UD-LIOM: A Universal Tightly Coupled Direct LiDAR-Inertial Odometry and Mapping Framework

Baosheng Zhang, Lin Zhang, *Senior Member, IEEE*, Shengjie Zhao, *Senior Member, IEEE*, and Yicong Zhou, *Senior Member, IEEE*

***Abstract*—LiDAR-based SLAM, as a key high-precision pose measurement technology for autonomous driving and embodied intelligence, has attracted considerable attention. In addition, to accommodate diverse platforms, various types of LiDARs have emerged in recent years. Thus, it is imperative to design SLAM systems capable of adapting to different types of LiDARs. However, variations in scanning patterns among different LiDAR types impose significant challenges to the cross-LiDAR compatibility of existing SLAM methods. Moreover, to enhance the robustness and accuracy of these LiDAR SLAM systems, many studies have integrated inertial data into the framework. However, existing LiDAR-inertial fusion methods are mostly tailored for specific LiDAR types, and the residual-level deep fusion of IMU constraints with universal LiDAR processing pipelines remains to be further explored. While state-of-the-art (SOTA) methods have realized tight coupling of IMU and LiDAR in back-end optimization, their poor adaptability to diverse LiDAR scanning patterns and shallow fusion at the factor/filter level restricts practical deployment in multisensor scenarios. To address these challenges of cross-LiDAR type universality and multisensor data synergy in LiDAR SLAM, we propose universal tightly-coupled direct LiDAR-inertial odometry and mapping framework (UD-LIOM), a universal tightly coupled LiDAR–inertial odometry and mapping framework that makes two major contributions. First, unlike existing direct LiDAR SLAM methods that process each LiDAR point uniformly, UD-LIOM leverages the common scanning "ring" structure across different LiDAR types to develop a universal LiDAR measurement data classification strategy. By dividing each LiDAR scan into edge and planar points and performing dedicated registration with edge/planar submaps, UD-LIOM builds more precise geometric constraints, thus enabling cross-type compatibility with diverse LiDAR scanning patterns and enhancing pose estimation accuracy. Second, in terms of IMU fusion, UD-LIOM integrates IMU preintegration for front-end motion distortion correction and fuses IMU preintegration residuals with LiDAR measurement residuals into a unified multiconstraint optimization model, realizing residual-level deep fusion to boost system robustness across complex scenarios. Experimental results on public and self-collected datasets demonstrate that UD-LIOM achieves superior measurement accuracy and system robustness compared with other SOTA methods. For full reproducibility, all code and data are available at https://cslinzhang.github.io/UD-LIOM**

***Index Terms*—LiDAR-inertial odometry, multisensor fusion measurement, pose measurement optimization, SLAM, universal measurement system.**

I. Introduction

TO ACCOMMODATE diverse platforms and application scenarios, various types of LiDARs have emerged in recent years. However, existing LiDAR SLAM methods exhibit a significant lag in iterative development, as they are initially designed for only a few specific types of LiDAR and have failed to fundamentally address the challenge of measurement compatibility across different LiDAR types [1], [2]. This limitation gives rise to multiple challenges: a single SLAM system cannot be seamlessly deployed across different platforms; upgrading to new LiDAR types often requires redesigning specialized algorithms, sharply increasing development costs. Moreover, limited compatibility reduces adaptability to LiDAR variations, easily causing feature mismatches, registration failures, and localization drift, and even safety risks such as collisions or robot tipping [3], [4], [5]. Although several cross-LiDAR type SLAM solutions [6], [7], [8] have been proposed in recent years, most of them remain confined to traditional mechanical LiDARs and mainstream nonrepetitive scanning solid-state LiDARs (e.g., Livox AVIA). Therefore, developing an efficient multisensor SLAM system capable of adapting to a wider variety of LiDAR types and meeting the demands of future LiDAR technologies remains an urgent research problem in instrumentation and measurement (I&M).

Existing LiDAR SLAM methods can be roughly categorized into two categories: indirect methods and direct methods. Indirect methods, typified by LeGO-LOAM [9] and LiLi-OM [10], rely primarily on feature extraction to establish correspondences, thereby constructing constraint models. However, such methods exhibit strong dependence on LiDAR scanning pattern—specific feature extraction strategies struggle to adapt to LiDARs with different scanning patterns, significantly limiting their universality. In contrast, direct methods (e.g., FSAT-LIO2 [6] and its variants [11]) eliminate the need for feature extraction. However, the local normal vec-

Received 16 January 2026; revised 11 March 2026; accepted 28 March 2026. Date of publication 16 April 2026; date of current version 24 April 2026. This work was supported in part by the National Natural Science Foundation of China under Grant 62272343 and in part by the Fundamental Research Funds for the Central Universities. The Associate Editor coordinating the review process was Dr. Luca Santoro. *(Corresponding author: Lin Zhang.)*

Baosheng Zhang, Lin Zhang, and Shengjie Zhao are with the School of Computer Science and Technology, Tongji University, Shanghai 201804, China (e-mail: baoshengzhang@tongji.edu.cn; cslinzhang@tongji.edu.cn; shengjiezhao@tongji.edu.cn).

Yicong Zhou is with the Department of Computer and Information Science, University of Macau, Macau, China (e-mail: yicongzhou@um.edu.mo).

