This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

PALoc: Advancing SLAM Benchmarking With Prior-Assisted 6-DoF Trajectory Generation and Uncertainty Estimation

Xiangcheng Hu, Linwei Zheng, Jin Wu, Ruoyu Geng, Yang Yu, Hexiang Wei, Xiaoyu Tang, *Member, IEEE*, Lujia Wang, *Member, IEEE*, Jianhao Jiao, *Member, IEEE*, and Ming Liu, *Senior Member, IEEE*

***Abstract*—Accurately generating ground truth (GT) trajectories is essential for simultaneous localization and mapping (SLAM) evaluation, particularly under varying environmental conditions. This study presents PALoc, a systematic approach that leverages a prior map-assisted framework for the first-time generation of dense six-degree-of-freedom GT poses, significantly enhancing the fidelity of SLAM benchmarks across both indoor and outdoor environments. Our method excels in handling degenerate and stationary conditions frequently encountered in SLAM datasets, thereby increasing robustness and precision. A critical feature of PALoc is the detailed derivation of covariance within the factor graph, enabling an in-depth analysis of pose uncertainty propagation. This analysis plays a pivotal role in illustrating specific pose uncertainty and in elevating trajectory reliability from both theoretical and practical perspectives. In addition, we provide an open-source toolbox for the criteria of map evaluation, facilitating the indirect assessment of overall trajectory precision. Experimental results show at least a 30% improvement in map accuracy and a 20% increase in direct trajectory accuracy compared to the iterative closest point algorithm across diverse environments, with substantially enhanced robustness Our publicly available solution, PALoc, extensively applied in the FusionPortable dataset, is geared toward SLAM benchmark augmentation and represents a significant advancement in SLAM evaluation.**

***Index Terms*—LiDAR Degeneracy, LiDAR Localization, simultaneous localization and mapping (SLAM) Benchmarking, uncertainty estimation.**

I. Introduction

A. Motivation and Challenges

GROUND truth (GT) trajectory generation is crucial for assessing simultaneous localization and mapping (SLAM). It requires sophisticated geometric computations to fully explore the capabilities of GT estimation methods. Its primary goal is to establish a benchmark for thoroughly assessing the fidelity of SLAM technologies. GT trajectories offer more than just positional data; they provide a robust framework for the critical analysis and enhancement of navigation and mapping techniques. This is particularly vital in the fields of robotics and autonomous systems, where slight errors in SLAM assessments can result in significant real-world discrepancies. The development of GT trajectory generation methods is, therefore, of critical importance, reflecting its significant impact and the keen interest it generates within the community [3], [4].

Despite their centrality in SLAM evaluation, the existing GT trajectory generation methods face notable limitations. Tracking-based approaches, relying on expensive instruments, such as motion capture system (MOCAP), total station (TS), global navigation satellite system (GNSS), and inertial navigation system (INS), are hindered by limited environmental coverage [5] and occlusion susceptibility [6], often resulting in trajectories with unbounded error [3], [4], limited degrees of freedom (DoFs) (e.g., three DoFs) [7], and sparse tracking (e.g., 1 Hz) [8]. Scanning-based methods, while innovative in using laser scanners for prior map construction and subsequent dense six-DoF trajectory generation through point cloud localization, grapple with poor local accuracy [9], localization failures [10] during intense movements, and inability to handle degenerate scenarios [8]. Most critically, these methods lack theoretical analysis and quantitative results, leaving a gap in measuring overall trajectory quality in practical applications.

Manuscript received 7 December 2023; accepted 29 January 2024. Recommended by Technical Editor Y. Gu and Senior Editor M. Indri. This work was supported in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2021B1515120032, in part by the Hong Kong University of Science and Technology (Guangzhou) Joint Funding Program under Grant 2024A03J0618, and in part by the Project of Hetao Shenzhen–Hong Kong Science and Technology Innovation Cooperation Zone under Grant HZQB-KCZYB-2020083. *(Corresponding authors: Jianhao Jiao; Ming Liu.)*

Xiangcheng Hu, Jin Wu, Ruoyu Geng, Yang Yu, and Hexiang Wei are with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong (e-mail: xhubd@connect.ust.hk).

Linwei Zheng and Lujia Wang are with the System Hub, Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511458, China (e-mail: eewanglj@hkust-gz.edu.cn).

Xiaoyu Tang is with the School of Physics and Telecommunication Engineering, South China Normal University, Guangzhou 510631, China (e-mail: tangxy@scnu.edu.cn).

Jianhao Jiao is with the Department of Computer Science, University College London, WC1E 6BT London, U.K. (e-mail: ucacjji@ucl.ac.uk).

Ming Liu is with the Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511458, China (e-mail: eelium@hkust-gz.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMECH.2024.3362902.

Digital Object Identifier 10.1109/TMECH.2024.3362902

To address these challenges, our proposed factor-graph-based methodology considers temporal sensor information to ensure robustness, local accuracy, and consistency across various platforms. We have integrated specialized modules for degeneracy handling and zero-velocity potential update (ZUPT), refining trajectory accuracy and ensuring robustness against environmental dynamics. Our estimation of pose uncertainty for each frame offers theoretical insights and practical implications for trajectory generation tasks. In addition, by integrating a quantitative evaluation module inspired by multiview stereo or other reconstruction techniques [11], [12], [13], we offer a nuanced assessment of the generated trajectories. This comprehensive approach not only resolves the existing challenges in GT trajectory generation methods but also establishes a new benchmark in this domain, demonstrating the practical feasibility and theoretical robustness of our method.

B. Contribution

In this article, we present a comprehensive method for generating high-quality GT trajectories to augment SLAM benchmark datasets. Our key contributions are as follows.

1) We introduce a prior-assisted localization system to generate dense six-DoF trajectories for SLAM benchmarking, incorporating degeneracy-aware map factors and ZUPT factors to boost robustness and precision.
2) We conduct a detailed analysis of uncertainty propagation within our factor graph system, estimating the uncertainty of each pose, which informs the quality and robustness of the final trajectories, offering theoretical and practical insights for trajectory generation tasks.
3) We offer an open-source toolbox[1] for map evaluation criteria, acting as indirect precision indicators of the generated trajectories, emphasizing the practical utility and adaptability of our method in diverse applications.

Leveraging a loosely coupled fusion approach based on the factor graph, our method swiftly adapts to state-of-the-art (SOTA) LiDAR-centric SLAM systems, such as LiDAR Odometry, LiDAR-Inertial Odometry (LIO), and LiDAR-Visual-Inertial Odometry (LVIO), demonstrating versatility, scalability, and compatibility. Applied extensively in the FusionPortable dataset [2], our method successfully produced trajectories for 13 of 16 sequences, achieving at least a 30% improvement in map accuracy and a 20% increase in direct trajectory accuracy compared to the iterative closest point (ICP) [1] algorithm across diverse campus environments. To our knowledge, this approach represents the first open-source solution[2] designed specifically for crafting six-DoF GT trajectories in benchmarking, marking a significant contribution to SLAM research.

II. Related Work

Generating GT trajectory is a pivotal task in constructing SLAM benchmark. This section briefly reviews GT trajectory generation methods within SLAM datasets.

A category of tracking-based GT trajectory generation methods heavily relies on tracking sensors, offering high accuracy but fraught with several drawbacks. Datasets such as in [3], [14], [15], [16], and [17] primarily employ GNSS/INS, facing challenges like limited satellite visibility and the multipath effect, which significantly reduce accuracy in outdoor urban areas and are not applicable for indoor dataset creation. In contrast, dataset such as in [2], [5], and [18] employ MOCAP to generate six-DoF GT trajectories. However, their dependence on MOCAP limits data collection to its coverage area, limiting the diversity of capturable scenarios. TSs [7], while precise, encounter low pose frequency and are limited to positional tracking, struggling with coverage and occlusions.

Scanning-based methods utilize sensor measurements and prior maps [19], introducing LiDAR measurement noise and map noise but enabling GT trajectory creation in any area of a mapped scene. Innovations such as in [9] employ laser scanners for map construction, typically using brute-force normal distributions transform (NDT) [20] or ICP [1] algorithms for point cloud localization. While these methods have good global accuracy, they often yield sparse GT poses and lack robustness and local accuracy, particularly in LiDAR-degenerate environments. Distributed algorithms [21], [22] also contribute significantly to estimation and target localization. LIO/LVIO methods [23], [24], [25], [26], [27], [28], [29], [30] estimate poses using temporal sensor data, handling scene changes and intense movements with high local accuracy but limited global precision, and are rarely used for GT trajectory generation. In large outdoor scenarios, the fusion of LIO/LVIO with GNSS/INS has been employed for GT generation [2], overcoming conventional GNSS/INS limitations.

Following the scanning-based method, our approach further integrates LIO/LVIO within a unified factor-graph-based fusion framework. This enables our method to achieve both excellent local precision and global consistency, even in dynamic movements. To enhance robustness and accuracy, we implement degeneracy-aware map factors and ZUPT factors. We offer map evaluation benchmarks for succinct trajectory precision indication, combined with theoretical analysis of pose uncertainty propagation within our system. By estimating the uncertainty of each pose, we provide clear insights into pose quality, thus guiding trajectory generation tasks from both theoretical and practical perspectives. This comprehensive method differentiates our approach in the SLAM dataset landscape, marking a significant departure from existing practices.

III. Problem Statement and Formulation

In this section, we formulate the trajectory generation task as a localization problem. Our method employs a factor graph framework, accounting for various factors essential for accurate trajectory estimation in complex environments.

