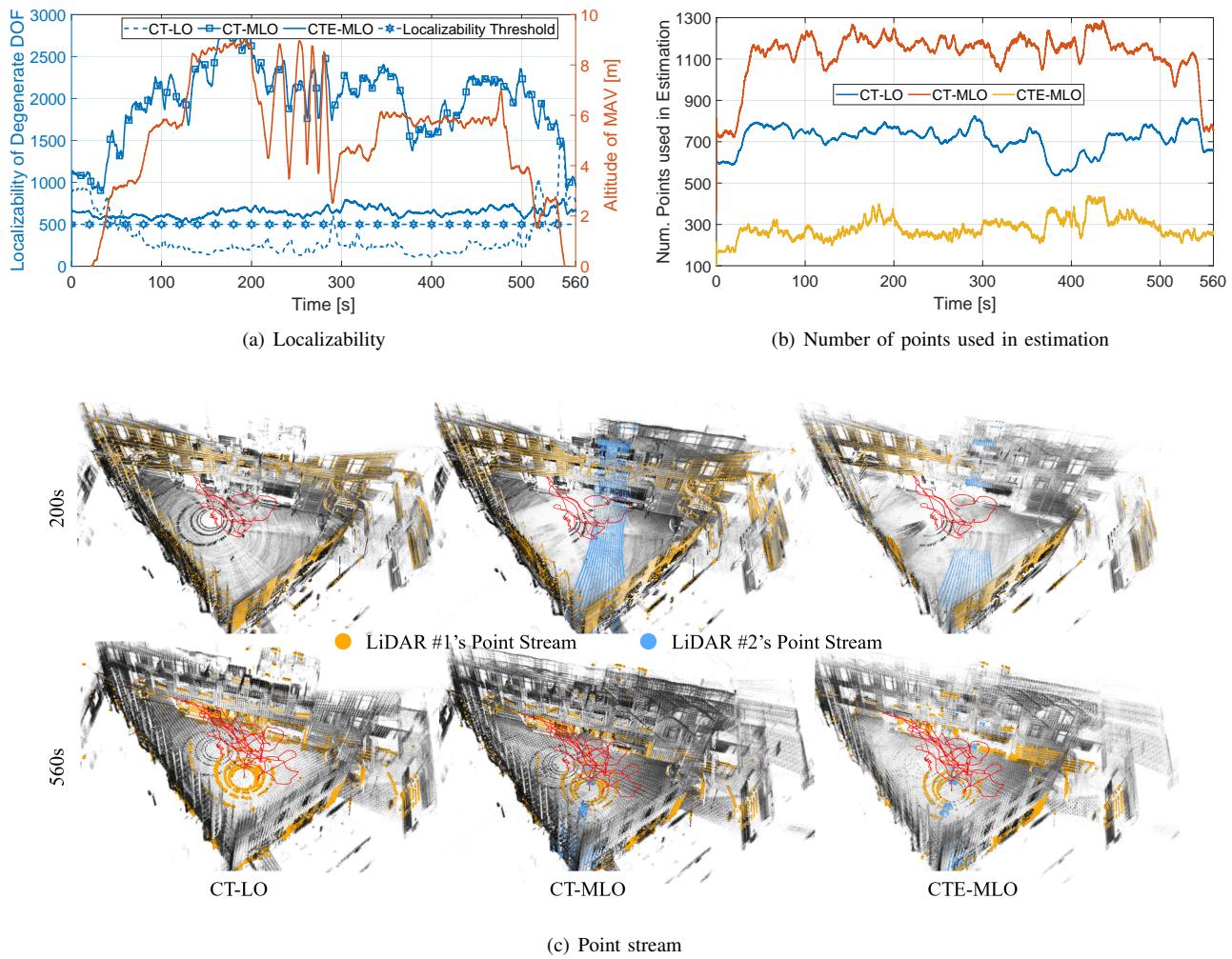


FIGURE 5. Localizability and used points of CT-LO, CT-MLO, and CTE-MLO under a degenerate sequence (*tmp01*) of the NTU VIRAL dataset.

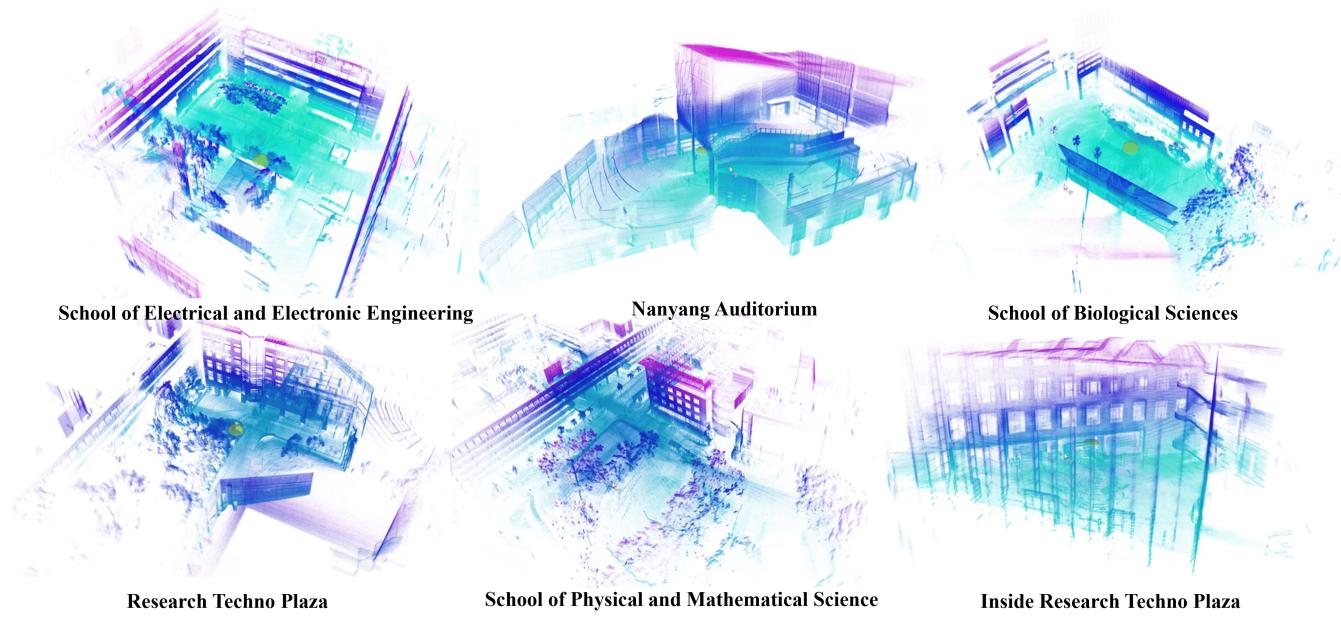


FIGURE 6. Reconstruction result of a part of NTU landmarks using the proposed CTE-MLO on the NTU VIRAL dataset.

TABLE 2. Efficiency for processing one scan over the NTU VIRAL dataset.