Digital Object Identifier 10.1109/TIM.2026.3684679

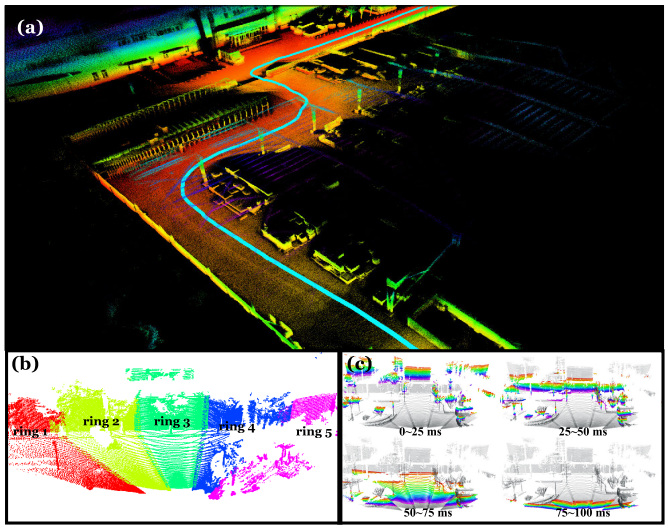

Fig. 1. (a) Localization and mapping result of the proposed method utilizing RSM1 solid-state LiDAR. (b) Scanning pattern of the RSM1 solid-state LiDAR presents the measurement results of different scanning "rings," with each "ring" depicted in a distinct color. The overlapping regions between adjacent "rings" introduce additional errors due to intrinsic parameter deviations of the LiDAR, thereby affecting the overall system accuracy. (c) Points from the RSM1 are colorized according to their timestamps with rainbow color scale, while the gray regions represent the complete scan.

tor fitting mechanism applied uniformly to each point leads to the loss of a substantial amount of valid point cloud information—a drawback particularly evident in solid-state LiDARs with a small field of view (FoV), which directly undermines the robustness of these methods. Furthermore, for emerging repetitive-scanning solid-state LiDARs (such as the RSM1 LiDAR shown in Fig. 1(b) and (c), internal parameter errors cause consistency deviations in point cloud data when adjacent "rings" repeatedly scan the same region. This characteristic further exacerbates the accuracy degradation of traditional direct SLAM methods based on local normal vector estimation.

Additionally, to enhance the robustness and accuracy of SLAM systems, numerous LiDAR-inertial fusion frameworks have been proposed. State-of-the-art (SOTA) approaches such as LIO-SAM, FAST-LIO2, and LINS achieve tight LiDAR–inertial coupling in back-end optimization. Nevertheless, most existing methods implement fusion at the module level, typically via factor-graph-based parallel fusion or filter-based serial fusion. In addition, these tightly coupled strategies are generally tailored to specific LiDAR types, and their feature extraction and registration schemes lack universality for diverse scanning patterns. Consequently, residual-level LiDAR-inertial integration within a universal LiDAR processing pipeline remains an open challenge for cross-sensor compatibility.

As mentioned above, despite decades of advancements in LiDAR-inertial SLAM research, two critical research gaps remain unresolved: 1) enhancing measurement compatibility and accuracy across LiDAR types and 2) achieving deeper fusion of IMU and LiDAR measurement data for universal systems. These two unresolved challenges are also the focus of this article. In this article, we propose a universal tightly coupled direct LiDAR-inertial odometry and mapping framework, UD-LIOM—essentially a novel multisensor fusion system for high-precision localization and mapping. Specifically, as a tightly coupled LiDAR–inertial SLAM framework, UD-LIOM first performs temporal synchronization between LiDAR point clouds and IMU data, followed by motion distortion correction of the raw point clouds using the synchronized IMU measurements. The de-distorted points are then classified into edge and planar points, which are, respectively, registered with their corresponding submaps. Finally, a comprehensive optimization model is constructed by fusing LiDAR measurements with IMU preintegration, thereby achieving high-precision system pose estimation.

Unlike existing direct methods that uniformly process each point, this study is based on the common "ring" scanning structure inherent to various types of LiDAR. Each point is classified into three categories—valid edge points, planar points, and invalid points—according to the local range smoothness of its neighboring points within the same "ring." Subsequently, the retained valid edge points and planar points are used to establish correspondences with the corresponding edge submap and planar submap, respectively. This not only constructs a more refined LiDAR measurement residual constraint model, but also enables incremental construction of the edge submap and planar submap.

By reconstructing the feature processing logic and innovatively designing a classified submap registration mechanism based on the common "ring" structure of diverse LiDAR types, this method fundamentally resolves incompatibility issues caused by scanning pattern discrepancies. This structural optimization of the entire feature classification and registration pipeline preserves more effective point-level features and breaks the adaptation limitation of traditional methods (e.g., LOAM) to specific LiDAR types. Furthermore, UD-LIOM integrates residual-level deep LiDAR-inertial fusion into the universal LiDAR processing pipeline: IMU preintegration is used for front-end motion distortion correction, and IMU preintegration residuals are jointly and equally optimized with LiDAR edge/planar geometric residuals in the back-end multiconstraint optimization model at each iteration. This residual-level deep fusion realizes tighter coupling of LiDAR and IMU information than the traditional factor/filter-level fusion, and the systematic integration with universal LiDAR processing makes the system maintain high robustness and accuracy while adapting to diverse LiDAR types, which is the key difference from existing fusion methods tailored for specific LiDARs.