A. Notations and Definitions

For clarity and precision in our problem formulation, we adopt a consistent notation system. The inertial measurement unit (IMU) is rigidly attached to the LiDAR sensor, defining

[1] [Online]. Available: https://github.com/JokerJohn/Cloud_Map_Evaluation
[2] [Online]. Available: https://github.com/JokerJohn/PALoc

the body frame $\boldsymbol{B}$ that serves as the reference frame for the system. The world frame is denoted by $\boldsymbol{W}$, and the LiDAR frame is denoted by $\boldsymbol{L}$. The robot pose at any discrete time instance k is denoted by $\boldsymbol{X}_k$, comprising its position, $\boldsymbol{t}_k \in \mathbb{R}^3$, and orientation, $\boldsymbol{R}_k \in \mathrm{SO}(3)$. Velocity, accelerator biases, and gyroscope biases are denoted by $\boldsymbol{v}_k, \boldsymbol{b}_{a,k}$, and $\boldsymbol{b}_{\omega,k}$, respectively, but are not included in the state variable as our focus is on pose graph. The state vector $\boldsymbol{\mathcal{X}}$ is, thus, defined in terms of the pose: $\boldsymbol{\mathcal{X}} = \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_P\}$, where P represents the total number of time steps. The environmental context is provided by the prior map $\mathcal{M}$, with $\mathcal{M}_k$ referencing the relevant subset at the initial pose $\boldsymbol{x}_k$. Observations at time k incorporate LiDAR $\mathcal{L}_k$ and IMU measurements $\mathcal{I}_k = \{\boldsymbol{a}_k, \boldsymbol{\omega}_k\}$, collectively represented as $\mathcal{Z}_k = \{\mathcal{M}_k, \mathcal{L}_k, \mathcal{I}_k\}$.

B. Maximum a Posteriori Estimation

The cornerstone of our state estimation is the maximum a posteriori (MAP) framework, which aims to identify the most likely state variables $\boldsymbol{\mathcal{X}}$. Given the observations $\mathcal{Z}$, this process is described by

$$\hat{\boldsymbol{\mathcal{X}}} = \arg\max_{\boldsymbol{\mathcal{X}}} [p(\boldsymbol{\mathcal{X}}|\mathcal{Z})] \qquad = \arg\max_{\boldsymbol{\mathcal{X}}} [p(\mathcal{Z}|\boldsymbol{\mathcal{X}})p(\boldsymbol{\mathcal{X}})]$$

where $p(\mathcal{Z})$ is a constant.

Under the conditional independence assumption, the joint likelihood decomposes into a product of individual probabilities as follows:

$$\hat{\boldsymbol{\mathcal{X}}} = \arg\max_{\boldsymbol{\mathcal{X}}} \left[p(\boldsymbol{\mathcal{X}}) \prod_{k=1}^{P} \prod_{i=1}^{M} p(\boldsymbol{z}_i^k | \boldsymbol{X}_k, \boldsymbol{\mathcal{X}}) \right] \tag{1}$$

where P and M denote the number of time steps and factors. Equation (1) is equivalent to minimizing the sum of squared residuals across all factors

$$\hat{\boldsymbol{\mathcal{X}}} = \arg\min_{\boldsymbol{\mathcal{X}}} \left[\sum_{k=1}^{P} \sum_{i=1}^{M} \boldsymbol{r}_i^k(\boldsymbol{\mathcal{X}})^\top \boldsymbol{\Sigma}_i^{-1} \boldsymbol{r}_i^k(\boldsymbol{\mathcal{X}}) \right].$$

The optimization of $\boldsymbol{\mathcal{X}}$ is achieved using nonlinear least squares techniques, such as Gauss–Newton methods, accommodating the inherent uncertainties in sensor data.

C. Factor Graph Formulation

In our system, we represent the factor graph as $\mathcal{G} = (\mathcal{X}, \mathcal{F}, \mathcal{E})$, forming the backbone of our pose estimation framework. Here, $\mathcal{X}$ comprises the set of state variables related to robot poses, $\mathcal{F}$ embeds constraints between these variables, and $\mathcal{E}$ indicates the connections between factors and variables. A graphical illustration of our factor graph, integrating five distinct factor types, is shown in Fig. 3. Detailed explanations of these factors are provided as follows.

1) Lidar-Based Odometry Factor (LO): This factor imposes constraints on consecutive robot poses $\boldsymbol{X}_k$ and $\boldsymbol{X}_{k-1}$, aligning them based on odometry measurements. The intricate details of this factor are elaborated in Section VII-A.

2) Loop Closure Factor (LC): The loop closure factor aims to enforce congruence between two robot poses, $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$, which, despite being temporally distant, should be spatially proximal due to a loop closure event.

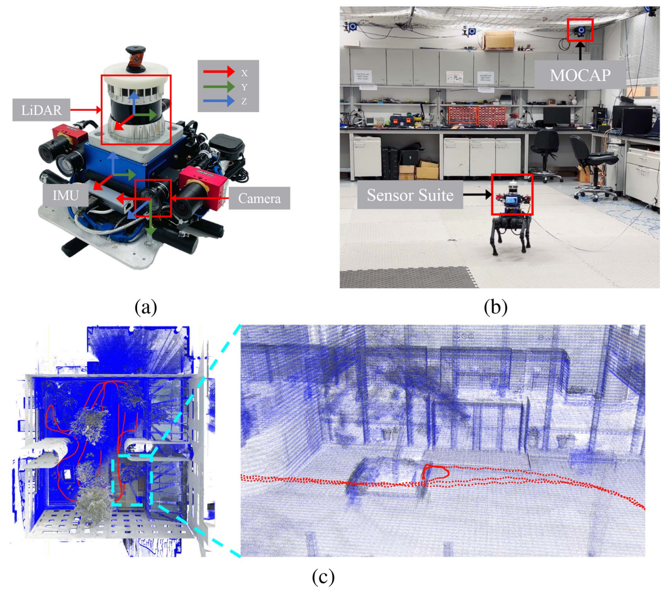


Fig. 1. (a) Sensor configuration with corresponding coordinate frames for the SLAM benchmarking. (b) Quadruped robot equipped with sensor suite in Motion Capture Room. (c) Prior RGB point cloud map with the estimated trajectory (red line) and map (blue point cloud) by PALoc on `garden_day`.

3) No Motion Factor (NM): Designed to cater to stationary periods, the no motion factor [31] ensures that the relative pose remains constant during such intervals. Further insights into this factor are provided in Section V-C.

4) Gravity Factor (GF): This factor critically influences the estimation process under ZUPT conditions by constraining the magnitude of the gravity vector. A detailed formulation of this factor can be found in Section V.

5) Degeneracy-Aware Map Factor (DM): The degeneracy-aware map factor constrains the robot pose with a prior map, thus ensuring robustness in degenerate scenarios. Detailed discussions on this factor are presented in Section VI.

IV. System Overview

We develop our system under a set of simplifying assumptions to streamline the design process.

1) The LiDAR and IMU are hardware time synchronized and well calibrated, ensuring precise data alignment.
2) Our research utilizes pose SLAM [32], where only poses are included in the graph, ensuring efficiency and optimal map construction.

Fig. 2 presents an overview of our system's architecture, with the caption detailing its components and workflow.

V. ZUPT Factors

Inspired by previous gravity and ZUPT work [26], [31], [33], [34], we implement the gravity factor and no motion factor to handle these frequently occurred static scenes.

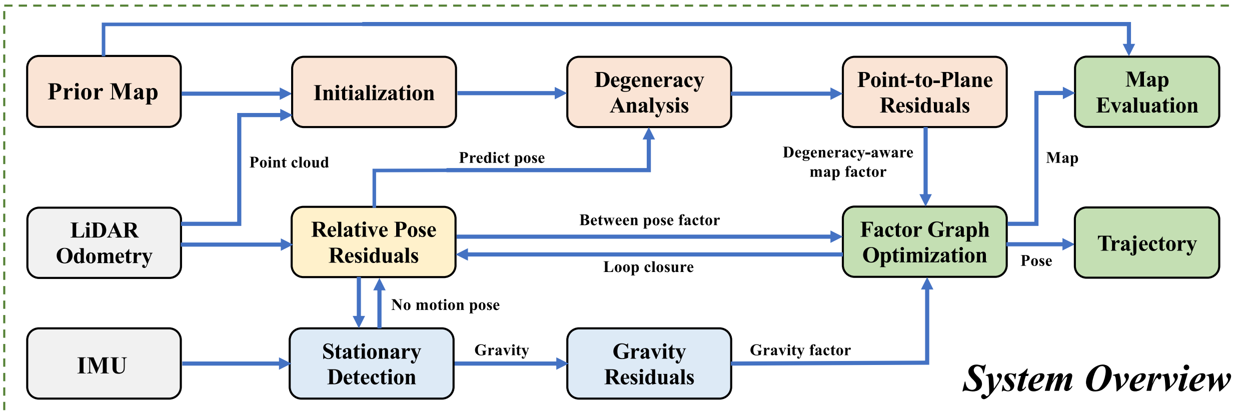


Fig. 2. System pipeline overview. This figure illustrates the architecture of our system, starting with initialization on a prior map and LiDAR odometry. Degeneracy analysis and point-to-plane registration are employed to create a degeneracy-aware map factor. The system also processes odometry and IMU data for stationary detection, forming no motion factors, and integrating gravity factors. Following the optimization of the factor graph on a framewise basis, loop closure detection is carried out, contributing to the loop factor. This sequential pipeline results in the generation of estimated poses and maps, which assists in the indirect evaluation of trajectory accuracy.