	CT-LO				CT-MLO				CTE-MLO				MLOAM				MARS			
	Total	DS	ES	Map	Total	DS	ES	Map	Total	DS	ES	Map	Total	DS	ES	Map	Total	DS	ES	Map
eee01	0.17	0.01	0.11	0.06	0.42	0.01	0.25	0.16	0.12	0.06	0.04	0.02	1.56	0.45	0.62	0.49	0.21	0.00	0.16	0.04
eee02	0.12	0.01	0.08	0.03	0.17	0.01	0.11	0.05	0.11	0.06	0.03	0.01	1.04	0.26	0.55	0.23	0.18	0.00	0.15	0.03
eee03	0.10	0.01	0.07	0.03	0.15	0.01	0.10	0.04	0.10	0.06	0.03	0.01	1.12	0.38	0.53	0.21	0.20	0.00	0.15	0.04
nya01	0.16	0.01	0.10	0.05	0.11	0.01	0.07	0.02	0.09	0.05	0.03	0.01	0.94	0.31	0.49	0.12	0.17	0.00	0.15	0.02
nya02	0.07	0.01	0.05	0.01	0.11	0.01	0.08	0.03	0.09	0.05	0.03	0.01	0.90	0.26	0.51	0.12	0.18	0.00	0.16	0.02
nya03	0.07	0.01	0.05	0.02	0.33	0.01	0.21	0.11	0.12	0.06	0.04	0.01	1.09	0.30	0.51	0.28	0.19	0.00	0.15	0.03
sbs01	0.09	0.01	0.05	0.02	0.14	0.01	0.09	0.04	0.09	0.05	0.03	0.01	0.92	0.26	0.45	0.21	0.16	0.00	0.13	0.02
sbs02	0.09	0.01	0.06	0.02	0.12	0.01	0.08	0.03	0.10	0.06	0.03	0.01	1.13	0.41	0.47	0.24	0.15	0.00	0.13	0.02
sbs03	0.14	0.01	0.09	0.04	0.19	0.01	0.12	0.06	0.10	0.06	0.03	0.01	1.10	0.37	0.47	0.25	0.16	0.00	0.13	0.02
rtp01	-	-	-	-	0.31	0.02	0.18	0.11	0.14	0.07	0.05	0.02	1.89	0.82	0.54	0.53	0.27	0.00	0.22	0.05
rtp02	0.13	0.01	0.08	0.04	0.19	0.02	0.12	0.05	0.13	0.06	0.04	0.02	1.38	0.51	0.50	0.37	0.26	0.00	0.22	0.03
rtp03	0.07	0.01	0.04	0.01	0.21	0.02	0.13	0.06	0.13	0.07	0.04	0.02	1.43	0.55	0.50	0.38	0.26	0.00	0.20	0.04
spms01	0.23	0.01	0.13	0.08	0.36	0.01	0.22	0.12	0.17	0.08	0.06	0.03	1.77	0.61	0.52	0.64	0.28	0.00	0.20	0.08
spms02	0.28	0.00	0.18	0.10	-	-	-	-	0.18	0.07	0.07	0.04	1.43	0.36	0.42	0.64	-	-	-	-
spms03	-	-	-	-	0.37	0.01	0.22	0.13	0.14	0.07	0.04	0.02	1.24	0.35	0.49	0.39	-	-	-	-
tnp01	-	-	-	-	0.11	0.02	0.07	0.02	0.09	0.05	0.03	0.01	0.77	0.25	0.39	0.12	0.20	0.00	0.18	0.01
tnp02	0.10	0.01	0.07	0.02	0.37	0.02	0.21	0.14	0.11	0.06	0.03	0.01	0.74	0.12	0.40	0.21	0.19	0.00	0.17	0.01
tnp03	0.15	0.01	0.09	0.05	0.74	0.02	0.44	0.29	0.10	0.06	0.03	0.01	0.65	0.10	0.38	0.16	0.19	0.00	0.17	0.01
Average	0.13	0.01	0.08	0.04	0.26	0.01	0.16	0.09	0.12	0.06	0.04	0.02	1.17	0.37	0.49	0.31	0.20	0.00	0.17	0.03

¹ - denotes the system failure, and the best result is highlighted in Blue.

time consumption for processing one scan, including the point cloud sampling, estimation, and mapping) and the time interval Δt . The localizability-aware point cloud sampling allows CTE-MLO to select points with high localizability contribution, which induces a high-level compression of the dense point cloud merged from multiple LiDARs. As noted in Section VI-A.1 (Ablation Study), the localizability-aware point cloud sampling enables CTE-MLO to achieve 75% and 78% efficiency improvements in estimation and mapping, respectively, compared to CT-MLO. Besides the localizability-aware point cloud sampling, combining the Gaussian process regression with the Kalman filter can also improve the estimation efficiency of LiDAR odometry, which enables continuous-time estimation within only a few linear iterations. To demonstrate this, the estimation efficiency of CT-MLO is compared with that of MLOAM [36] and MARS [58], both of which utilize nonlinear solvers for estimation. Compared to MLOAM [36], CT-MLO improves estimation efficiency by more than 67%. Even compared to MARS [58], using voxel-to-voxel registration, CT-MLO demonstrates better estimation efficiency with point-to-voxel registration. Benefiting from the combination of the Gaussian process regression with the Kalman filter and the localizability-aware point cloud sampling, as results are shown in Table 2, CTE-MLO achieves the best average estimation and mapping efficiency when compared with SOTA MLO over all 18 sequences. CTE-MLO achieves 43%–90% efficiency improvement compared to MARS [58] and MLOAM [36], respectively. The average efficiency of CTE-MLO achieves 0.116 which means the proposed method only requires

11.6% of the time interval $\Delta t = 10ms$ to realize localization and mapping.

B. Real-world Applications

To attest to the practicality, CTE-MLO is applied to enable estimation and mapping for three autonomous platforms, called truck, sweeper, and MAV. As shown in Fig. 7, both truck and sweeper are equipped with multiple heterogeneous LiDARs to enable panoramic LiDAR observation in horizontal view and onboard computer with an Intel Core i9-13900H CPU. Different from the ground vehicles, which are not sensitive to the weight and power of onboard equipment, two light-weight solid-state LiDARs are mounted on MAV, and an edge computation platform (Nvidia Orin NX) is devoted as the onboard computer. We conduct multiple validations of the proposed method in three scenarios, including seaport, campus, and field forest environments. As the detail of real-world application sequences shown in Table 3, the total validation duration reaches 3 hours, covering a total distance of over 44 kilometers.

1) Truck Localization in Large-scale Port Scenario

To attest to the practicality, we applied the CTE-MLO to an autonomous driving truck and conducted over 32km of driving tests in three distinct areas within a large-scale port scenario. The total area of the three test regions reaches $1.182km^2$. As shown in Fig. 7(a), the truck is equipped with 4 mechanical LiDAR (1 Ouster OS1 and 3 Robosense Helios 32) and 3 Robosense M1 solid-state LiDAR. Port

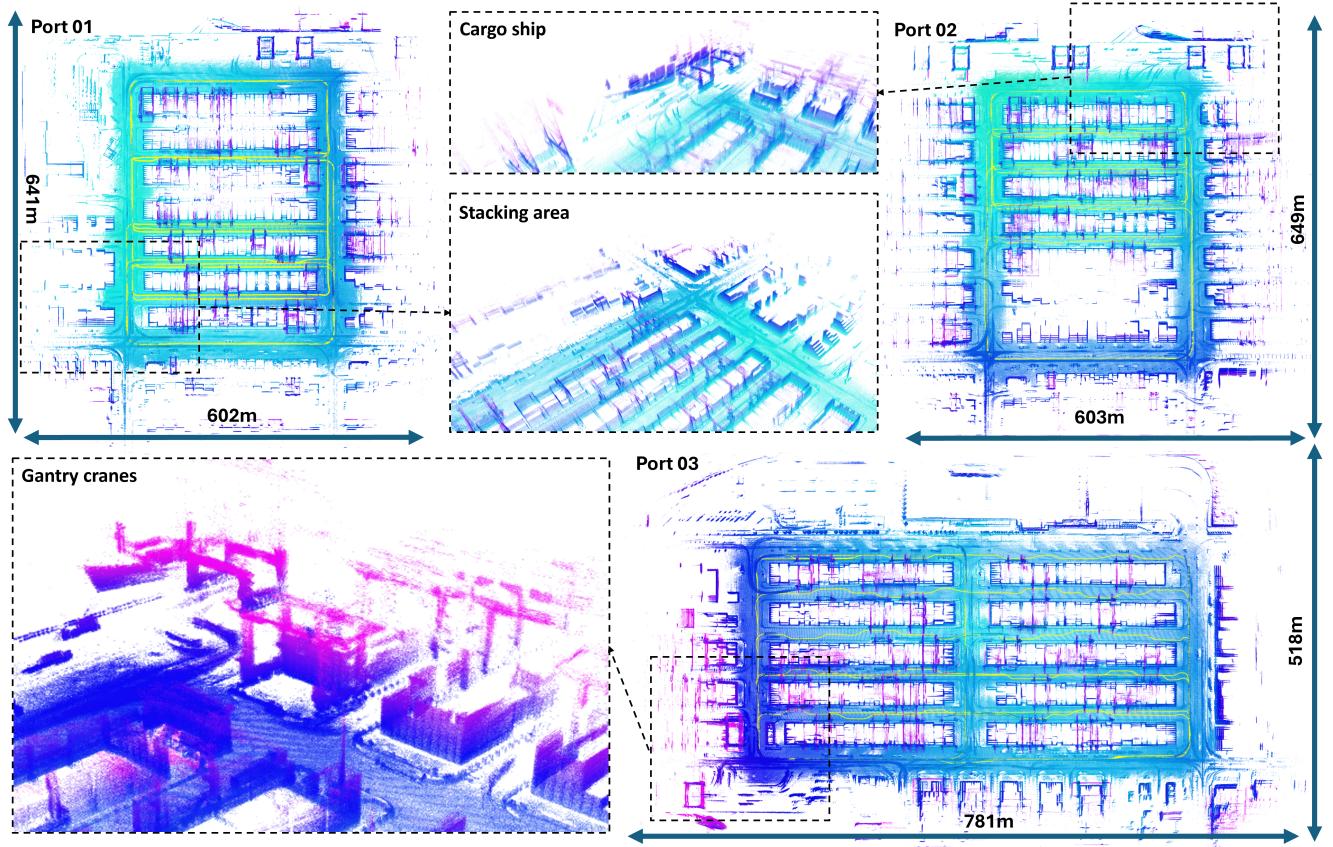
**FIGURE 7.** Platforms used in real-word applications.**TABLE 3.** Data sequences information used in real-world applications.