It is important to clarify the distinction between this work and our prior study, D-LIOM [12]. D-LIOM focuses on enabling high-precision 3-D tightly coupled direct LiDAR-inertial odometry by integrating probabilistic scan-to-submap direct registration with IMU preintegration and bias estimation, under the assumption of relatively fixed and homogeneous LiDAR scanning configurations, which limits its adaptability to heterogeneous LiDARs. In contrast, this work addresses a fundamentally different problem: achieving measurement universality across heterogeneous LiDAR types with diverse and even repetitive scanning patterns. To this

end, UD-LIOM abandons the probabilistic direct scan-to-submap paradigm adopted in D-LIOM and instead introduces a novel ring-structure-driven point classification and geometry-aware submap registration strategy. By exploiting the scanline ("ring") structure shared across different LiDARs, points are classified into edge and planar categories based on local range smoothness, and dedicated geometric residuals are established with corresponding submaps, thereby fundamentally decoupling registration performance from sensor-specific scanning mechanisms. Moreover, IMU preintegration is incorporated as an explicit constraint jointly optimized with LiDAR geometric residuals in a unified multiconstraint framework, enabling deeper LiDAR-inertial coupling than in D-LIOM. As a result, UD-LIOM is not an extension of D-LIOM but a fundamentally distinct universal LiDAR-inertial measurement framework, designed to support mechanical LiDARs as well as both nonrepetitive and repetitive scanning solid-state LiDARs, addressing an urgent need for cross-sensor compatibility in I&M.

The main contributions of this work are summarized as

1) A novel universal tightly coupled direct LiDAR-inertial odometry and mapping framework, UD-LIOM, is proposed. By employing a tightly coupled direct strategy, the proposed method integrates a universal point cloud classification and registration scheme with a residual-level multiconstraint optimization model, which enables strong adaptation to diverse LiDAR types while improving system robustness and accuracy.
2) A universal point cloud classification and registration pipeline is proposed for various LiDAR scanning patterns, via structural reconstruction of smoothness-based feature processing and innovative classified submap design. Leveraging the common LiDAR scanning "ring," all valid edge/planar points are retained with a dual-threshold strategy for dedicated submap registration, enabling cross-type LiDAR compatibility and mitigating interring inconsistencies in repetitive-scanning solid-state LiDARs.
3) A residual-level deep fusion framework that structurally integrates universal LiDAR processing with tightly coupled LiDAR-inertial optimization is proposed. By jointly and iteratively minimizing LiDAR geometric residuals and IMU preintegration residuals within a unified back-end formulation, the framework achieves deeper cross-modal fusion, thereby enhancing system universality and robustness in complex environments.
4) Comprehensive qualitative and quantitative experimental tests are conducted on both public and self-recorded datasets. The results demonstrate that UD-LIOM exhibits superior robustness and accuracy compared with existing LiDAR SLAM methods.

The remainder of this article is organized as follows. Section II provides a brief overview of related work and analyzes the existing research gaps. Section III details the proposed UD-LIOM framework. Comparative experiments and analysis are presented in Section IV. Finally, Section V summarizes our findings and discusses the future work.

II. Related Work

In this section, we briefly review existing LiDAR odometry and SLAM systems, categorized by LiDAR types and IMU fusion strategies, including mechanical LiDAR-based, solid-state LiDAR-based, and LiDAR–inertial fusion-based approaches.

A. Mechanical LiDAR-Based Methods

In recent decades, mechanical LiDAR has rapidly advanced and become one of the most widely used perception sensors in unmanned intelligence applications, owing to its exceptional measurement precision and reliable performance in complex environments. Numerous LiDAR SLAM solutions have been specifically developed for mechanical LiDAR systems, with LOAM [13] standing out as a seminal approach. LOAM extracts edge and planar features from point clouds and constructs a cost function to estimate the system's pose. Building upon LOAM, several methods have been proposed, with LeGO-LOAM [9] emerging as a lightweight LiDAR SLAM solution widely adopted in practical engineering applications. LeGO-LOAM achieves real-time six degrees of freedom (DoF) pose estimation by segmenting point clouds and utilizing parallel pose estimation techniques based on the segmented data. Similarly, Wang et al. [14] introduced a novel ground segmentation approach and enhanced feature extraction by considering the curvatures of the point clouds. Based on LOAM as well, Shan et al. [15] proposed LIO-SAM, a tightly coupled LiDAR-IMU framework for mechanical LiDARs, based on factor graph optimization. Subsequently, the same research team further developed LVI-SAM [16], a LiDAR-visual-inertial system designed for mechanical LiDARs. This system comprises two tightly coupled subsystems: LIS (extracts LiDAR edge/planar features for scan-matching) and VIS (uses LiDAR depth to optimize visual matching). LIS assists VIS initialization, while VIS provides LIS scan-matching initial guesses. Around the same period, F-LOAM [17] was proposed as a lightweight solution for mechanical LiDAR. It uses a non-iterative two-stage method to correct distortion and weights features by horizontal smoothness to improve matching stability. Later, Wang et al. [18] introduced FD-SLAM, a feature- and distribution-based LiDAR SLAM system designed for mechanical LiDARs. FD-SLAM integrates feature-based fast pose tracking with distribution-based keyframe matching, combining efficiency and accuracy. Recently, Xu et al. [19] developed an intermittent VIO-assisted LiDAR SLAM system that incorporates a degeneracy recognition module and intermittent VIO. In this system, LiDAR supports VIO initialization and depth association, while VIO provides initial scan-matching poses for degenerate LiDAR.