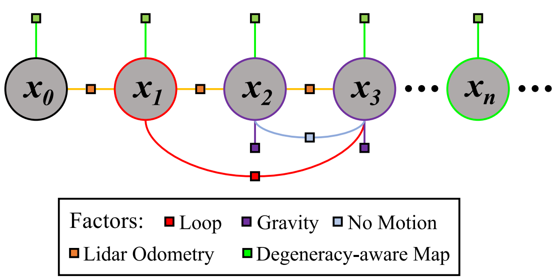


Fig. 3. Factor graph of our proposed system. Gray circles represent different system states at specific times, and colored rectangles symbolize various factors. The purple rectangle signifies the gravity factor, connected to states outlined in purple, illustrating gravity constraints during stationary periods. States with green outlines are indicative of LiDAR degenerate scenarios, and those with red outlines mark instances of successful loop closure detection. An in-depth discussion on uncertainty propagation is detailed in Section VII-B.

A. Stationary Detection

The system needs to pass a two-stage evaluation to be seen as static by analyzing the IMU data and LIO poses. Given a set of recent accelerometer measurements at time i, $\{\boldsymbol{a}_i, \boldsymbol{\omega}_i\}$, we initially compute the mean values and variations of these measurements as $\overline{\boldsymbol{a}} = \frac{1}{N}\Sigma_{i=1}^{N}\boldsymbol{a}_i$ and $\overline{\boldsymbol{\omega}} = \frac{1}{N}\Sigma_{i=1}^{N}\boldsymbol{\omega}_i$. First, we establish thresholds $\boldsymbol{\epsilon}_a$ and $\boldsymbol{\epsilon}_\omega$ for the variations of acceleration and angular velocity. If $\Delta\boldsymbol{a} < \boldsymbol{\epsilon}_a$ and $\Delta\boldsymbol{\omega} < \boldsymbol{\epsilon}_\omega$, the system is tentatively classified as static as follows:

$$\Delta\boldsymbol{a} = \max_i \|\boldsymbol{a}_i - \overline{\boldsymbol{a}}\|, \quad \Delta\boldsymbol{\omega} = \max_i \|\boldsymbol{\omega}_i - \overline{\boldsymbol{\omega}}\|. \tag{2}$$

Next, we examine the LIO pose, characterized by the relative translation $\boldsymbol{t}_{i,i-1}$ and rotation $\boldsymbol{R}_{i,i-1}$. We set thresholds $\boldsymbol{\epsilon}_t$ and $\boldsymbol{\epsilon}_R$ for the variations in translation and rotation, respectively. If both $|\boldsymbol{t}_{i,i-1}| < \boldsymbol{\epsilon}_t$ and $|\angle(\boldsymbol{R}_{i,i-1})| < \boldsymbol{\epsilon}_R$, we further confirm the static state. Our refined two-stage approach offers a consistent method for detecting ZUPT conditions.

B. Gravity Factor

We aim to estimate the pose $\boldsymbol{p} \in SE(3)$, given that the gravity vector $\boldsymbol{g} \in \mathbb{R}^3$ is a normalized constant. The measured acceleration in the body frame, $\boldsymbol{a}_m^b$, when stationary, is an approximation of the negative gravity vector. This acceleration is transformed into the world frame $\boldsymbol{a}^w$ by the rotation matrix $\boldsymbol{R}$ encapsulated in the pose $\boldsymbol{p}$, $\boldsymbol{a}^w = \boldsymbol{R}\boldsymbol{a}_m^b$. The error $\boldsymbol{r}_{\text{gf}}$ is then defined as the deviation of the normalized world frame acceleration from the fixed gravity direction $\boldsymbol{r}_{\text{gf}} = \boldsymbol{a}^w/\|\boldsymbol{a}^w\| - \boldsymbol{g}$. The Jacobian matrix $\boldsymbol{J}_{\text{gf}}$ with respect to the pose is given as

$$\boldsymbol{J}_{\text{gf}} = \begin{bmatrix} \boldsymbol{0}_{3\times3} & \frac{\boldsymbol{R}[\boldsymbol{a}_m^b]_\times}{\|\boldsymbol{a}^w\|} \end{bmatrix}_{3\times6} \tag{3}$$

where $[\boldsymbol{a}_m^b]_\times$ denotes the skew-symmetric matrix of the vector $\boldsymbol{a}_m^b$. The covariance matrix $\boldsymbol{\Sigma}$ is then derived from the inverse of the Fisher information matrix $\boldsymbol{H}$ and is obtained as follows:

$$\boldsymbol{H} = \boldsymbol{J}_{\text{gf}}^\top \boldsymbol{W}_{\text{gf}} \boldsymbol{J}_{\text{gf}}, \quad \boldsymbol{\Sigma}_{\text{gf}} \approx \boldsymbol{H}^{-1} \tag{4}$$

where $\boldsymbol{W}_{\text{gf}}$ is the weight matrix associated with IMU noise.

C. No Motion Factor

The no motion factor can be defined as $\boldsymbol{r}_{\text{nm}} = \text{Log}(\boldsymbol{X}_t^{-1}\boldsymbol{X}_{t-1})$, where $\boldsymbol{X}_t$ and $\boldsymbol{X}_{t-1}$ denote the consecutive poses, and Log translates SE(3) into its Lie algebra. The covariance matrix $\boldsymbol{\Sigma}_{\text{nm}}$ is depicted as a diagonal matrix with small entries, indicating minimal uncertainty in both rotation and translation.

VI. Degeneration-Aware Map Factor

Inspired by [23], [35], [36], [37], [38], and [39], our proposed DM approach augments the traditional point-to-plane ICP algorithm by integrating degeneracy detection and uncertainty estimation modules.

A. Point-to-Plane ICP

The point-to-plane ICP residual is expressed as

$$\boldsymbol{r}_i = (\boldsymbol{R}\boldsymbol{p}_i + \boldsymbol{t} - \boldsymbol{q}_i) \cdot \boldsymbol{n}_i \tag{5}$$

where $\boldsymbol{p}_i$ and $\boldsymbol{q}_i$ represent the matched point pairs from the current LiDAR sweep and the prior map, respectively, and $\boldsymbol{n}_i$ denotes the associated plane normal. The optimization objective is to determine the rotation $\boldsymbol{R}$ and translation $\boldsymbol{t}$ that minimize the sum of squared residuals

$$(\boldsymbol{R}^*, \boldsymbol{t}^*) = \arg\min_{\boldsymbol{R},\boldsymbol{t}} \boldsymbol{\Sigma}_{i=1}^{N} \|\boldsymbol{r}_i\|^2. \tag{6}$$

Utilizing the Hessian matrix $\boldsymbol{H}$, pose updates are computed as

$$\boldsymbol{H} \begin{bmatrix} \Delta\boldsymbol{\theta}^\top & \Delta\boldsymbol{t}^\top \end{bmatrix}^\top = -\boldsymbol{J}_{\text{dm}}^\top \boldsymbol{r}_{\text{dm}} \tag{7}$$

where $\boldsymbol{J}_{\text{dm}}$ denotes the Jacobian of residuals and $\boldsymbol{r}_{\text{dm}}$ is the residual vector. The solution to this system provides incremental updates $\Delta\boldsymbol{\theta}$ and $\Delta\boldsymbol{t}$. We then compute the covariance matrix $\boldsymbol{\Sigma}_{\text{dm}}$ as

$$\boldsymbol{\Sigma}_{\text{dm}} \approx \boldsymbol{H}^{-1}, \quad \boldsymbol{H} = \boldsymbol{J}_{\text{dm}}^\top \boldsymbol{W}_{\text{dm}} \boldsymbol{J}_{\text{dm}}. \tag{8}$$

This approximation assumes Gaussian noise and a local quadratic cost function through the inverse Hessian matrix $\boldsymbol{H}$. Here, $\boldsymbol{W}_{\text{dm}}$ is the weighting matrix related to LiDAR measurement noise.

B. Degeneracy Detection

The ICP covariance derived from the Hessian matrix is often inadequate for comprehensively identifying degenerate conditions, which are particularly prevalent in environments with repetitive or sparse features [37]. The objective in addressing degeneration is to reliably identify degenerate directions and then mitigate their effects through optimization strategies or sensor weight adjustments. Following the methodology proposed by Tagliabue et al. [36], we employ singular value decomposition (SVD) on the Hessian matrix to analyze the condition numbers continually to ascertain the principal directions of degeneracy

$$\boldsymbol{H}_{\boldsymbol{X}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top \tag{9}$$

where $\boldsymbol{H}_{\boldsymbol{X}}$ is the Hessian submatrix for rotation or translation, and SVD is indeed used to discern the singular values $\boldsymbol{\Sigma}$ with orthogonal matrices $\boldsymbol{U}$ and $\boldsymbol{V}$. Due to different scales of translation and rotation part, the separate condition number $\kappa(\boldsymbol{H}_{\boldsymbol{X}})$ is then given as $\kappa(\boldsymbol{H}_{\boldsymbol{X}}) = \sigma_{\max}/\sigma_{\min}$, with $\sigma_{\max}$ and $\sigma_{\min}$ being the largest and smallest singular values, respectively. A high condition number signifies potential degeneration, indicating a need for adaptive constraints.

Next, the directional contributions of each correspondence pair to the pose update are then determined by $\boldsymbol{C} = -\boldsymbol{J}^\top\boldsymbol{r}$, which yields a vector $\boldsymbol{C}$ representing the influence on the six pose parameters. We compute the relative contributions in translation and rotation by calculating the ratio of correspondences that maximally contribute to each pose parameter: $\text{CR}_{\text{dim}} = N_{\text{dim}}/N_{\text{total}}$. The contribution ratio CR_{dim} for a specific dimension is defined by the number of correspondences, N_{dim}, with the maximal contribution in that dimension, relative to the total correspondences, N_{total}. A dimension with a higher CR_{dim} demonstrates a predominant influence, directing the adjustment of constraints to effectively counteract degeneracy. This mechanism ensures precise detection and management of significant degeneracy, thus preserving the system's accuracy. In line with LOAM's [23], [39] efficiency-first approach, degeneracy detection is conducted only in the first iteration, which is an effective strategy in practical scenarios.