Scenario	Sequence	Distance (m)	Druation (min:sec)	LiDARs	LiDAR Data Stream
Campus	Campus 01	966.0	07:24	3 ML+1 SSL	✓
	Campus 02	4420.6	21:10	3 ML+1 SSL	✓
	Campus 03	5467.3	26:12	3 ML+1 SSL	✓
Port	Port 01	12148.4	34:40	4 ML+3 SSL	LiDAR #1: 591-654s, 675-801s, 1269-1597s, 1609-1664s. LiDAR #7: ✗
	Port 02	9796.7	44:48	4 ML+3 SSL	LiDAR #7: ✗
	Port 03	11009.7	44:57	4 ML+3 SSL	LiDAR #5: ✗ LiDAR #7: ✗
Forest	Forest 01	526.0	05:14	2 SSL	✓

¹ ML and SSL are denoting Mechanical LiDAR and Solid State LiDAR, respectively.

² In sequence Port 01, LiDAR #1 experienced data lose due to overly high temperatures.

³ ✓ denotes that all LiDARs equipped on the platform are working fine, and ✗ denotes the data of LiDAR is totally lost due to the hardware failure.

**FIGURE 8.** The reconstruction result of CTE-MLO in a port scenario.

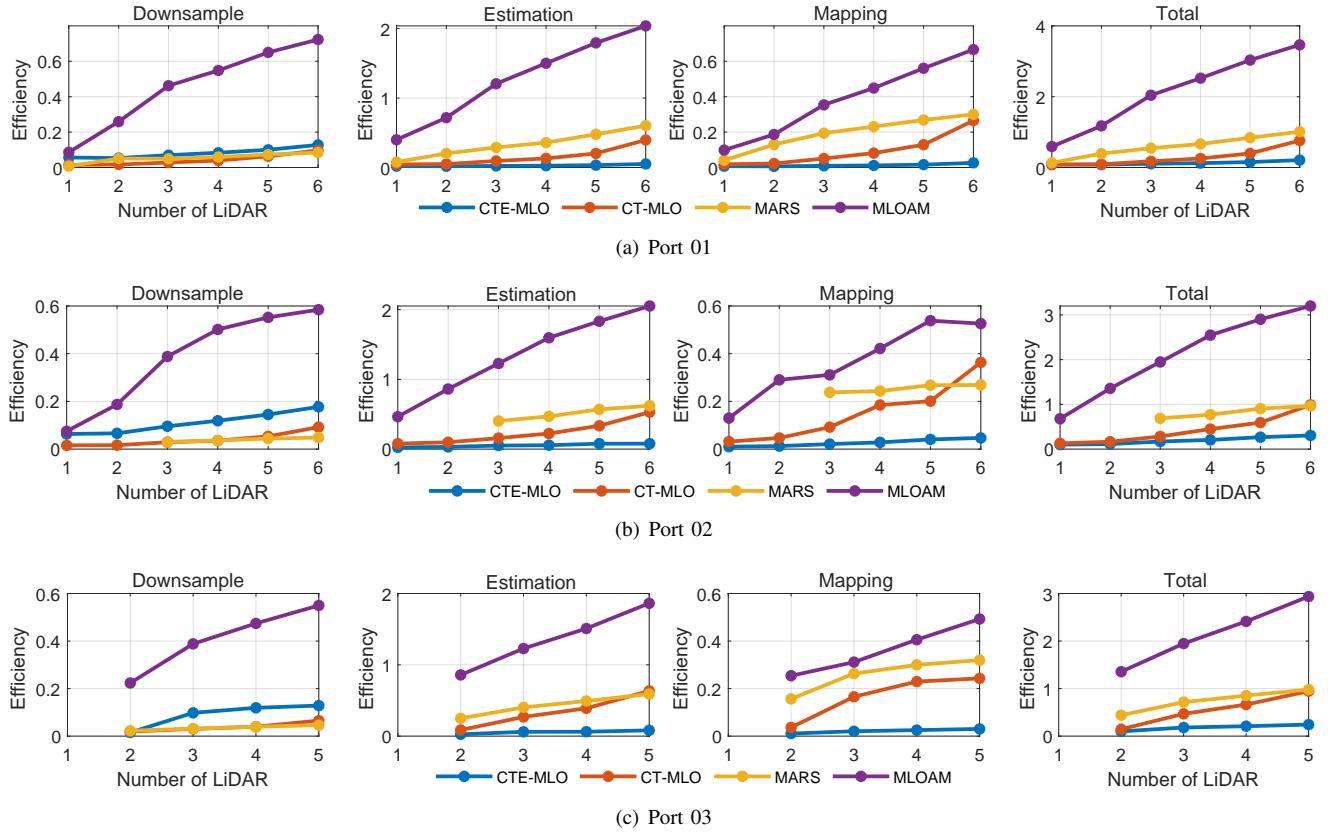


FIGURE 9. Benchmark results of efficiency for each individual component with different numbers of LiDARs in large-scale port scenario.

scenarios are challenging for SLAM methods due to the presence of numerous moving objects (such as trucks and container cranes) and point cloud occlusions. Thanks to the decentralized multi-LiDAR synchronization strategy proposed in Section IV-A, CTE-MLO can merge LiDAR points sampled from different LiDAR into one scan to provide panorama observation. The high-quality point cloud shown in Fig. 8 illustrates that the proposed method can provides robust trajectory estimation to reconstruct a dense 3D, high-precision map in a large-scale port scenario.

Fig. 9 reports the benchmark results of efficiency for each individual component when performing the large-scale reconstruction with different numbers of LiDARs. As mentioned in Section VI-A.3, the efficiency (defined in (37)) is measured by the ratio between processing time and the time interval ($\Delta t = 10ms$ for CT-MLO, and CTE-MLO; $\Delta t = 50ms$ for MLOAM [36] and MARS [58]). In the sequence *Port 02* (Fig. 9(b)), the results of MARS [58] are absent when the number of LiDAR is less than 3. This is attributed to the failure of the estimation due to insufficient constraints. For the sequence *Port 03* (Fig. 9(c)), all the benchmark methods failed when using a single LiDAR observation and worked fine when the number of LiDARs is greater than or equal to two, which demonstrates the effectiveness of MLO. Thanks to the Gaussian process Kalman filter derived in Section III, which enables continuous-time trajectory estimation in only a few linear iterations, CTE-

MLO and CT-MLO outperform both MLOAM [36] and MARS [58] in total efficiency. For estimation and mapping, the real-time performance of CT-MLO, MLOAM [36], and MARS [58] decreases as the number of LiDARs increases due to the large number of point clouds generated by multiple LiDARs worsens the computational complexity. The localizability-aware point cloud sampling strategy adopted by CTE-MLO reduces the computational complexity of estimation and mapping with the consideration of localizability, which makes the estimation and mapping efficiency of CTE-MLO nearly irrelevant to the number of LiDARs. As shown in Fig. 2(c), a large number of redundant points are eliminated by the localizability-aware point cloud sampling strategy. With the integration of localizability-aware point cloud sampling and continuous-time filter-based registration, CTE-MLO achieves a demonstratively competitive efficiency when compared with SOTA methods. The average efficiency of CTE-MLO for processing with 5-6 LiDAR scans in sequences *Port 01*, *Port 02*, and *Port 03* are $0.208 \ll 1.0$, $0.305 \ll 1.0$, and $0.243 \ll 1.0$, demonstrating that the proposed localization and mapping system achieves almost 3-5 times real-time performance without dropping any scan.