Despite these advancements, these methods rely on the 360-degree scanning pattern of mechanical LiDAR and are not directly applicable to other solid-state LiDARs.

B. Solid-State LiDAR-Based Methods

In recent years, to meet the application requirements of platforms such as unmanned aerial vehicles and embodied intelligence, various compact and low-power solid-state

LiDAR systems have been developed. Concurrently, numerous SLAM methods tailored specifically for these solid-state LiDARs have also emerged. As a pivotal work in the early development of solid-state LiDAR SLAM technology, LOAM-Livox [20] addresses small-FoV and irregular-scanning LiDAR issues (sparsity, mismatch, and blur) via filtering, enhanced feature extraction, and motion compensation. Later, SOTA methods FAST-LIO [21], and its improved version FAST-LIO2 [6] adopted iterative Kalman filtering techniques to accurately estimate pose and are compatible with nonrepetitive scanning solid-state LiDAR, such as Livox AVIA. Afterward, Li et al. [10] proposed a tightly coupled LiDAR-inertial odometry solution that leverages point cloud features and exhibits compatibility with both mechanical LiDARs and Livox AVIA solid-state LiDARs. As a leading solid-state LiDAR, the Livox AVIA has become the core platform for numerous recently proposed SLAM methods. For example, the Livox AVIA-based system SLAM-TSM [22] incorporates total station measurements as constraints within graph optimization, effectively mitigating trajectory drift in GNSS-denied indoor environments and enabling optimization without relying on loop closure. Additionally, Wang et al. [23] introduced a novel tightly coupled SLAM method for the Livox AVIA, combining enhanced VIO with dual-domain residuals and LIO with keyframe static point registration. By fusing these modules with IMU data, the proposed method achieves low end-to-end drift, particularly in degenerate sequences.

Despite the development of numerous methods for solid-state LiDARs [10], [20], [21], [22], [23], these approaches are specifically designed for nonrepetitive scanning patterns (e.g., Livox AVIA and Livox MID-40) and encounter challenges when applied to repetitive scanning solid-state LiDARs (e.g., RSM1).

C. LiDAR-Inertial Fusion-Based Methods

Dynamic environments and high-frequency vibrations present significant challenges to LiDAR-based SLAM methods. Additionally, point cloud distortion caused by the low scanning frequency of LiDAR and high-speed platform motion further limits the performance of these systems. To mitigate these challenges, fusing IMU data for pose estimation has become an effective solution, and various LiDAR-inertial fusion-based SLAM methods have been proposed. These methods can be categorized into loosely coupled and tightly coupled ones. Loosely coupled methods, such as LOAM [13] and LeGO-LOAM [9], process IMU and LiDAR estimations independently, then fuse the results to produce merged outputs. In these approaches, IMU data is not directly involved in the pose estimation process. Recently, Zhang et al. [24] proposed a novel loosely coupled stereo VINS. In this method, IMU independently estimates pose/bias via preintegration and VQF (decoupling accelerometer inclination correction to avoid bias impact), while the stereo visual module independently estimates pose by fusing point-line features. These two modules are fused through dual feedback loops without IMU pose involving visual estimation, ensuring a front-end update rate of over 30 Hz for resource-constrained platforms. In contrast, tightly coupled LiDAR-inertial methods enhance robustness and accuracy by integrating IMU measurements more tightly. LINS [25] is a tightly coupled LiDAR-inertial odometry method designed specifically for ground systems. It leverages high-frequency IMU data to mitigate point cloud distortion and employs iterative error Kalman filtering for joint optimization of LiDAR and IMU data, refining pose estimation. Another notable tightly coupled method is LIO-SAM [15], which incorporates IMU preintegration into the optimization factor graph, contributing to the overall optimization process. Additionally, methods such as R2LIVE [26], FAST-LIO [21], and FAST-LIVO [27] are all tightly coupled LiDAR-inertial fusion-based methods for odometry and mapping, representing significant advancements in the field.

These tightly coupled methods have fully validated the effectiveness of fusing IMU preintegration constraints into back-end optimization, laying a solid foundation for LiDAR-inertial fusion. However, most of them adopt factor-graph parallel fusion (e.g., LIO-SAM) or filter-based serial prediction-update fusion (e.g., FAST-LIO2, LINS) at the module level, lacking residual-level deep integration between IMU and LiDAR constraints. In addition, most of these methods are tailored for specific LiDAR types, and their feature extraction, registration, and fusion strategies lack universality for diverse scanning patterns. When applied to repetitive-pattern solid-state LiDARs [e.g., RSM1, as shown in Fig. 1(b)], measurement errors induced by intrinsic parameter deviations can significantly degrade system performance, and the traditional module-level LiDAR-inertial fusion cannot be effectively utilized to compensate for such deficiencies. Therefore, there is an urgent need to develop a universal SLAM framework that supports multiple LiDAR types (mechanical, nonrepetitive, and repetitive scanning solid-state LiDARs), and systematically integrates residual-level deep LiDAR-inertial fusion with a universal LiDAR processing pipeline. This integration can leverage the tighter coupling of IMU's high-frequency motion residuals and LiDAR geometric residuals to make up for the geometric feature deficiency of diverse LiDARs in complex scenarios, which is a key research gap in the current cross-sensor LiDAR-inertial fusion field.