VII. Uncertainty Estimation in Factor Graph

In contrast to the majority of graph-based SLAM systems, such as LOAM and its enhancements [40], [41], which rely exclusively on a fixed diagonal noise model, there is a notable limitation in their ability to assess the uncertainty associated with the final pose estimation. This section will introduce a method for estimating the uncertainty of the odom factor and then analyze how each factor in the entire factor graph affects the pose uncertainty propagation.

A. Odom Factor

Given two poses $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ with their respective covariance matrices $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ from the front-end odometry, and their cross covariance $\boldsymbol{\Sigma}_{12}$, the joint covariance matrix is

$$\boldsymbol{\Sigma}_{\text{joint}} = \begin{bmatrix} \boldsymbol{\Sigma}_1 & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{12}^T & \boldsymbol{\Sigma}_2 \end{bmatrix}. \tag{10}$$

Our goal is to estimate the covariance of the relative pose $\boldsymbol{X}_{12} = \boldsymbol{X}_1^{-1}\boldsymbol{X}_2$, which can be computed by transforming the joint distribution of $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$. The Schur complement [42] of $\boldsymbol{\Sigma}_2$ in the joint covariance matrix is calculated as

$$\boldsymbol{\Sigma}_{\text{schur}} = \boldsymbol{\Sigma}_1 - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_{12}^T. \tag{11}$$

Due to the computational challenges associated with the inversion of $\boldsymbol{\Sigma}_2$ in practical applications, we adopt a Jacobian-based approach to approximate the covariance. The full covariance propagation is computed as

$$\boldsymbol{\Sigma}_{\text{full}} \approx \boldsymbol{J}_1\boldsymbol{\Sigma}_1\boldsymbol{J}_1^T + \boldsymbol{J}_1\boldsymbol{\Sigma}_{12}\boldsymbol{J}_2^T + \boldsymbol{J}_2\boldsymbol{\Sigma}_{12}^T\boldsymbol{J}_1^T + \boldsymbol{J}_2\boldsymbol{\Sigma}_2\boldsymbol{J}_2^T. \tag{12}$$

Under the assumption [42], [43], [44], [45] of independence between $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, we simplify this by taking $\boldsymbol{\Sigma}_{12} = \boldsymbol{0}$. Hence, we can use a more straightforward Jacobian-based propagation

$$\boldsymbol{\Sigma}_{\text{lo}} \approx \boldsymbol{J}_1\boldsymbol{\Sigma}_1\boldsymbol{J}_1^T + \boldsymbol{J}_2\boldsymbol{\Sigma}_2\boldsymbol{J}_2^T. \tag{13}$$

The computation of pose Jacobians in our highly linear systems is complex and challenging. We directly employ the adjoint representation for approximation. Let $\text{Ad}_{X_{12}}$ represent the adjoint of the relative pose $\boldsymbol{X}_{12}$; then, the covariance propagation can be approximated as

$$\boldsymbol{\Sigma}_{\text{lo}} \approx \text{Ad}_{X_{12}}\boldsymbol{\Sigma}_1\text{Ad}_{X_{12}}^T + \boldsymbol{\Sigma}_2. \tag{14}$$

B. Uncertainty Propagation in Factor Graph

We represent the uncertainty of different factors through their covariance matrices, which are inverted to form the corresponding information matrices. Then, the global information matrix $\boldsymbol{\Lambda}$ is synthesized from these individual matrices and is related to the global Jacobian matrix $\boldsymbol{J}$. For clarity, we consider a minimal case with just three poses $\boldsymbol{X} = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3\}$. Incorporating various factors between these poses, as shown in Fig. 3, we construct the total Jacobians and residual vectors as

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{J}_{\text{lo}}^{1,2} & -\boldsymbol{J}_{\text{lo}}^{1,2} & \boldsymbol{0}_{6\times6} \\ \boldsymbol{0}_{6\times6} & \boldsymbol{J}_{\text{lo}}^{2,3} & -\boldsymbol{J}_{\text{lo}}^{2,3} \\ \boldsymbol{J}_{\text{dm}}^{1} & \boldsymbol{0}_{6\times6} & \boldsymbol{0}_{6\times6} \\ \boldsymbol{0}_{6\times6} & \boldsymbol{J}_{\text{dm}}^{2} & \boldsymbol{0}_{6\times6} \\ \boldsymbol{0}_{6\times6} & \boldsymbol{0}_{6\times6} & \boldsymbol{J}_{\text{dm}}^{3} \\ \boldsymbol{0}_{3\times6} & \boldsymbol{J}_{\text{gf}}^{2} & \boldsymbol{0}_{3\times6} \\ \boldsymbol{0}_{3\times6} & \boldsymbol{0}_{3\times6} & \boldsymbol{J}_{\text{gf}}^{3} \\ \boldsymbol{J}_{\text{lc}}^{1,3} & \boldsymbol{0}_{6\times6} & \boldsymbol{J}_{\text{lc}}^{3,1} \\ \boldsymbol{0}_{6\times6} & \boldsymbol{J}_{\text{nm}}^{2,3} & -\boldsymbol{J}_{\text{nm}}^{2,3} \end{bmatrix}_{48\times18}, \quad \boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}_{\text{lo}}^{1,2} \\ \boldsymbol{r}_{\text{lo}}^{2,3} \\ \boldsymbol{r}_{\text{dm}}^{1} \\ \boldsymbol{r}_{\text{dm}}^{2} \\ \boldsymbol{r}_{\text{dm}}^{3} \\ \boldsymbol{r}_{\text{gf}}^{2} \\ \boldsymbol{r}_{\text{gf}}^{3} \\ \boldsymbol{r}_{\text{lc}}^{1,3} \\ \boldsymbol{r}_{\text{nm}}^{2,3} \end{bmatrix}. \tag{15}$$

The optimization process seeks to minimize the squared norm of the linearized residual vector

$$\delta\boldsymbol{x}^* = \arg\min_{\delta\boldsymbol{x}} \|\boldsymbol{r} + \boldsymbol{J}\delta\boldsymbol{x}\|^2. \tag{16}$$

A common approach to update the state vector involves using the inverse [46]

$$\delta\boldsymbol{x}^* = -(\boldsymbol{J}^\top\boldsymbol{J})^{-1}\boldsymbol{J}^\top\boldsymbol{r}, \quad \boldsymbol{X} \leftarrow \boldsymbol{X} \oplus \delta\boldsymbol{x}^*. \tag{17}$$

The accurate covariance matrix $\boldsymbol{W}$, a block-diagonal matrix of individual covariance matrices, can be defined as

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{\Omega}_{\text{lo}}^{1,2} & \boldsymbol{0}_{6\times6} & \boldsymbol{0}_{6\times6} & \cdots & \boldsymbol{0}_{6\times6} \\ \boldsymbol{0}_{6\times6} & \boldsymbol{\Omega}_{\text{lo}}^{2,3} & \boldsymbol{0}_{6\times6} & \cdots & \boldsymbol{0}_{6\times6} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{0}_{6\times6} & \boldsymbol{0}_{6\times6} & \cdots & \boldsymbol{\Omega}_{\text{lc}}^{1,3} & \boldsymbol{0}_{6\times6} \\ \boldsymbol{0}_{6\times6} & \boldsymbol{0}_{6\times6} & \cdots & \boldsymbol{0}_{6\times6} & \boldsymbol{\Omega}_{\text{nm}}^{2,3} \end{bmatrix}_{48\times48} \tag{18}$$

where the zero blocks signify the independent contributions of each factor. Then, the global information matrix $\boldsymbol{\Lambda}$ is formulated as

$$\boldsymbol{\Lambda} = \boldsymbol{J}^\top\boldsymbol{W}\boldsymbol{J}. \tag{19}$$

Applying the Cholesky decomposition [47] to $\boldsymbol{\Lambda}$ yields the covariance matrix $\boldsymbol{\Sigma}$, as follows:

$$\boldsymbol{\Lambda} = \boldsymbol{L}\boldsymbol{L}^\top, \quad \boldsymbol{\Sigma} = \boldsymbol{L}^{-\top}\boldsymbol{L}^{-1}. \tag{20}$$

In the covariance matrix $\boldsymbol{\Sigma}$, the diagonal elements denote variances and the off-diagonal elements represent covariances, highlighting the interdependence of state variables. Incremental algorithms such as iSAM2 [48] efficiently update graphs by exploiting the sparsity in $\boldsymbol{J}$ and $\boldsymbol{\Lambda}$, showcasing the evolving nature of uncertainty propagation in factor graphs.

VIII. Experimental Results

We conducted a comprehensive series of real-world experiments in various indoor and outdoor environments to rigorously assess the accuracy and robustness of our proposed system.

A. Experimental Setup

1) Hardware and Software Setups: In addition to utilizing the FusionPortable dataset [2], our experiments were conducted using three distinct sensor combinations:

1) Ouster-128 OS1 LiDAR with measurement noise of 3 cm, with a 200-Hz time-synchronized STIM300 IMU; shown in Fig. 1(a)
2) Pandar XT32 LiDAR with measurement noise of 1 cm, combined with a 100-Hz time-synchronized SBG IMU (`redbird_01` and `parkinglot_01`);
3) Pandar XT32 LiDAR with 1-cm measurement noise, coupled with a 700-Hz 3DM-GQ7 IMU (`redbird_02`).

We utilized two types of GT trajectories for validation:

1) an indoor MOCAP system providing 120-Hz six-DoF data with 1-cm accuracy; shown in Fig. 1(b)
2) an outdoor 50-Hz 3DM-GQ7 INS system, achieving a best case accuracy of 1.4 cm.