2) Autonomous Sweeping in Campus Scenario

As shown in Fig. 7(b), the sweeper is equipped with 3 mechanical LiDARs (1 Robosense Helios 16 and 2 Robosense

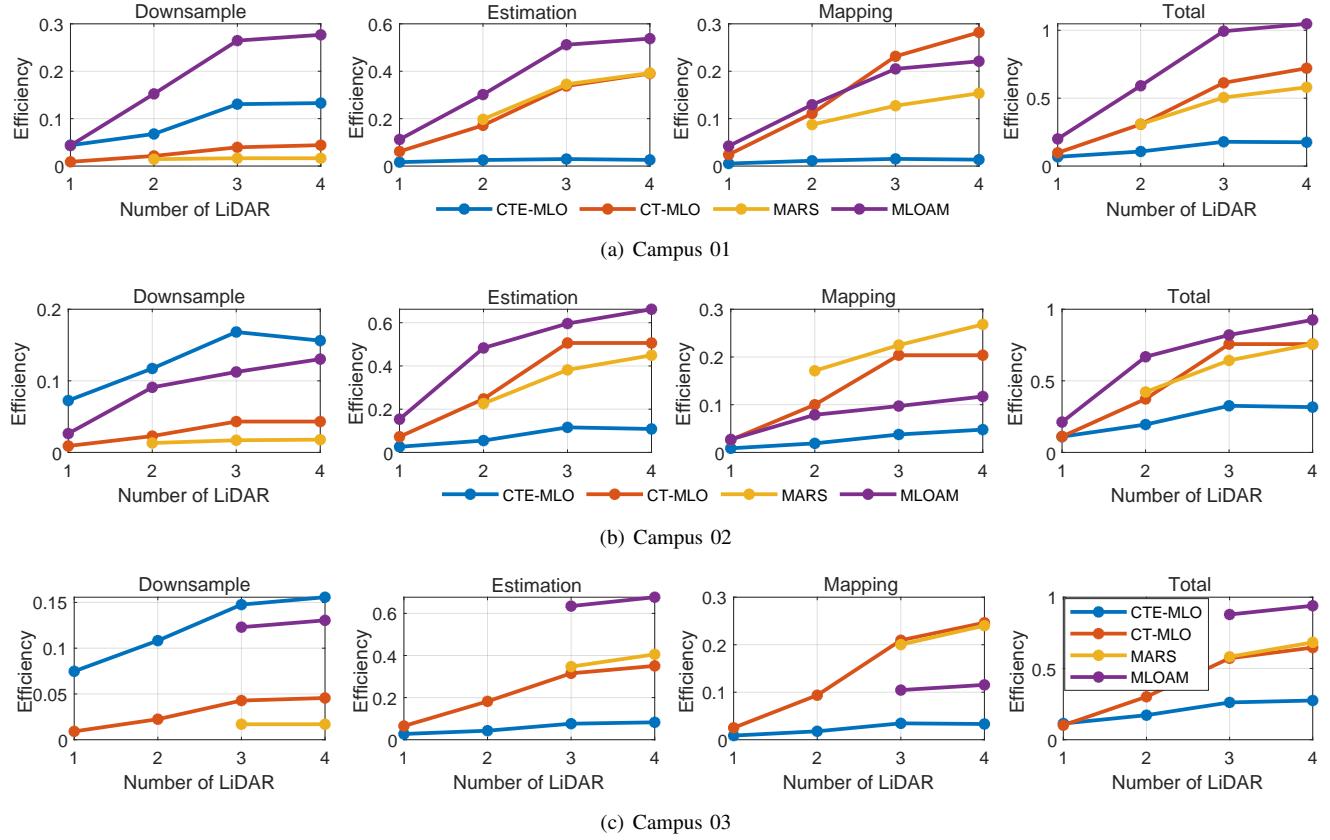


FIGURE 10. Benchmark results of efficiency for each individual component with different numbers of LiDARs when performing autonomous sweeping.

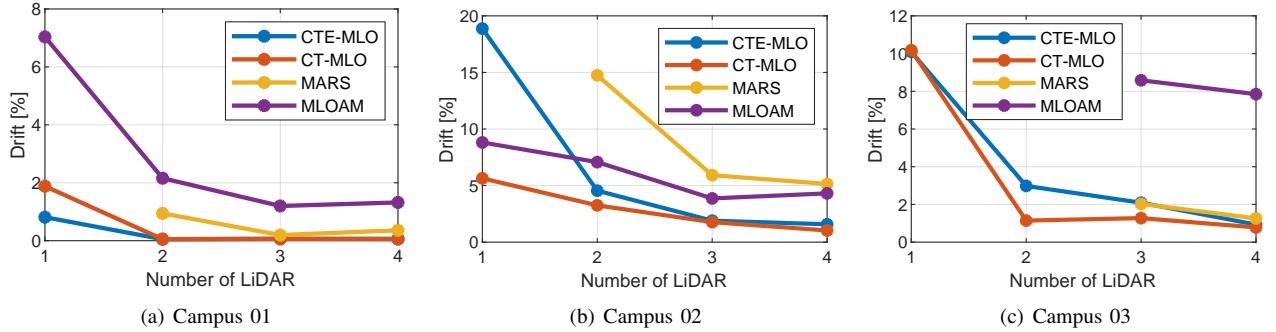


FIGURE 11. Benchmark results of accuracy with different numbers of LiDARs when performing autonomous sweeping.

Helios 32) and 1 Robosense Bpearl solid-state LiDAR. The relationship between the performance (real-time performance and accuracy) achieved by the proposed method and SOTA methods with respect to the number of LiDAR is demonstrated in Fig. 10 and Fig. 11, respectively. For a fair comparison, the real-time performance is measured by the efficiency defined in (37), and the accuracy is measured by drift (ratio between relative translation error and total distance). Due to the numerous buildings on campus, GNSS suffers from the multi-path effect and is inaccurate most of the time, inspired by [82], [83], we collect a prior map of the initial area (container loading area) of the sweeper. The relative translation error is calculated by registering the LiDAR scan

with the prior map when the sweeper returns to the initial area. For the sequence *Campus 01*, the drifts of both CT-MLO and CTE-MLO are well-bounded by 0.06% when the number of LiDARs is greater than or equal to two. Thanks to the localizability-aware point cloud sampling strategy, only points that have a high contribution to localizability are adopted in CTE-MLO, which enables CTE-MLO to achieve adjacent accuracy to CT-MLO with almost no sacrifice in real-time performance when the number of LiDAR increases. For sequences *Campus 02* and *Campus 03*, it can be seen that as the number of LiDARs increases, CTE-MLO achieves a significant improvement in accuracy with very little sacrifice in real-time performance. For sequence *Campus 03*, the