III. Methodology

A. System Overview

As depicted in Fig. 2, the proposed LiDAR-inertial SLAM framework first performs synchronization of the raw LiDAR-IMU data. Then, the IMU preintegration results are utilized to correct motion distortion in the point clouds, followed by extracting edge and planar points through range smoothness analysis. Subsequently, these features are registered with their corresponding edge/planar submaps to establish geometric constraints. After that, the framework performs joint optimization of LiDAR measurements and IMU preintegration within a multiconstraint optimization model, enabling incremental pose estimation and map construction. To enhance global consistency, sliding-window local optimization is utilized to refine both poses and the map through secondary adjustments. Finally, the system updates the global map based on the

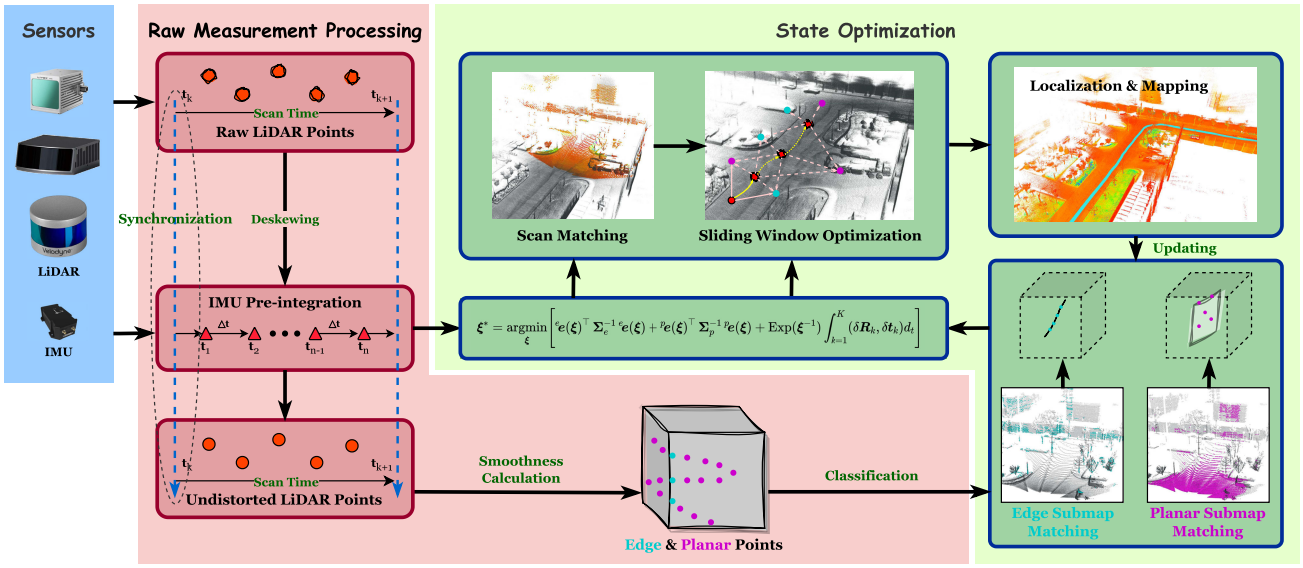

Fig. 2. Overview of the UD-LIOM. Initially, the raw point clouds and IMU data are preprocessed, including IMU preintegration, distortion correction, and temporal synchronization between the LiDAR and IMU data. The undistorted point clouds are then classified into edge and planar points based on range smoothness and registered with their corresponding submaps. Subsequently, LiDAR observation residuals and IMU preintegration are jointly used to build a multiconstraint optimization model for local pose estimation and sliding-window optimization, yielding the system pose and global map. Finally, based on localization and mapping results, the submaps are updated and incorporated into the processing of subsequent inputs.

estimated pose and maintains the submaps to facilitate efficient matching and optimization for subsequent inputs.

B. Raw Measurement Preprocessing

Typically, the LiDAR scan at 10 Hz (yielding raw point clouds every 100 ms), while the IMU operates at 100–200 Hz, providing data at 10–20× the LiDAR frequency. This high-frequency IMU data enables critical functions such as point cloud deskewing and local motion estimation. In this section, we present details of the processing for these raw data, encompassing the IMU preintegration and universal point cloud classification.

1) IMU Preintegration: The IMU provides high-frequency measurements of angular velocity ($\tilde{\boldsymbol{\omega}}^{\mathcal{I}}$) and linear acceleration ($\tilde{\boldsymbol{a}}^{\mathcal{I}}$) in its local coordinate system. These raw measurements are modeled as

$$\tilde{\boldsymbol{\omega}}^{\mathcal{I}} = \boldsymbol{\omega}^{\mathcal{I}} + {}^{\omega}\boldsymbol{b} + {}^{\omega}\boldsymbol{n} \tag{1}$$

$$\tilde{\boldsymbol{a}}^{\mathcal{I}} = \boldsymbol{R}^{\mathcal{I}\mathcal{G}}\left(\boldsymbol{a}^{\mathcal{I}\mathcal{G}} - \boldsymbol{g}^{\mathcal{G}}\right) + {}^{a}\boldsymbol{b} + {}^{a}\boldsymbol{n} \tag{2}$$

where $\boldsymbol{\omega}^{\mathcal{I}}$ represents the IMU angular velocity, and $\boldsymbol{R}^{\mathcal{I}\mathcal{G}}$ is the rotation matrix from the global frame $(\cdot)^{\mathcal{G}}$ to the IMU frame $(\cdot)^{\mathcal{I}}$, $\boldsymbol{a}^{\mathcal{I}\mathcal{G}}$ and $\boldsymbol{g}^{\mathcal{G}}$ denote the IMU acceleration and gravitational acceleration in the global frame, respectively, ${}^{a}\boldsymbol{b}$ and ${}^{\omega}\boldsymbol{b}$ are biases in acceleration and gyroscope measurements, while ${}^{a}\boldsymbol{n}$ and ${}^{\omega}\boldsymbol{n}$ represent the measurement noise for acceleration and gyroscope, respectively.