Prior maps were obtained using Leica MS60, BLK360, or RTC360 scanners. Fig. 1(c) showed a dense prior map collected with BLK360 in the garden_day sequence, along with the GT localization trajectory generated by the PALoc.Computational experiments were conducted on a desktop with an Intel i7 CPU, 96-GB DDR4 RAM, and 1-TB SSD. Point cloud processing was done using the Point Cloud Library and Open3D [49], while GTSAM [50] addressed the optimization challenges. Voxel filter sizes were set to 0.1 m for LiDAR frames and 0.5 m for prior maps.

2) Experimental Design: Our research entailed a comprehensive set of experiments to evaluate our method, covering trajectory evaluation, map evaluation, degeneracy analysis, ablation study, and run-time analysis. We conducted trajectory assessment experiments to measure the accuracy of our algorithm against real trajectories in both indoor and outdoor settings, including degenerate scenarios. These experiments are compatible with various LIO front ends. Recognizing the challenge of acquiring GT trajectories in practice, we also conducted map evaluation experiments to estimate the overall trajectory error. This baseline included:

1) LIO: FAST-LIO2 (FL2)[3], LIO-SAM (LS)[4], and LIO-Mapping (LM)[5];
2) map-based localization: HDL-Localization (HDL)[6] and ICP;
3) prior-assisted localization: FAST-LIO-Localization (FL2L).[7]

In our experiments, we utilized FL2 and LS as front-end odometries, termed PFL2 and PLS, respectively. To assess the effectiveness of the core modules of our algorithm, particularly DM and ZUPT factors, an ablation study and in-depth analyses of degenerate scenarios faced by ground robots were performed.

[3] [Online]. Available: https://github.com/hku-mars/FAST_LIO
[4] [Online]. Available: https://github.com/JokerJohn/LIO_SAM_6AXIS
[5] [Online]. Available: https://github.com/hyye/lio-mapping
[6] [Online]. Available: https://github.com/koide3/hdl_localization
[7] [Online]. Available: https://github.com/HViktorTsoi/FAST_LIO_LOCALIZATION

TABLE I
QUANTITATIVE COMPARISON OF ATE [CM] AND RPE [CM] ACROSS DIVERSE ENVIRONMENTS AND PLATFORMS USING DIFFERENT FRONT-END ODOMETRY METHODS (PLS AND PFL2)

Sequence	**PLS**		**PFL2**	
	ATE ↓	**RPE ↓**	**ATE ↓**	**RPE ↓**
Handheld Indoor				
MCR_slow	5.70	3.89	7.36	2.49
MCR_normal	7.71	4.95	10.12	3.78
Quadruped Robots Indoor				
MCR_slow_00	2.00	0.90	3.73	0.54
MCR_slow_01	2.93	1.51	2.81	0.70
MCR_normal_00	3.96	2.49	5.00	0.72
MCR_normal_01	3.58	0.93	5.23	0.88
Handheld Outdoor				
parkinglot_01*	–	–	26.10	3.06
redbird_01	–	–	9.98	1.34
redbird_02	–	–	10.32	1.34

* Degenerate scenarios. A dash (–) indicates sequences that were not tested.

We also conducted comprehensive comparisons to evaluate the performance of these core modules.

3) Evaluation Metrics: We employed widely used metrics such as absolute trajectory error (ATE) and relative pose error (RPE) for trajectory evaluation [51]. For map evaluation, we utilized metrics like accuracy (AC) [11], [12], and Chamfer distance (CD) [52]. We set the maximum distance for corresponding point pairs to 0.2 m, aligned the maps to exclude unrelated points, and used an error threshold of $\boldsymbol{\tau} = 0.1$ m for computing these evaluation metrics. Let $\mathcal{P}$ represent the point cloud sampled from the estimated map and the GT point cloud map $\mathcal{M}$. For a point $p \in \mathcal{P}$, we define its distance to the GT map as

$$\boldsymbol{d}(\boldsymbol{p}, \mathcal{M}) = \min_{\boldsymbol{m} \in \mathcal{M}} \|\boldsymbol{p} - \boldsymbol{m}\|. \tag{21}$$

Accuracy is defined as the average Euclidean distance between estimated points and their corresponding matches in the GT map, calculated as $\mathrm{AC} = \frac{1}{N}\boldsymbol{\Sigma}_{i=1}^{N}[\mathrm{dis}(\hat{p}_i, p_i) < \tau]$. The Chamfer distance calculates the average of the squared distances between each point in one set and its closest point in the other set

$$\mathrm{CD}_{\mathcal{P},\mathcal{M}} = \frac{1}{2\|\mathcal{P}\|}\sum_{\boldsymbol{p}\in\mathcal{P}} \boldsymbol{d}(\boldsymbol{p}, \mathcal{M})^2 + \frac{1}{2\|\mathcal{M}\|}\sum_{\boldsymbol{m}\in\mathcal{M}} \boldsymbol{d}(\boldsymbol{m}, \mathcal{P})^2. \tag{22}$$

B. Trajectory Evaluation

Table I showcases the ATE and RPE of our PLS and PFL2 algorithms, tested on various platforms, including handheld and quadruped robots. These data span both indoor and outdoor settings, underscoring the precision of our algorithm in trajectory generation. We directly obtain the trajectory ATE and RPE results of HDL for motion capture room (MCR)-related data from [2]. Compared to our proposed PFL2 and PLS, HDL frequently experiences localization failures in these quadruped robot scenarios. In the scenarios where HDL successfully runs, our PFL2 and PLS exhibit at least a 20% improvement in both ATE and RPE accuracy compared to it. This demonstrates the robustness and high precision of our algorithm. In indoor scenarios, particularly in the MCR-related datasets confined to approximately 8 m × 5 m rooms, the PFL2 algorithm exhibits overall trajectory accuracy below 5 cm for the quadruped platform, despite the presence of significant near-field measurement noise from the LiDAR. On handheld platforms, the accuracy is notably lower compared to that of the quadruped platform. Conversely, the PLS, utilizing a different front-end odometry, shows superior precision in these feature-rich confined spaces, highlighting the flexibility of our approach to adapt to SOTA odometry methods. Benefiting from our factor graph framework and the loosely coupled integration of pose fusion, our system exhibits remarkable adaptability to SOTA LiDAR-based odometry methods. This versatility and practicality underline the robustness and applicability of our approach in real-world scenarios.

Outdoor experiments with PFL2 were conducted in larger, approximately 1000 m, tracks, including both common campus pathways and areas with severe degeneration. In regular outdoor settings, PFL2 achieved an ATE of around 10 cm and an RPE of 1.5 cm. However, in degenerate scenarios, ATE accuracy significantly decreased to approximately 30 cm, while RPE remained relatively stable. These findings underscore the accuracy and robustness of our algorithm in diverse environments and its capacity to handle challenging scenarios, including degeneration.

C. Map Evaluation

In practical applications without available GT trajectories, evaluating the generated trajectory presents a challenge. Therefore, we assess the accuracy of the estimated maps, using them as an indirect measure of trajectory precision. Table II highlights the robustness and accuracy of our algorithm in various campus environments and across different platforms.

Compared to the FL2 and other LIO methods, our PFL2 achieves a substantial improvement in map AC, with at least a 50% enhancement. The CD is reduced by at least 30%. Furthermore, in contrast to map-based localization algorithms, PFL2 successfully generates trajectories across all data sequences, consistently achieving near-optimal accuracy. Although it did not attain the best results in the `MCR_normal` sequence, this is likely due to the higher noise levels in LiDAR measurements at close range. This underscores the ability of PFL2 in diverse scenarios and its superiority in leveraging map information for enhanced trajectory estimation. Our approach consistently achieves top-tier results in AC and CD metrics, outperforming SOTA techniques in demanding scenarios. Notably, the algorithm exhibits remarkable map accuracy and resilience in long-range degeneration situations, such as in the `corridor_day` and `escalator_day` datasets, marked by significant variations in Z-axis height [see Fig. 5(e)].

Fig. 4 displays the error area distribution between the estimated maps and the GT map for several sequences, applying a threshold of less than 0.2 m. Considering minor variations in the

TABLE II
MapAC [cm] and CD Comparison for 20-cm Threshold in Diverse Environments

	LM		LS		FL2		ICP		HDL		FL2L		PFL2	
Sequence	AC ↓	CD ↓	AC ↓	CD ↓	AC ↓	CD ↓	AC ↓	CD ↓	AC ↓	CD ↓	AC ↓	CD ↓	AC ↓	CD ↓
Handheld Outdoor														
garden_day	4.14	7.95	3.94	7.87	5.98	10.73	3.64	7.27	6.06	10.65	3.50	7.20	**3.48**	**7.03**
garden_night	4.36	8.08	3.92	7.83	5.91	10.78	3.23	7.06	6.12	10.71	3.52	7.23	**3.19**	**6.99**
canteen_day	5.65	9.45	5.48	9.24	6.32	10.51	4.86	8.23	6.59	10.63	5.59	9.02	**4.71**	**8.08**
canteen_night	5.60	9.60	5.29	9.61	6.77	10.87	5.07	8.42	6.93	10.96	5.56	8.94	**4.76**	**8.16**
Handheld Indoor														
escalator_day	5.40	9.10	8.65	15.09	6.92	10.36	×	×	7.66	11.28	4.29	7.51	**3.88**	**7.13**
building_day	10.11	19.05	7.65	15.01	6.68	12.63	6.72	8.54	7.11	11.35	4.19	7.88	**4.14**	**7.49**
corridor_day*	7.40	13.14	6.24	12.51	7.28	12.91	×	×	8.08	11.59	5.59	8.44	**3.99**	**5.79**
MCR_slow	7.66	13.84	4.08	6.54	6.19	8.55	**3.96**	**5.92**	×	×	×	×	4.63	6.52
MCR_normal	4.28	7.92	3.85	7.01	×	×	×	×	×	×	×	×	**3.71**	**5.86**
dynamic_02	9.58	18.07	9.78	18.49	9.63	17.98	4.13	8.09	7.44	10.96	4.62	8.34	**4.06**	**7.82**
dynamic_04	9.66	18.25	9.81	18.28	9.68	18.00	4.30	7.90	4.35	11.36	5.06	8.42	**4.20**	**7.67**

* Degenerate scenarios. × signifies a failure of the algorithm on the respective datasets.
Bold indicates the best accuracy, underline indicates the second best.