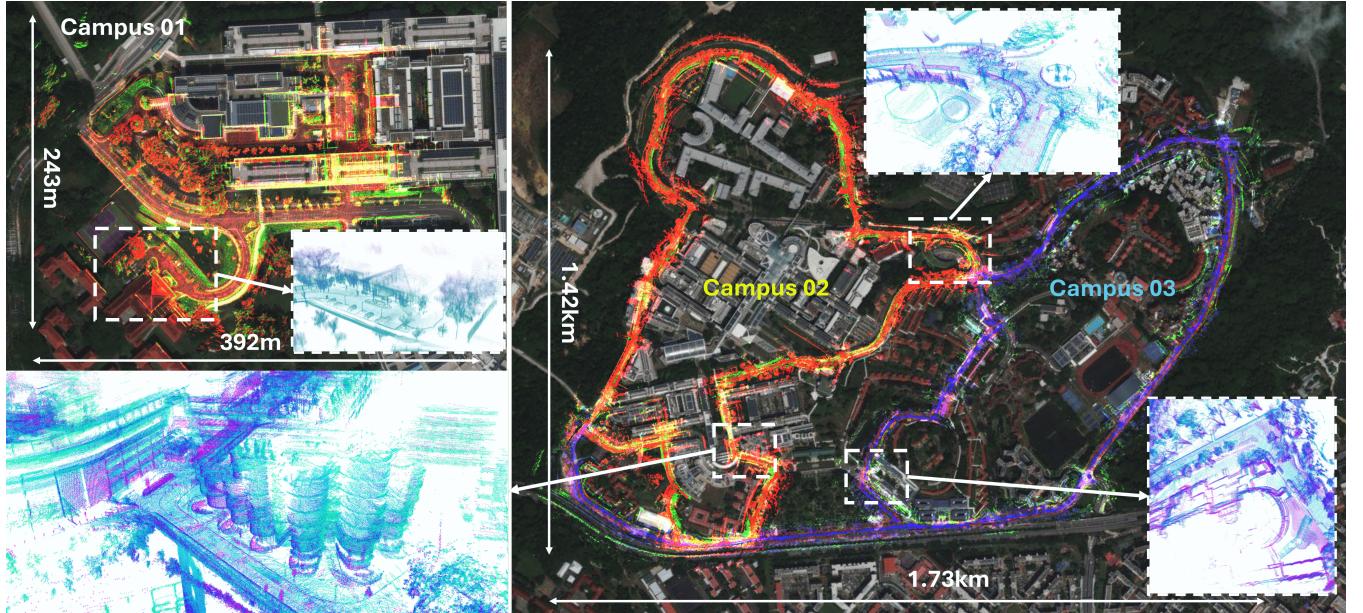


FIGURE 12. The reconstruction result of the CTE-MLO during autonomous sweeping.

results of MLOAM [36] and MARS [58] are absent when the number of LiDAR is less than 3. This is attributed to a large number of LiDAR points being blocked by the sweeper, which leads to estimation degeneration. CTE-MLO realizes stable localization and mapping under different numbers (1-4) and configurations (mechanical and solid-state) of LiDAR. The relative drift of CTE-MLO achieves 0.06% and 1.5% when using 4 LiDARs in mid-scale (*Campus 01*) and large-scale scenarios (*Campus 02*, *Campus 03*), respectively. For the real-time performance, the average efficiency of CTE-MLO for processing with 4 LiDAR in sequence *Campus 01*, *Campus 02*, and *Campus 03* are $0.175 \ll 1.0$, $0.316 \ll 1.0$, and $0.275 \ll 1.0$, which is 2-3 times faster than MARS [58], and 3-5 times faster than MLOAM [36]. As shown in Fig. 12, the point cloud reconstructed by CTE-MLO is well-aligned with the satellite image, and the details of buildings and roads are clear.

3) Autonomous MAV for Field Forest Exploration

To assess the practicality of the proposed method, we embedded the CTE-MLO into a MAV, enabling an autonomous exploration in a forest. As shown in Fig. 7(c), the MAV platform is equipped with two solid-state LiDAR (Livox Mid 360 LiDAR and Intel Realsense L515) and a low-power onboard computer (Nvidia Orin NX). During the autonomous exploration, CTE-MLO is employed to provide real-time trajectory estimation using data from two solid-state LiDARs and coordinate with the planning and control module. The planning and control module is in charge of generating and tracking efficient, collision-free trajectories using the model predictive path integral [84] and adaptive model predictive control (details can be found in our previous works DPPM

[85] and FxTDO-MPC [86]). As shown in Fig. 13, the MAV completed high-precision scanning in a dense forest environment. The area of the exploration regions reaches $2563.66m^2$. During the exploration, the MAV platform can block the point cloud observed by Livox Mid 360, leading to a lack of ground observations in the LiDAR odometry. To address this issue, we install the RealSense L515 downward to provide point cloud measurements on the ground. As depicted in Fig. 14, the proposed CTE-MLO can fully utilize observations from Livox Mid 360 and Intel Realsense L515 point clouds, enabling a more stable estimation of altitude and z-axis velocity compared to single LiDAR odometry.

VII. Conclusion and Future Work

In this paper, a continuous-time and efficient MLO, called CTE-MLO, is developed to achieve accurate localization and mapping at high frequencies by utilizing observations from multiple LiDARs. With the detail derivation, we combine the Gaussian process estimation with the Kalman filter which achieves continuous-time trajectory estimation instead of conventional discrete-time state estimation. The analytic Jacobians for the continuous-time LiDAR update are also derived. Empowered by the continuous-time trajectory estimation, CTE-MLO enables each LiDAR point in a point stream to query the corresponding continuous-time trajectory using its time stamp and handles the streaming of continuous-time LiDAR points using non-rigid registration without undistorting points to a certain instant (as noted in Remark 1). Benefiting from the nature of continuous-time LiDAR measurements, a decentralized multi-LiDAR synchronization scheme is designed to merge and synchronize point clouds from different LiDARs using a LiDAR splitting technique. The proposed synchronization scheme is totally

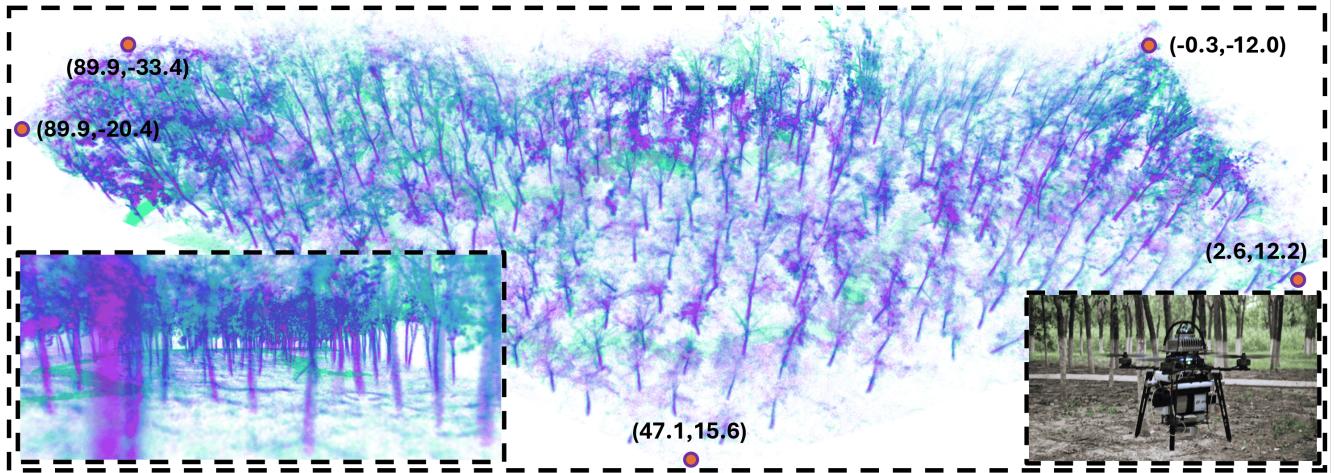


FIGURE 13. Point cloud map reconstructed through the autonomous MAV exploration.