To correct LiDAR motion distortion and integrate IMU-derived motion increments into pose optimization, the relative rotation and translation between consecutive IMU nodes must be estimated. Among these, the rotational transformation between adjacent IMU nodes can be computed using the angular velocity model [(1)]. For the translation transformation, velocity information over a local time interval is required, as described by the acceleration model [(2)]. Therefore, the velocity ($\Delta\tilde{\boldsymbol{v}}_{ij}^{\mathcal{I}}$), translation ($\Delta\tilde{\boldsymbol{t}}_{ij}^{\mathcal{I}}$), and rotation transformation ($\Delta\tilde{\boldsymbol{R}}_{ij}^{\mathcal{I}}$) between adjacent ith and jth IMU nodes are calculated as

$$\Delta\tilde{\boldsymbol{v}}_{ij}^{\mathcal{I}} = \boldsymbol{R}_i^{\mathcal{I}}\left(\tilde{\boldsymbol{a}}^{\mathcal{I}}\Delta t_{ij}\right) \tag{3}$$

$$\Delta\tilde{\boldsymbol{t}}_{ij}^{\mathcal{I}} = \boldsymbol{R}_i^{\mathcal{I}}\left(\tilde{\boldsymbol{v}}_i^{\mathcal{I}}\Delta t_{ij} + \frac{1}{2}\tilde{\boldsymbol{a}}^{\mathcal{I}}\Delta {t_{ij}}^2\right) \tag{4}$$

$$\Delta\tilde{\boldsymbol{R}}_{ij}^{\mathcal{I}} = \mathrm{Exp}\left(\tilde{\boldsymbol{\omega}}^{\mathcal{I}}\Delta t_{ij}\right) \tag{5}$$

where $\boldsymbol{R}_i^{\mathcal{I}}$ is the rotation matrix of the ith node, and Δt_{ij} is the time interval between the ith and jth IMU nodes.

For LiDAR-IMU fusion, the cumulative transformation between two LiDAR scans is obtained by chaining K intermediate IMU preintegrations. The total rotation $\Delta\boldsymbol{R}_K$ and translation $\Delta\boldsymbol{t}_K$ are defined as

$$\Delta\boldsymbol{R}_K = \prod_{k=1}^{K}\mathrm{Exp}\left(\tilde{\boldsymbol{\omega}}_k^{\mathcal{I}}\cdot\Delta t_k\right) \tag{6}$$

$$\Delta\boldsymbol{t}_K = \sum_{k=1}^{K}\boldsymbol{R}_k^{\mathcal{I}}\left(\tilde{\boldsymbol{v}}_k^{\mathcal{I}}\cdot\Delta t_k + \frac{1}{2}\tilde{\boldsymbol{a}}_k^{\mathcal{I}}\cdot\Delta {t_k}^2\right) \tag{7}$$

where $\tilde{\boldsymbol{\omega}}_k^{\mathcal{I}}$ and $\tilde{\boldsymbol{a}}_k^{\mathcal{I}}$ are the kth IMU measurement within the LiDAR scan interval.

2) Point Cloud Classification: One of the core contributions of this method lies in fundamentally addressing the limited compatibility of existing solutions when dealing with the differentiated scanning patterns of various LiDAR types, especially repetitive scanning solid-state LiDARs (e.g., RSM1). Inspired by the smoothness-based feature metric in LOAM, we reconstruct the feature processing logic and design a universal point classification scheme based on the inherent scanning "ring" structure of all LiDARs—the core difference

from LOAM is not the basic smoothness metric, but the structural optimization of feature extraction logic and the deep integration with a novel classified submap registration mechanism. By analyzing the range smoothness between point $\boldsymbol{p}_i^{\mathcal{L}}$ and its $2N$ neighbors $\{\boldsymbol{p}_{ij}^{\mathcal{L}}\}_{j=1}^{2N}$ within the same scanning "ring" [(8)], we classify the point $\boldsymbol{p}_i^{\mathcal{L}}$ into three categories: edge point, planar point, and invalid point [(9)]. Different from LOAM's segmented sparse feature selection (only 24 feature points per frame for 360° mechanical LiDAR), our dual-threshold strategy abandons complex screening rules (e.g., incident angle filtering and distribution uniformity constraint) and retains all effective edge/planar points after filtering invalid ones. This design realizes dense feature extraction, which maximizes the use of sparse point cloud data from small-FoV solid-state LiDAR and avoids the feature insufficiency problem of LOAM in such scenarios. Moreover, our point classification is not an independent module—it is deeply bound to the classified edge/planar submap registration mechanism (Section III-C1). This closed-loop design completely abandons LOAM's cross-scan-line geometric fitting strategy, thus breaking the dependence on parallel scanning lines and adapting to nonparallel, overlapping scanning rings of repetitive-scanning solid-state LiDAR. Here, the range smoothness of point $\boldsymbol{p}_i^{\mathcal{L}}$ is given by