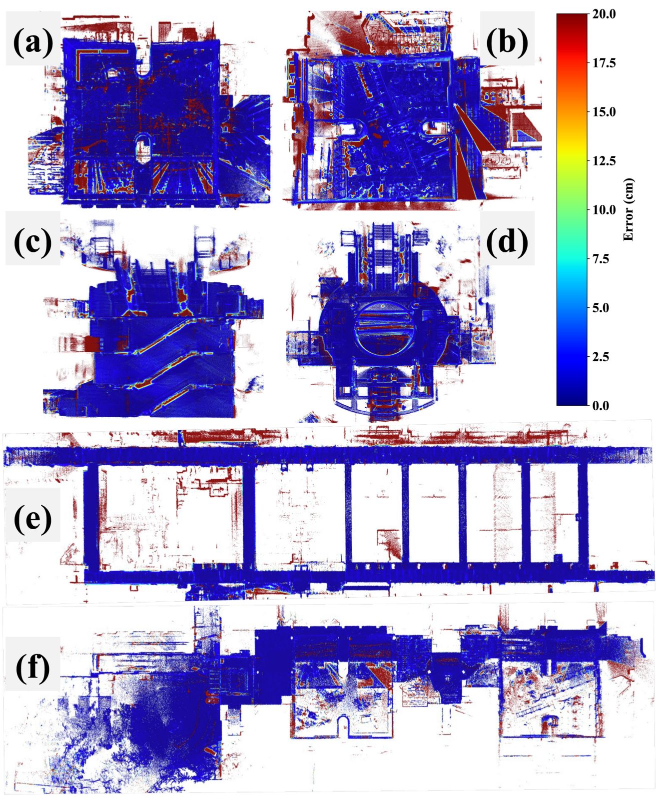


Fig. 4. Error map of diverse campus scenes. The degree of color transition from blue to red indicates an increasing error in the mapped area. (a) `garden_day` (259.5 m). (b) `canteen_day` (253.1 m). (c) XZ view of `escalator_day` (600.1 m). (d) XY view of `escalator_day` (600.1 m) with ceiling removal. (e) `corridor_day` (656.4 m). (f) `building_day` (717.8 m).

map and differences in scanning patterns between the LiDAR and GT Laser scanner, most areas estimated by our algorithm maintain an accuracy of nearly 3 cm. This level of precision, comparable to LiDAR measurement noise, indirectly verifies the accuracy and effectiveness of our trajectory estimation, supporting its use as a GT trajectory alternative.

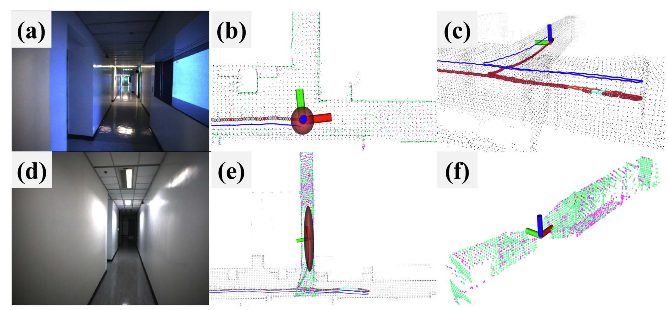


Fig. 5. Translation degeneracy analysis in corridor. (a) and (d) represent real-world corridor scenes. The black point cloud represents the prior map, and the red sphere with coordinate axes represents the relative constraint strength in the XYZ dimensions but is unrelated to the overall size of the ellipsoid. The flatter the ellipsoid, the more severe the degeneration in a specific dimension. The blue and light blue trajectories and the red points on the trajectories represent the FL2 odometry trajectory, our algorithm trajectory, and the pose with DM constraints. Our algorithm eliminates Z-axis drift error while ensuring robustness in a U-turn intersection (c). The point clouds of different colors in (f) indicate the corresponding number of constraints in XYZ dimensions (see Section VI-B).

D. Degeneracy Analysis

In many ground robot scenarios, a majority of the point cloud data is constituted by ground points, effectively constraining the roll, pitch, and Z dimensions during pose estimation. This often leads to degeneracy in translation along the XY-axis or rotation in the yaw direction. We thoroughly analyze this degeneracy in two distinct environments: the `corridor_day` (X-axis degeneracy) and `parkinglot_01` (yaw-axis degeneracy).

Fig. 5 presents a detailed analysis of degeneracy in the `corridor_day` scenario, especially in narrow passages [see Fig. 5(d)], where degeneracy is identified and high-precision pose estimation is maintained. The red ellipses in the figures indicate the condition number in translation dimensions at different corridor sections. At feature-rich intersections [see Fig. 5(a)], the constraints are well balanced, as shown by the uniform ellipses in Fig. 5(b). However, in the narrow corridors

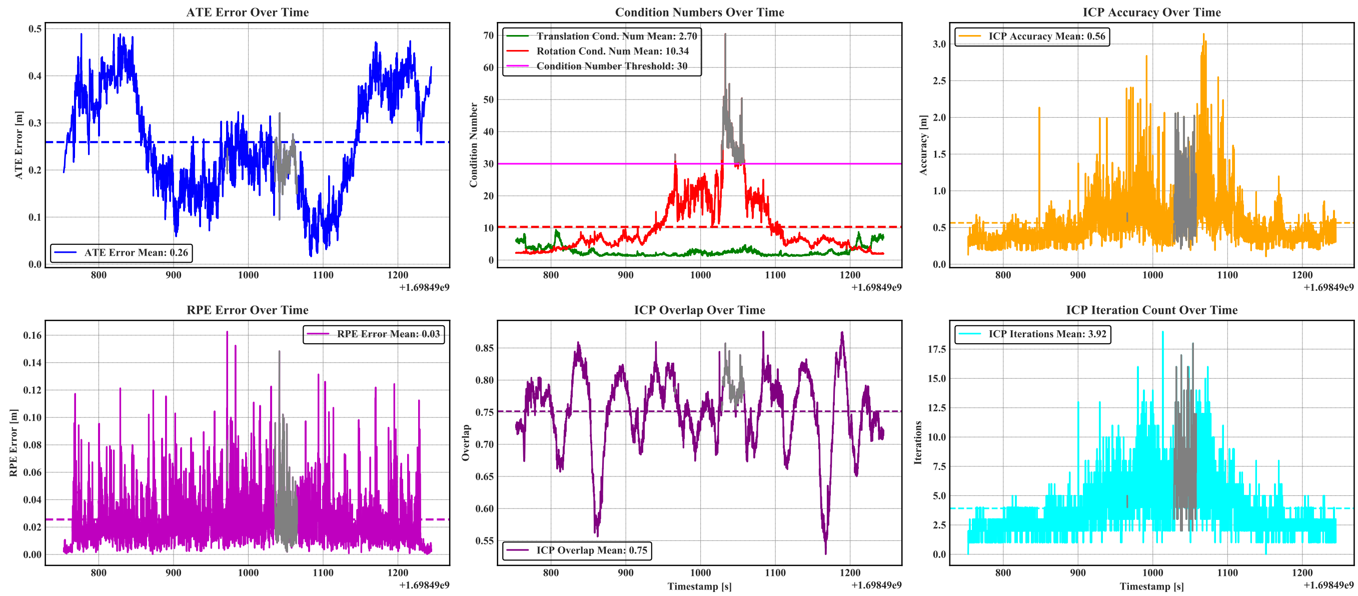


Fig. 6. Degeneracy scenario analysis. The figure provides a detailed analysis of the DM factor in PALoc, applied to the `parkinglot_01` dataset with yaw degeneracy. It includes a time-variant depiction of ATE, ICP accuracy, ICP iteration count, ICP overlap rate, and the condition numbers for translation and rotation. A specific threshold for the condition number, marked by the gray area, is set to identify degeneracy. It is observed that during rotational degeneracy in the scene, there is a notable decline in ICP accuracy, and ICP overlap rate, along with a significant increase in ICP iteration counts. These observations align well with the actual scenarios encountered in the parking lot setting.

[see Fig. 5(e)], the elongated ellipses along the X-axis signify severe X-axis degeneracy, aligning with our theoretical analysis in Section VI. Fig. 5(f) investigates degeneration due to limited LiDAR points constraining the X- and Z-axes, leading to Z-axis error accumulation in narrow corridors, showcasing an improvement over the original FL2 LiDAR odometry. The blue and light blue trajectories in Fig. 5(c) represent the LIO and PFL2 algorithms. The effectiveness of our DM module in mitigating drift, especially in the Z-axis for LIO, is evident, corroborating the map evaluation results in Table II.

Fig. 7 demonstrates the robustness of our PFL2 algorithm compared to FL2L using the `parkinglot_01` dataset. Specifically, Fig. 7(c) highlights the XY uncertainty visualization in a targeted dataset region, where the relative size of spheres indicates their uncertainty levels. This visualization guides us in determining the reliability of different dimensions in each pose estimate, underscoring the robustness of our algorithm and how uncertainty aids in assessing pose reliability.