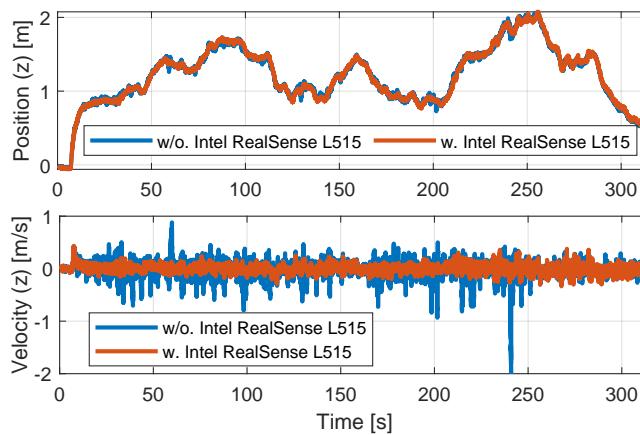


FIGURE 14. Comparison results of position and velocity estimation (in z-axis) with or without Intel RealSense L515 LiDAR observation.

decentralized without the prior definition of the primary LiDAR and can update the synchronized LiDAR scan with a fixed frequency. Moreover, considering the trade-off between real-time performance and accuracy in the MLO system, a point cloud sampling technique is developed, which significantly improves the real-time performance of the CTE-MLO with almost no sacrifice of accuracy. The extensive real-world validations of CTE-MLO, across industrial to field scenarios, demonstrate its effectiveness as a generalized continuous-time MLO. CTE-MLO remains robust under extremely challenging conditions, such as LiDAR degeneration or arbitrary LiDAR data loss, while providing accurate real-time localization and mapping for mobile robots. The combination of Gaussian process regression and the Kalman filter for continuous-time trajectory estimation enhances the extendability of CTE-MLO, enabling seamless deployment across diverse platforms with heterogeneous LiDAR configurations. Furthermore, the localizability-aware point cloud sampling ensures nearly LiDAR-number-irrelevant real-time performance for estimation and mapping, enabling CTE-

MLO to support autonomy on diverse platforms, from large-size vehicles (e.g., full-size trucks and sweepers) to power-limited agility platforms such as MAVs demonstrated in Section VI-B.3. With the successful integration of a series of theory and practical implementations (e.g. decentralized synchronization, localizability-aware point cloud sampling, continuous-time registration, and voxel map management) into the Kalman filter, the proposed method is demonstratively competitive compared to SOTA MLO methods.

Currently, thanks to the tightly coupled fusion of multi-LiDAR observations, CTE-MLO achieves accurate and robust trajectory estimation in real-world scenarios. However, extreme weather conditions (such as heavy rain, and fog) may impact the robustness of LiDAR odometry. Leveraging the exceptional extendability of continuous-time estimation (as noted in Remark 4), future work could consider integrating CTE-MLO with additional sensors (e.g., radar, thermal cameras, GNSS, wheel odometry) that are not sensitive to the extreme weather. The proposed continuous-time estimation allows observations from different sensors to constrain the continuous-time trajectory, making it easy to incorporate additional sensors into CTE-MLO without requiring strict synchronization. The extensibility of CTE-MLO presents significant potential for developing a robust and full-fledged tightly coupled multi-sensor fusion localization and mapping system capable of adapting to diverse weather conditions.

REFERENCES

- [1] R. L. Johns *et al.*, “A framework for robotic excavation and dry stone construction using on-site materials,” *Science Robotics*, vol. 8, no. 84, p. eabp9758, 2023. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.abp9758>
- [2] P. Egli, L. Terenzi, and M. Hutter, “Reinforcement learning-based bucket filling for autonomous excavation,” *IEEE Transactions on Field Robotics*, vol. 1, pp. 170–191, 2024.
- [3] M. Gulzar, Y. Muhammad, and N. Muhammad, “Navigating roundabouts and unprotected turns in autonomous driving,” *IEEE Transactions on Field Robotics*, vol. 1, pp. 27–46, 2024.
- [4] H. Wu, Y. Li, W. Xu, F. Kong, and F. Zhang, “Moving event detection from lidar point streams,” *Nature Communications*,

- vol. 15, no. 1, p. 345, Jan 2024. [Online]. Available: <https://doi.org/10.1038/s41467-023-44554-8>
- [5] N. Stathoulopoulos, B. Lindqvist, A. Koval, A.-A. Agha-Mohammadi, and G. Nikolakopoulos, "Frame: A modular framework for autonomous map merging: Advancements in the field," *IEEE Transactions on Field Robotics*, vol. 1, pp. 1–26, 2024.
- [6] M. Tranzatto *et al.*, "Cerberus in the darpa subterranean challenge," *Science Robotics*, vol. 7, no. 66, p. eabp9742, 2022. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.abp9742>
- [7] N. Kottege *et al.*, "Heterogeneous robot teams with unified perception and autonomy: How team csiro data61 tied for the top score at the darpa subterranean challenge," *Field Robotics*, vol. 4, p. 313–359, 2024.
- [8] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," in *Robotics: Science and Systems*, 2014. [Online]. Available: <https://api.semanticscholar.org/CorpusID:18612391>
- [9] K. Chen, B. T. Lopez, A.-a. Agha-mohammadi, and A. Mehta, "Direct lidar odometry: Fast localization with dense point clouds," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2000–2007, 2022.
- [10] H. Wang, C. Wang, and L. Xie, "Lightweight 3-d localization and mapping for solid-state lidar," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1801–1807, 2021.
- [11] M. Palieri, B. Morrell, A. Thakur, K. Ebadi, J. Nash, A. Chatterjee, C. Kanellakis, L. Carbone, C. Guaragnella, and A.-a. Agha-mohammadi, "Locus: A multi-sensor lidar-centric solution for high-precision odometry and 3d mapping in real-time," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 421–428, 2021.
- [12] A. Reinke *et al.*, "Locus 2.0: Robust and computationally efficient lidar odometry for real-time 3d mapping," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9043–9050, 2022.
- [13] K. Chen, R. Nemiroff, and B. T. Lopez, "Direct lidar-inertial odometry: Lightweight lio with continuous-time motion correction," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 3983–3989.
- [14] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.
- [15] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 5135–5142.
- [16] H. Li, B. Tian, H. Shen, and J. Lu, "An intensity-augmented lidar-inertial slam for solid-state lidars in degenerated environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–10, 2022.
- [17] D. He, W. Xu, N. Chen, F. Kong, C. Yuan, and F. Zhang, "Point-lio: Robust high-bandwidth light detection and ranging inertial odometry," *Advanced Intelligent Systems*, vol. 5, no. 7, p. 2200459, 2023.
- [18] X. Zheng and J. Zhu, "Traj-lio: In defense of lidar-only odometry using an effective continuous-time trajectory," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1961–1968, 2024.
- [19] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 809–816.
- [20] T. Tuna, J. Nubert, Y. Nava, S. Khattak, and M. Hutter, "X-icp: Localizability-aware lidar registration for robust localization in extreme environments," *IEEE Transactions on Robotics*, vol. 40, pp. 452–471, 2024.
- [21] Z. Wu, W. Wang, J. Zhang, Y. Wang, G. Peng, and D. Wang, "Mtgt: Magnetic-lead global localization and tracking in degenerated repetitive environments," *IEEE/ASME Transactions on Mechatronics*, vol. 29, no. 6, pp. 4687–4698, 2024.
- [22] Z. Wu, W. Wang, C. Zhao, Y. Yue, J. Zhang, H. Shen, and D. Wang, "Mm4mm: Map matching framework for multi-session mapping in ambiguous and perceptually-degraded environments," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 4399–4405.
- [23] J. Zhang, M. Kaess, and S. Singh, "A real-time method for depth enhanced visual odometry," *Autonomous Robots*, vol. 41, no. 1, pp. 31–43, Jan 2017. [Online]. Available: <https://doi.org/10.1007/s10514-015-9525-1>
- [24] J. Zhang and S. Singh, "Laser-visual-inertial odometry and mapping with high robustness and low drift," *Journal of field robotics*, vol. 35, no. 8, pp. 1242–1264, 2018.
- [25] T. Shan, B. Englot, C. Ratti, and D. Rus, "Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5692–5698.
- [26] J. Lin and F. Zhang, "R3live: A robust, real-time, rgb-colored, lidar-inertial-visual tightly-coupled state estimation and mapping package," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 10672–10678.
- [27] H. Lu, H. Shen, B. Tian, X. Zhang, Z. Yang, and Q. Zong, "Flight in gps-denied environment: Autonomous navigation system for micro-aerial vehicle," *Aerospace Science and Technology*, vol. 124, p. 107521, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S127096382200195X>
- [28] X. Liu, W. Wen, and L.-T. Hsu, "Glio: Tightly-coupled gnss/lidar/imu integration for continuous and drift-free state estimation of intelligent vehicles in urban areas," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 1412–1422, 2024.
- [29] W. Wen and L.-T. Hsu, "3d lidar aided gnss nlos mitigation in urban canyons," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18224–18236, 2022.
- [30] W. Lee, Y. Yang, and G. Huang, "Efficient multi-sensor aided inertial navigation with online calibration," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5706–5712.
- [31] Z. Yuan, F. Lang, T. Xu, and X. Yang, "Liwo: Lidar-inertial-wheel odometry," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 1481–1488.
- [32] T.-M. Nguyen, M. Cao, S. Yuan, Y. Lyu, T. H. Nguyen, and L. Xie, "Viral-fusion: A visual-inertial-ranging-lidar sensor fusion approach," *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 958–977, 2022.
- [33] Y. Hu, X. Li, D. Kong, P. Ni, W. Hu, and X. Song, "An enhanced-lidar/uwb/ins integrated positioning methodology for unmanned ground vehicle in sparse environments," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 7, pp. 9404–9415, 2024.
- [34] T.-M. Nguyen, M. Cao, S. Yuan, Y. Lyu, T. H. Nguyen, and L. Xie, "Liro: Tightly coupled lidar-inertia-ranging odometry," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 14484–14490.
- [35] W. Wen and L.-T. Hsu, "Towards robust gnss positioning and real-time kinematic using factor graph optimization," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5884–5890.
- [36] J. Jiao, H. Ye, Y. Zhu, and M. Liu, "Robust odometry and mapping for multi-lidar systems with online extrinsic calibration," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 351–371, 2022.
- [37] J. Lin, X. Liu, and F. Zhang, "A decentralized framework for simultaneous calibration, localization and mapping with multiple lidars," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4870–4877.
- [38] S. Das, N. Mahabadi, M. Fallon, and S. Chatterjee, "M-lio: Multi-lidar, multi-imu odometry with sensor dropout tolerance," in *2023 IEEE Intelligent Vehicles Symposium (IV)*, 2023, pp. 1–7.
- [39] J. Lee, M. Bjelonic, A. Reske, L. Wellhausen, T. Miki, and M. Hutter, "Learning robust autonomous navigation and locomotion for wheeled-legged robots," *Science Robotics*, vol. 9, no. 89, p. eadi9641, 2024. [Online]. Available: <https://www.science.org.remotexs.ntu.edu.sg/doi/abs/10.1126/scirobotics.ad9641>
- [40] S. Zhao, H. Zhang, P. Wang, L. Nogueira, and S. Scherer, "Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 8729–8736.
- [41] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, 1991.
- [42] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," in *Proceedings of Robotics: Science and Systems*, Seattle, USA, June 2009.
- [43] P. Biber and W. Strasser, "The normal distributions transform: a new approach to laser scan matching," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, vol. 3, 2003, pp. 2743–2748 vol.3.
- [44] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4758–4765.
- [45] Y. Pan, P. Xiao, Y. He, Z. Shao, and Z. Li, "Mulls: Versatile lidar slam via multi-metric linear least square," in *2021 IEEE International*

- Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 633–11 640.
- [46] E. Jelavic, J. Nubert, and M. Hutter, “Open3d slam: Point cloud based mapping and localization for education,” in *Robotic Perception and Mapping: Emerging Techniques, ICRA 2022 Workshop*. ETH Zurich, Robotic Systems Lab, 2022, p. 24.
- [47] Q.-Y. Zhou, J. Park, and V. Koltun, “Open3D: A modern library for 3D data processing,” *arXiv:1801.09847*, 2018.
- [48] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, “Kiss-icp: In defense of point-to-point icp – simple, accurate, and robust registration if done the right way,” *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 1029–1036, 2023.
- [49] K. Koide, J. Miura, and E. Menegatti, “A portable three-dimensional lidar-based system for long-term and wide-area people behavior measurement,” *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, p. 1729881419841532, 2019. [Online]. Available: <https://doi.org/remotexs.ntu.edu.sg/10.1177/1729881419841532>
- [50] S. Chen, H. Ma, C. Jiang, B. Zhou, W. Xue, Z. Xiao, and Q. Li, “Ndtiloam: A real-time lidar odometry and mapping with weighted ndt and lfa,” *IEEE Sensors Journal*, vol. 22, no. 4, pp. 3660–3671, 2022.
- [51] Z. Liu and F. Zhang, “Balm: Bundle adjustment for lidar mapping,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3184–3191, 2021.
- [52] T. Qin, P. Li, and S. Shen, “Vins-mono: A robust and versatile monocular visual-inertial state estimator,” *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.
- [53] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, “Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam,” *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.
- [54] K. Sun, K. Mohta, B. Pfommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, “Robust stereo visual inertial odometry for fast autonomous flight,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, 2018.
- [55] T. Deng, G. Shen, T. Qin, J. Wang, W. Zhao, J. Wang, D. Wang, and W. Chen, “Pls slam: Progressive neural scene representation with local to global bundle adjustment,” in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 19 657–19 666.
- [56] Z. Liu, X. Liu, and F. Zhang, “Efficient and consistent bundle adjustment on lidar point clouds,” *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4366–4386, 2023.
- [57] X. Liu, Z. Liu, F. Kong, and F. Zhang, “Large-scale lidar consistent mapping using hierarchical lidar bundle adjustment,” *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1523–1530, 2023.
- [58] J. Quenzel and S. Behnke, “Real-time multi-adaptive-resolution-surfel 6d lidar odometry using continuous-time trajectory optimization,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 5499–5506.
- [59] J. Lv, K. Hu, J. Xu, Y. Liu, X. Ma, and X. Zuo, “Clins: Continuous-time trajectory estimation for lidar-inertial system,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 6657–6663.
- [60] T.-M. Nguyen, X. Xu, T. Jin, Y. Yang, J. Li, S. Yuan, and L. Xie, “Eigen is all you need: Efficient lidar-inertial continuous-time odometry with internal association,” *IEEE Robotics and Automation Letters*, vol. 9, no. 6, pp. 5330–5337, 2024.
- [61] J. Lv, X. Lang, J. Xu, M. Wang, Y. Liu, and X. Zuo, “Continuous-time fixed-lag smoothing for lidar-inertial-camera slam,” *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 4, pp. 2259–2270, 2023.
- [62] C. Sommer, V. Usenko, D. Schubert, N. Demmel, and D. Cremers, “Efficient derivative computation for cumulative b-splines on lie groups,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 145–11 153.
- [63] P. Dellenbach, J.-E. Deschaud, B. Jacquet, and F. Goulette, “Cticp: Real-time elastic lidar odometry with loop closure,” in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 5580–5586.
- [64] C. H. Tong, P. Furgale, and T. D. Barfoot, “Gaussian process gauss-newton for non-parametric simultaneous localization and mapping,” *The International Journal of Robotics Research*, vol. 32, no. 5, pp. 507–525, 2013. [Online]. Available: <https://doi.org/10.1177/0278364913478672>
- [65] S. Anderson and T. D. Barfoot, “Full steam ahead: Exactly sparse gaussian process regression for batch continuous-time trajectory estimation on $se(3)$,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 157–164.
- [66] Y. Wu, D. J. Yoon, K. Burnett, S. Kammel, Y. Chen, H. Hvavle, and T. D. Barfoot, “Picking up speed: Continuous-time lidar-only odometry using doppler velocity measurements,” *IEEE Robotics and Automation Letters*, vol. 8, no. 1, pp. 264–271, 2023.
- [67] T.-M. Nguyen, S. Yuan, M. Cao, L. Yang, T. H. Nguyen, and L. Xie, “Miliom: Tightly coupled multi-input lidar-inertia odometry and mapping,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5573–5580, 2021.
- [68] T.-M. Nguyen, D. Duberg, P. Jensfelt, S. Yuan, and L. Xie, “Slicht: Multi-input multi-scale surfel-based lidar-inertial continuous-time odometry and mapping,” *IEEE Robotics and Automation Letters*, vol. 8, no. 4, pp. 2102–2109, 2023.
- [69] Y. Cai, W. Xu, and F. Zhang, “Ikd-tree: An incremental kd tree for robotic applications,” *arXiv preprint arXiv:2102.10808*, 2021.
- [70] C. Bai, T. Xiao, Y. Chen, H. Wang, F. Zhang, and X. Gao, “Fasterlio: Lightweight tightly coupled lidar-inertial odometry using parallel sparse incremental voxels,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4861–4868, 2022.
- [71] C. Yuan, W. Xu, X. Liu, X. Hong, and F. Zhang, “Efficient and probabilistic adaptive voxel mapping for accurate online lidar odometry,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8518–8525, 2022.
- [72] S. Anderson, T. D. Barfoot, C. H. Tong, and S. Särkkä, “Batch nonlinear continuous-time trajectory estimation as exactly sparse gaussian process regression,” *Autonomous Robots*, vol. 39, no. 3, pp. 221–238, Oct 2015. [Online]. Available: <https://doi.org/10.1007/s10514-015-9455-y>
- [73] P. S. Maybeck, *Stochastic models, estimation, and control*. Academic press, 1982.
- [74] J. Sherman and W. J. Morrison, “Adjustment of an inverse matrix corresponding to a change in one element of a given matrix,” *Annals of Mathematical Statistics*, vol. 21, pp. 124–127, 1950. [Online]. Available: <https://api.semanticscholar.org/CorpusID:123460064>
- [75] M. Woodbury, *Inverting Modified Matrices*, ser. Memorandum Report / Statistical Research Group, Princeton. Department of Statistics, Princeton University, 1950. [Online]. Available: https://books.google.com.sg/books?id=_zAnzgEACAAJ
- [76] H. Shen, Q. Zong, B. Tian, X. Zhang, and H. Lu, “Pgo-liom: Tightly coupled lidar-inertial odometry and mapping via parallel and gradient-free optimization,” *IEEE Transactions on Industrial Electronics*, vol. 70, no. 11, pp. 11 453–11 463, 2023.
- [77] H. Shen, Q. Zong, B. Tian, and H. Lu, “Voxel-based localization and mapping for multirobot system in gps-denied environments,” *IEEE Transactions on Industrial Electronics*, vol. 69, no. 10, pp. 10 333–10 342, 2022.
- [78] T. D. Barfoot and P. T. Furgale, “Associating uncertainty with three-dimensional poses for use in estimation problems,” *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 679–693, 2014.
- [79] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng *et al.*, “Ros: an open-source robot operating system,” in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.
- [80] J. Jiao, Y. Zhu, H. Ye, H. Huang, P. Yun, L. Jiang, L. Wang, and M. Liu, “Greedy-based feature selection for efficient lidar slam,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5222–5228.
- [81] T.-M. Nguyen, S. Yuan, M. Cao, Y. Lyu, T. H. Nguyen, and L. Xie, “Ntu viral: A visual-inertial-ranging-lidar dataset, from an aerial vehicle viewpoint,” *The International Journal of Robotics Research*, vol. 41, no. 3, pp. 270–280, 2022. [Online]. Available: <https://doi.org/10.1177/02783649211052312>
- [82] Q. Li, X. Yu, Q. Jorge Peña, and W. Tomi, “Multi-modal lidar dataset for benchmarking general-purpose localization and mapping algorithms,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 3837–3844.
- [83] H. Sier, Q. Li, X. Yu, J. Peña Queralta, Z. Zou, and T. Westerlund, “A benchmark for multi-modal lidar slam with ground truth in gnss-denied environments,” *Remote Sensing*, vol. 15, no. 13, p. 3314, 2023.
- [84] G. Williams, A. Aldrich, and E. A. Theodorou, “Model predictive path integral control: From theory to parallel computation,” *Journal*