$$s_i = \frac{1}{N}\left(\sum_{j=1}^{N}\left|r\left(\boldsymbol{p}_i^{\mathcal{L}}\right) - \overleftarrow{r}\left(\boldsymbol{p}_{ij}^{\mathcal{L}}\right)\right| + \sum_{j=1}^{N}\left|r\left(\boldsymbol{p}_i^{\mathcal{L}}\right) - \overrightarrow{r}\left(\boldsymbol{p}_{ij}^{\mathcal{L}}\right)\right|\right) \tag{8}$$

where $r(\cdot)$ denotes the range of the point, while $\overleftarrow{r}(\cdot)$ and $\overrightarrow{r}(\cdot)$ represent the ranges to the rear and front neighboring points of $\boldsymbol{p}_i^{\mathcal{L}}$, respectively. Finally, the classification of $\boldsymbol{p}_i^{\mathcal{L}}$ is detailed as

$$\boldsymbol{p}_i^{\mathcal{L}} \Rightarrow \begin{cases} s_i > \tau_e, & \text{edge point} \\ s_i < \tau_p, & \text{planar point} \end{cases} \tag{9}$$

where τ_e and τ_p are threshold parameters. When the smoothness value s_i falls within the intermediate range ($\tau_p < s_i < \tau_e$), the point $\boldsymbol{p}_i^{\mathcal{L}}$ is classified as invalid and discarded.

To address the practical deployment challenge of parameter adaptation across different LiDAR types, we further clarify the intrinsic correlation between the neighborhood size N, smoothness thresholds τ_e/τ_p, and LiDAR point cloud resolution (i.e., the number of points per degree of FoV), and propose a universal parameter configuration suitable for most scenarios. The neighborhood size N is positively correlated with point cloud resolution: for LiDARs with higher resolution, more valid adjacent points are available within a single scanning ring, and increasing N can obtain more reliable local geometric statistical samples to enhance the stability of smoothness calculation. For a lower resolution with limited valid adjacent points, reducing N can avoid the distortion of smoothness calculation caused by introducing too many nonneighboring points due to an excessively large neighborhood range, ensuring the validity of statistical samples. Both τ_e and τ_p exhibit a positive correlation with point cloud resolution but differ significantly in sensitivity: τ_e shows strong positive correlation and high sensitivity—at high resolution, dense points accurately quantify edge geometric mutations, leading to sharp increases in smoothness values, and an appropriate increase in τ_e filters out planar point noise-induced false mutations; at low resolution, sparse points weaken edge mutation quantization, requiring an appropriate decrease in τ_e to avoid missing real edge points. In contrast, τ_p has weak positive correlation and low sensitivity, as planar points' smoothness values fluctuate around extremely low values regardless of resolution, and minor adjustments to τ_p are sufficient to adapt to different LiDARs due to only slight noise deviations from resolution variations.

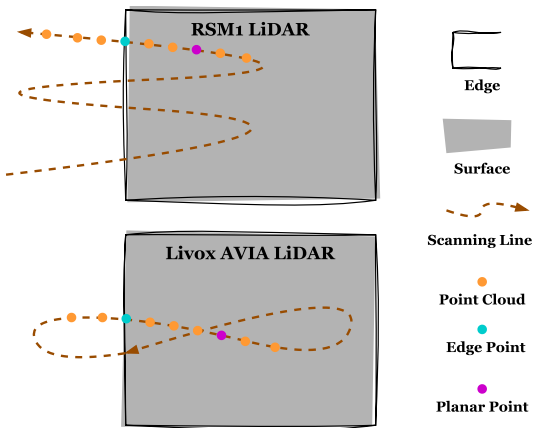

Fig. 3. Illustration of point cloud classification in different scanning patterns of RSM1 and Livox AVIA LiDARs. For each point, neighboring points along the same scanning line are first selected to calculate the corresponding range smoothness value, after which the points are classified based on their smoothness. This classification process is independent of the LiDAR scanning pattern, enabling UD-LIOM to adapt to various types of LiDARs.

Based on systematic verification across all experimental LiDARs (covering mechanical, nonrepetitive scanning solid-state, and repetitive scanning solid-state types) and combined with most LiDARs' point cloud resolution characteristics, the universal parameter configuration is determined as $N = 5$, $\tau_e = 0.5$, and $\tau_p = 0.2$. This configuration meets the feature classification requirements of experimental LiDARs: for high-resolution LiDARs, $\tau_e = 0.5$ distinguishes real edge points from planar point noise, and $\tau_p = 0.2$ shields minor planar point noise; for medium and low-resolution LiDARs, it ensures no missing edge points or misclassified planar points, balancing classification accuracy and practical engineering applicability across various LiDAR hardware.

This point cloud classification strategy assigns a geometric property to each point, obtaining rich point-level features for pose estimation. Compared to traditional feature extraction methods, this approach ensures higher feature density and feature accuracy in complex scenarios, such as environments with sparse texture or occlusions. This advantage is particularly beneficial for solid-state LiDARs, which typically generate a limited number of points per scan. Moreover, unlike other direct methods that uniformly process all points, this approach determines point attributes solely based on neighboring points within the same scanning "ring," effectively avoiding errors caused by internal parameter inaccuracies in repetitive scanning patterns (e.g., RSM1 in Fig. 1). As illustrated in Fig. 3, this classification scheme is independent of the scanning pattern—a key advantage that

enables our system to be compatible with various types of LiDARs.