Fig. 6 presents the performance analysis of the DM module in the `parkinglot_01` sequence using the PFL2, tracking various error metrics over time. Initially, the ATE exhibits higher values due to inaccuracies in initialization. Throughout the process, the translation condition number maintains a very low level, whereas the rotation condition number indicates severe degeneracy in the parking area, leading to similar trends in both ATE and RPE. However, degeneracy is relative, and we set a threshold of 30 for condition number detection only in the first step. This means that the periods when degeneracy is detected do not necessarily coincide with reduced point cloud matching accuracy—this would only be the case if the initial values were near perfect. Our approach is to discard the DM constraints when extreme values in the condition number are observed or when point cloud matching fails. Hence, this explains why the trends in ATE and RPE do not always align perfectly with the changes in the condition number. The data reveal how our PFL2, with its nuanced degeneracy handling, effectively manages various challenges in diverse environments.

TABLE III
ABLATION STUDY OF ATE [CM], RPE [CM], AND AC [CM] ON MCR_NORMAL_00

Method	ATE ↓	RPE ↓	AC ↓
FL2	9.80	1.20	5.47
w/o LO	22.3	4.93	7.71
w/o DM	9.61	1.10	5.86
w/o LC	5.47	<u>0.96</u>	5.38
w/o NM	5.38	1.58	4.11
w/o GF	<u>5.03</u>	1.19	4.06
All	**5.00**	**0.72**	**3.93**

Bold indicates the best performance; <u>underlined</u> signifies the second-best.

E. Ablation Evaluation

To systematically assess the contributions of different factors within our proposed system, we conducted an ablation study on the `MCR_normal_00`. This evaluation focused on the impact of the LO, LC, NM, GF, and DM on the overall accuracy of our system. Table III presents a comparative analysis of the ATE, RPE, and map AC of our proposed method when excluding these factors. Our findings reveal that the DM, GF, and NM are critical components, contributing significantly to the enhancement of trajectory accuracy. The exclusion of either

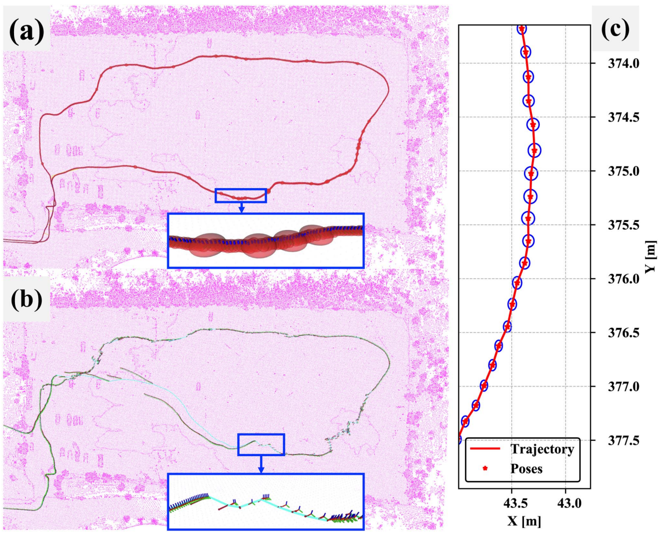


Fig. 7. Trajectory comparison and uncertainty visualization in rotation degenerate parking lot. (a) Robust performance of the PFL2 under conditions of rotational degeneracy, operating smoothly. Red spheres indicate the uncertainty in the translation part; larger spheres denote greater uncertainty, with reduced uncertainty along the Z-axis. Occasional large covariance spheres represent severe degeneracy causing matching errors, leading to the exclusion of these map factors. (b) Trajectory of FL2L in blue, with coordinate axes representing the poses, clearly showing direct localization failure in scenarios with rotational degeneracy. (c) Pose uncertainty in the X and Y dimensions at a 95% confidence; due to the scale of the visualization, the uncertainty in Y is almost comparable to X, aligning with the observed real-world scenario.

TABLE IV
PER-FRAME AVERAGE EXECUTION TIME [MS] COMPARISON OF KEY MODULES ON TWO DATASETS

Dataset	DM		GO		Total	
	PFL2	ICP	PFL2	ICP	PFL2	ICP
parkinglot_01	**205.72**	289.69	55.43	**43.13**	**261.35**	333.01
redbird_02	**191.72**	270.81	76.46	**60.65**	**269.29**	331.46

factor results in a notable increase in ATE, AC, and RPE. This ablation evaluation highlights the coherent and complementary nature of the components of our proposed method, ultimately resulting in a robust and accurate system.

F. Run-Time Analysis

We rigorously evaluate the performance of our proposed algorithm within extensive outdoor scenes, with a particular emphasis on the core modules that predominantly influence computational time: the DM and graph optimization (GO). Our DM module has been benchmarked against the point-to-plane ICP implementation from Open3D [49], under a strict parameter alignment such as K-nearest neighbors (KNN) radius and maximum iterations, ensuring a fair assessment of our algorithm's efficiency. Table IV elucidates our findings, revealing that our DM module reduces the per-frame processing duration by approximately 30% compared to the traditional ICP baseline, contributing to an overall minimization of at least 20% in the average per-frame

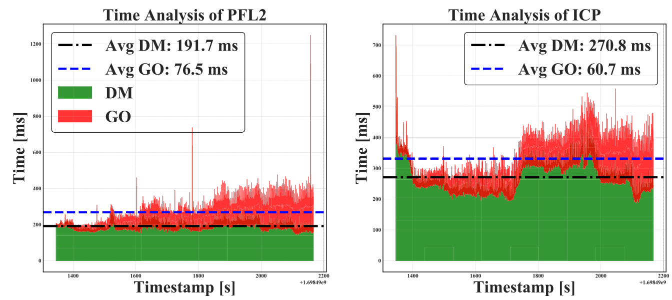


Fig. 8. Performance analysis and comparison on `redbird_02`.

execution time. Despite this significant enhancement, the performance gains in the GO module were marginal. We attribute this to the meticulous computation of covariance for each system factor, which slightly tempered the convergence pace during optimization. Fig. 8 graphically contrasts the execution times on the `redbird_02` dataset, underscoring the enhanced performance of our algorithm over the ICP method and affirming the robustness of our approach in large-scale outdoor robotic applications.

G. Discussion

Our proposed trajectory generation method exhibits competitive performance across diverse scenarios and platforms. Nonetheless, several known issues warrant future research. First, the factor graph size expands as the scene enlarges, resulting in reduced system efficiency. Second, the flexibility to arbitrarily replace the front-end odometry complicates the isolated analysis of odometry degeneration. In the degeneration detection aspect, adjusting certain thresholds for various degenerate scenes is necessary. Finally, while we have estimated the uncertainty of the poses, a quantitative assessment has not been conducted, leaving the results of uncertainty lacking experimental support.

IX. CONCLUSION

This article introduced a novel approach for generating dense six-DoF GT trajectories, enhancing SLAM evaluation. We employed a factor GO technique, overcoming the limitations of tracking-based methods and effectively handling degenerate and stationary scenarios. The significant advancement of our method lies in the derivation of covariance for each factor, improving our understanding of pose uncertainty. In addition, we integrated map evaluation criteria into this system, contributing to the precision of trajectory generation. Despite the reliance on prior maps, our open-source solution marks a notable step forward in SLAM benchmark augmentation. Future efforts will focus on expanding the applicability of our methods in diverse scenarios in robotics research.

ACKNOWLEDGMENT

The authors would like to express their gratitude to the editors and reviewers for their critical suggestions regarding the methodology and experiments of this study. The authors would like to thank Bowen Yang, Xieyuanli Chen, and Binqian

Jiang for their valuable corrections and insights on the grammar and logic of this article. Finally, the authors would like to gratefully acknowledge the assistance of ChatGPT in refining the manuscript.

References

[1] G. C. Sharp, S. W. Lee, and D. K. Wehe, "ICP registration using invariant features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 90–102, Jan. 2002.

[2] J. Jiao et al., "FusionPortable: A multi-sensor campus-scene dataset for evaluation of localization and mapping accuracy on diverse platforms," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 3851–3856.

[3] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.

[4] J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim, "Complex urban dataset with multi-level sensors from highly diverse urban environments," *Int. J. Robot. Res.*, vol. 38, no. 6, pp. 642–657, Apr. 2019.

[5] M. Burri et al., "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016.

[6] J. Delmerico, T. Cieslewski, H. Rebecq, M. Faessler, and D. Scaramuzza, "Are we ready for autonomous drone racing? The UZH-FPV drone racing dataset," in *Proc. Int. Conf. Robot. Autom.*, 2019, pp. 6713–6719.

[7] 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," *Int. J. Robot. Res.*, vol. 41, no. 3, pp. 270–280, 2022.

[8] L. Zhang et al., "Hilti-oxford dataset: A millimeter-accurate benchmark for simultaneous localization and mapping," *IEEE Robot. Autom. Lett.*, vol. 8, no. 1, pp. 408–415, Jan. 2023.

[9] M. Ramezani, Y. Wang, M. Camurri, D. Wisth, M. Mattamala, and M. Fallon, "The newer college dataset: Handheld LiDAR, inertial and vision with ground truth," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 4353–4360.

[10] H. Sier, Q. Li, X. Yu, J. P. Queralta, Z. Zou, and T. Westerlund, "A benchmark for multi-modal LiDAR SLAM with ground truth in GNSS-denied environments," *Remote Sens.*, vol. 15, Oct. 2022, Art. no. 3314.

[11] Q. Xu and W. Tao, "Planar prior assisted patchmatch multi-view stereo," in *Proc. Conf. Assoc. Adv. Artif. Intell.*, 2020, pp. 12516–12523.

[12] H. Aanæs, R. R. Jensen, G. Vogiatzis, E. Tola, and A. B. Dahl, "Large-scale data for multiple-view stereopsis," *Int. J. Comput. Vis.*, vol. 120, no. 2, pp. 153–168, Nov. 2016.