- of Guidance, Control, and Dynamics*, vol. 40, no. 2, pp. 344–357, 2017.
- [85] Y. Hui, X. Zhang, H. Shen, H. Lu, and B. Tian, “Dppm: Decentralized exploration planning for multi-uav systems using lightweight information structure,” *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 613–625, 2024.
- [86] L. Xu, B. Tian, C. Wang, J. Lu, D. Wang, Z. Li, and Q. Zong, “Fixed-time disturbance observer-based mpc robust trajectory tracking control of quadrotor,” *IEEE/ASME Transactions on Mechatronics*, pp. 1–11, 2024.



Hongming Shen is currently a Research Fellow with the Centre for Advanced Robotics Technology Innovation (CARTIN), Nanyang Technological University, Singapore. Before joining NTU, he received his Ph.D. degree in Control Theory and Control Engineering from Tianjin University, Tianjin, China, in 2023. He has published several papers in prestigious journals/conferences, including IEEE T-IE, IEEE/ASME T-MECH, IEEE T-IM, IEEE T-IV, RA-L, CVPR, ICRA, and IROS. He has also served as a reviewer in several well-

acknowledged journals, such as IEEE T-FR, IEEE T-IE, Satellite Navigation, and RA-L. He received IEEE ICARCV Best Paper Award in 2024, and 1st Prize in the Aerial Swarm Challenge in 2022. His current research interests include state estimation, multi-sensor fusion, localization and mapping, and aerial Robotics.



Zhenyu Wu received his B.Eng. degree of Electrical Engineering and Automation from Wuhan University, China, in 2016, the M.Sc. and Ph.D. degrees from Nanyang Technological University (NTU), Singapore, in 2017 and 2022, respectively.

He is currently a Research Assistant Professor with the Centre for Advanced Robotics Technology Innovation (CARTIN), NTU. He has published several papers in prestigious journals/conferences, including IEEE/ASME T-MECH, IEEE T-IM, ICRA, IROS, CVPR, and ITSC. In addition, he

has also served as a reviewer in a number of well-acknowledged journals and conferences, such as IEEE T-IE, T-GRS, T-IM, RA-L, ICRA, and IROS. He received 2024 IEEE ICARCV Best Paper Award. His research interests include intelligent perception, localization, and navigation for autonomous systems in complex environments.



Yulin Hui received a B.S. degree in Automation from the School of Electrical and Information Engineering, Tianjin University, Tianjin, China, in 2021. He is currently working toward a Ph.D. degree in control theory and control engineering at Tianjin University, China. He received 1st Prize in the Aerial Swarm Challenge in 2022. His main research interests include autonomous aerial robotics, multi-UAV cooperative exploration in complex environments, and fast trajectory planning in clustered environments.



transportation systems.

Wei Wang received his M.Sc. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, in 2022, and his B.Eng. degree in Automation from Zhejiang University, Hangzhou, China, in 2020. He is currently pursuing a doctoral degree at the School of Electrical and Electronic Engineering, NTU, Singapore. His research interests include multi-sensor fusion for autonomous vehicle localization and navigation in complex terrains and GPS-denied environments, as well as intelligent



modal mapping and localization for unmanned vehicles.



Tianchen Deng received the B.Eng. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2021. He is currently pursuing the Joint Ph.D. degree in control science and engineering with Shanghai Jiao Tong University, Shanghai, China and Nanyang Technological University, Singapore. His main research interests include visual simultaneous localization and mapping (VSLAM), vision-based localization, 3D Reconstruction and navigation of autonomous robot.



Yeqing Zhu is currently a Postdoctoral Research Fellow with the School of Aerospace Engineering, Beijing Institute of Technology. She is also a visiting scholar with the Centre for Advanced Robotics Technology Innovation (CARTIN), Nanyang Technological University (NTU), Singapore. She received a Ph.D. degree from Beijing Institute of Technology, Beijing, China, in 2022. Her main research interests include computer vision, simultaneous localization and mapping for autonomous robots, flight dynamics, and control.



and control for unmanned systems.



Danwei Wang (Life Fellow, IEEE) received his Ph.D. and M.S.E. degrees from the University of Michigan, Ann Arbor, USA, in 1989 and 1984, respectively. He received his B.E. degree from the South China University of Technology, China, in 1982.

Since 1989, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Currently, he is Full Professor and the Programme Director of the Centre for Advanced Robotics Technology Innovation (CARTIN). He is the Chair of IEEE Singapore Robotics and Automation Chapter and a senator in NTU Academics Council. He has served as general chairman, technical chairman and various positions in international robotics and control conferences, such as ICRA, IROS, and ICARCV conferences. He is an associate editor for the International Journal of Humanoid Robotics and served as an associate editor of Conference Editorial Board, IEEE Control Systems Society from 1998 to 2005. He has published 5 books and over 500 technical articles in the areas of robotics and control theories, as well as fault diagnosis and prognosis of complex engineering systems. He was a recipient of Alexander von Humboldt Fellowship, Germany, and ST Engineering Distinguished Professor Award, Singapore. He is a Fellow of the Academy of Engineering, Singapore (SAEng) and Life Fellow of the IEEE. His research interests include robotics, control theory and applications.