[13] S. A. Kay, S. Julier, and V. M. Pawar, "Challenges of benchmarking SLAM performance for construction specific applications," in *Proc. IEEE Int. Conf. Robot. Autom. Workshop*, 2018. [Online]. Available: https://drive.google.com/file/d/11ic-sv90fAo8YKGVq03_Qgg_enhLRXhb/view

[14] A. Huang et al., "A high-rate, heterogeneous data set from the DARPA urban challenge," *Int. J. Robot. Res.*, vol. 29, pp. 1595–1601, Nov. 2010.

[15] T. Peynot, S. Scheding, and S. Terho, "The Marulan data sets: Multi-sensor perception in a natural environment with challenging conditions," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1602–1607, Nov. 2010.

[16] G. Pandey, J. R. McBride, and R. M. Eustice, "Ford Campus vision and LiDAR data set," *Int. J. Robot. Res.*, vol. 30, no. 13, pp. 1543–1552, 2011.

[17] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford RobotCar dataset," *Int. J. Robot. Res.*, vol. 36, no. 1, pp. 3–15, Nov. 2016.

[18] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stückler, and D. Cremers, "The TUM VI benchmark for evaluating visual-inertial odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1680–1687.

[19] H. Ye, H. Huang, and M. Liu, "Monocular direct sparse localization in a prior 3D surfel map," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 8892–8898.

[20] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3D-NDT," *J. Field Robot.*, vol. 24, no. 10, pp. 803–827, 2007.

[21] M. Doostmohammadian and U. A. Khan, "On the genericity properties in distributed estimation: Topology design and sensor placement," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 195–204, Apr. 2013.

[22] M. Doostmohammadian, A. Taghieh, and H. Zarrabi, "Distributed estimation approach for tracking a mobile target via formation of UAVs," *IEEE IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 4, pp. 3765–3776, Oct. 2022.

[23] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," *Robot. Sci. Syst.*, vol. 2, no. 9, pp. 1–9, 2014.

[24] J. Zhang and S. Singh, "Visual-LiDAR odometry and mapping: Low-drift, robust, and fast," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 2174–2181.

[25] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[26] H. Ye, Y. Chen, and M. Liu, "Tightly coupled 3D LiDAR inertial odometry and mapping," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 3144–3150.

[27] C. Qin, H. Ye, C. E. Pranata, J. Han, S. Zhang, and M. Liu, "LINS: A LiDAR-inertial state estimator for robust and efficient navigation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 8899–8906.

[28] J. Jiao, H. Ye, Y. Zhu, and M. Liu, "Robust odometry and mapping for multi-LiDAR systems with online extrinsic calibration," *IEEE Trans. Robot.*, vol. 38, no. 1, pp. 351–371, Feb. 2022.

[29] H. Huang et al., "On bundle adjustment for multiview point cloud registration," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 8269–8276, Oct. 2021.

[30] J. Lin and F. Zhang, "R3 live: A robust, real-time, RGB-colored, LiDAR-inertial-visual tightly-coupled state estimation and mapping package," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2022, pp. 10672–10678.

[31] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: An open-source library for real-time metric-semantic localization and mapping," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 1689–1696.

[32] V. Ila, J. M. Porta, and J. Andrade-Cetto, "Information-based compact pose SLAM," *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 78–93, Feb. 2010.

[33] V. Kubelka, M. Vaidis, and F. Pomerleau, "Gravity-constrained point cloud registration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 4873–4879.

[34] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[35] B. Jiang and S. Shen, "A LiDAR-inertial odometry with principled uncertainty modeling," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 13292–13299.

[36] A. Tagliabue et al., "LION: LiDAR-inertial observability-aware navigator for vision-denied environments," in *Proc. 17th Int. Symp. Exp. Robot.*, 2021, pp. 380–390.

[37] T. Tuna, J. Nubert, Y. Nava, S. Khattak, and M. Hutter, "X-ICP: Localizability-aware LiDAR registration for robust localization in extreme environments," *IEEE Trans. Robot.*, vol. 40, pp. 452–471, 2024.

[38] J. Jiao et al., "Greedy-based feature selection for efficient LiDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 5222–5228.

[39] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 809–816.

[40] 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 *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Jan. 2021, pp. 5135–5142.

[41] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized LiDAR odometry and mapping on variable terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 4758–4765.

[42] T. D. Barfoot, *State Estimation for Robotics*. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[43] J. G. Mangelson, M. GhaffariZ, R. Vasudevan, and R. M. Eustice, "Characterizing the uncertainty of jointly distributed poses in the Lie algebra," *IEEE Trans. Robot.*, vol. 36, no. 5, pp. 1371–1388, Oct. 2020.

[44] T. D. Barfoot and P. T. Furgale, "Associating uncertainty with three-dimensional poses for use in estimation problems," *IEEE Trans. Robot.*, vol. 30, no. 3, pp. 679–693, Jun. 2014.

[45] Y. Chen, S. Huang, L. Zhao, and G. Dissanayake, "Cramé–Rao bounds and optimal design metrics for pose-graph SLAM," *IEEE Trans. Robot.*, vol. 37, no. 2, pp. 627–641, Apr. 2021.

[46] J. Solá et al., "A micro lie theory for state estimation in robotics," 2021, *arXiv: 1812.01537*.

[47] V. Ila, L. Polok, M. Solony, and P. Svoboda, "SLAM++—A highly efficient and temporally scalable incremental SLAM framework," *Int. J. Robot. Res.*, vol. 36, no. 2, pp. 210–230, 2017.

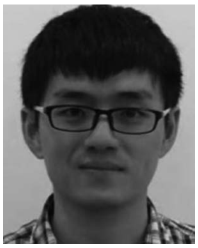
[48] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 3281–3288.

[49] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing," 2018, *arXiv:1801.09847*.

[50] F. Dellaert and GTSAM contributors, "borglab/gtsam," May 2022. [Online]. Available: https://github.com/borglab/gtsam

[51] Z. Zhang and D. Scaramuzza, "Rethinking trajectory evaluation for SLAM: A probabilistic, continuous-time approach," 2019, *arXiv:1906.03996*.

[52] T. Wu, L. Pan, J. Zhang, T. Wang, Z. Liu, and D. Lin, "Balanced chamfer distance as a comprehensive metric for point cloud completion," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 29088–29100.



Xiangcheng Hu received the B.Sc. degree from the North University of China, Taiyuan, China, in 2017, and the M.S. degree from Beihang University, Beijing, China, in 2020, both in LiDAR localization and mapping. He is currently working toward the Ph.D. degree with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong.

Linwei Zheng received the B.Eng. degree in electrical engineering and automation from Xi'an Jiaotong University, Xi'an, China, in 2017, and the M.Sc. degree in electronic engineering from the Hong Kong University of Science and Technology, Hong Kong, in 2018. He is currently working toward the Ph.D. degree with the Intelligent Autonomous Driving Center, Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China.

Jin Wu received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China. He is currently working toward the Ph.D. degree with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong.

From 2013 to 2014, he was an Exchange Undergraduate Student with Groep T, Katholieke Universiteit Leuven, Leuven, Belgium. From 2019 to 2020, he was with Tencent Robotics X, Shenzhen, China.

Ruoyu Geng received the B.Eng. degree in automobile engine from Jilin University, Changchun, China, in 2017, and the M.Sc. degree in electronic engineering in 2020 from the Hong Kong University of Science and Technology, Hong Kong, where she is currently working toward the Ph.D. degree in robotics and autonomous systems.

Yang Yu received the B.Eng. degree from Donghua University, Shanghai, China, in 2012, the M.Eng. degree from the Illinois Institute of Technology, Chicago, IL, USA, in 2014, and the Ph.D. degree from the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, all in electrical engineering.

Hexiang Wei received the B.Eng. degree in mechanical engineering from Northeastern University, Shenyang, China, in 2020. He is currently working toward the Ph.D. degree with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong.

His research interests include simultaneous localization and mapping, sensor fusion, and robotics.

Xiaoyu Tang (Member, IEEE) received the B.S. degree from South China Normal University, Guangzhou, China, in 2003, and the M.S. degree from Sun Yat-sen University, Guangzhou, in 2011. He is currently working toward the Ph.D. degree in information system development with the School of Physics and Telecommunication Engineering, South China Normal University.

Mr. Tang is a Member of the IEEE International Conference on Information Communication and Signal Processing Technical Committee.

Lujia Wang (Member, IEEE) received the Ph.D. degree from the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, in 2015.

From 2016 to 2021, she was an Associate Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. She is currently a Senior Lecturer of the Red Bird M.Phil. Program with the Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China.

Jianhao Jiao (Member, IEEE) received the B.Eng. degree in instrument science from Zhejiang University, Hangzhou, China, in 2017, and the Ph.D. degree from the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, in 2021.

He was a Research Associate with the Hong Kong University of Science and Technology. He is currently a Senior Research Fellow with the Department of Computer Science, University College London, London, U.K.

Dr. Jiao is an Associate Editor for 2024 IEEE International Conference on Intelligent Robots and Systems.

Ming Liu (Senior Member, IEEE) received the Ph.D. degree from the Department of Mechanical and Process Engineering, ETH Zurich, Zurich, Switzerland, in 2013, under the supervision of Prof. Roland Siegwart.

He is currently an Associate Professor of Robotics and Autonomous Systems with the Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China, as a Thrust Head, where he is also the Director of the Intelligent Autonomous Driving Center, coordinating research and development on robotics and logistics. His research works have been widely reported in more than 300 journal and conference papers.

Dr. Liu was an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS and a Committee Member of 2021 Robotics: Science and Systems Conference.