C. State Optimization

Following the aforementioned IMU preintegration and point cloud classification, this section first elaborates on how to construct a multiconstraint optimization model based on the IMU preintegration results and LiDAR observations to achieve accurate system pose estimation. Subsequently, the construction details of the sliding window and the map update process are introduced.

1) Measurement Model and Cost Function: After the points are processed by the proposed classification scheme, a valid edge point set ($\{{}^e\boldsymbol{p}_i^{\mathcal{L}}\}_{i=1}^n$) and planar point set ($\{{}^p\boldsymbol{p}_j^{\mathcal{L}}\}_{j=1}^m$) can be obtained. To accurately estimate the system's pose and maintain the consistency of the global map, residual errors between these points and the local submap are constructed and minimized. Notably, this study divides the submap into edge submap ${}^e\mathcal{S}$ and planar submap ${}^p\mathcal{S}$, which is a core innovation different from LOAM's cross-scan-line geometric fitting. This design enables efficient parallel dedicated registration of edge/planar points with their corresponding submaps, abandons the dependence on LiDAR scanning line distribution (parallel/nonparallel), and constructs more refined geometric constraints based on different point types.

Specifically, the initial pose for optimization ($\boldsymbol{R}_*^{\mathcal{LG}}, \boldsymbol{t}_*^{\mathcal{LG}}$) for the current system is first determined based on the previously estimated system pose and the latest IMU preintegration results. Subsequently, we perform parallel processing on the edge points $\{{}^e\boldsymbol{p}_i^{\mathcal{L}}\}_{i=1}^n$ and planar points $\{{}^p\boldsymbol{p}_j^{\mathcal{L}}\}_{j=1}^m$. Specifically, each edge point ${}^e\boldsymbol{p}_i^{\mathcal{L}} \in \{{}^e\boldsymbol{p}_i^{\mathcal{L}}\}_{i=1}^n$ and planar point ${}^p\boldsymbol{p}_j^{\mathcal{L}} \in \{{}^p\boldsymbol{p}_j^{\mathcal{L}}\}_{j=1}^n$ are transformed into the global coordinate system based on ($\boldsymbol{R}_*^{\mathcal{LG}}, \boldsymbol{t}_*^{\mathcal{LG}}$) first. Subsequently, the iKD-tree is utilized to retrieve the $K = 5$ nearest neighbors (denoted as $\{\boldsymbol{p}_k^{e\mathcal{S}}\}_{k=1}^K$ and $\{\boldsymbol{p}_k^{p\mathcal{S}}\}_{k=1}^K$) for ${}^e\boldsymbol{p}_i^{\mathcal{L}}$ and ${}^p\boldsymbol{p}_j^{\mathcal{L}}$ from their respective edge submap (${}^e\mathcal{S}$) and planar submap (${}^p\mathcal{S}$). After that, we calculate the centroid ($\bar{\boldsymbol{p}}_i^{e\mathcal{S}}$, $\bar{\boldsymbol{p}}_j^{p\mathcal{S}}$) and covariance matrix (${}^e\boldsymbol{C}_i$, ${}^p\boldsymbol{C}_j$) of $\{\boldsymbol{p}_k^{e\mathcal{S}}\}_{k=1}^K$ and $\{\boldsymbol{p}_k^{p\mathcal{S}}\}_{k=1}^K$, respectively

$$\bar{\boldsymbol{p}}_{i/j}^{e\mathcal{S}/p\mathcal{S}} = \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{p}_k^{e\mathcal{S}/p\mathcal{S}} \tag{10}$$

$${}^{e/p}\boldsymbol{C}_{i/j} = \frac{1}{K}\sum_{k=1}^{K}\left(\boldsymbol{p}_k^{e\mathcal{S}/p\mathcal{S}} - \bar{\boldsymbol{p}}_{i/j}^{e\mathcal{S}/p\mathcal{S}}\right)\left(\boldsymbol{p}_k^{e\mathcal{S}/p\mathcal{S}} - \bar{\boldsymbol{p}}_{i/j}^{e\mathcal{S}/p\mathcal{S}}\right)^{\top} \tag{11}$$

and eigenvalue decomposition is performed on ${}^e\boldsymbol{C}_i$ and ${}^p\boldsymbol{C}_j$, respectively, to obtain their eigenvalues ($\lambda_1 < \lambda_2 < \lambda_3$) and corresponding eigenvectors ($\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3$).

Since local points in ${}^e\mathcal{S}$ usually exhibit a spatial linear structure, it is assumed that the points in $\{\boldsymbol{p}_k^{e\mathcal{S}}\}_{k=1}^K$ are fit to the spatial line L_i. In this case, the eigenvector $\boldsymbol{v}_3$ corresponding to the maximum eigenvalue λ_3 represents the direction vector of L_i. Subsequently, we select two points on L_i: $\boldsymbol{p}_s^{L_i} = \bar{\boldsymbol{p}}_i^{e\mathcal{S}} + \eta\boldsymbol{v}_3$ and $\boldsymbol{p}_e^{L_i} = \bar{\boldsymbol{p}}_i^{e\mathcal{S}} - \eta\boldsymbol{v}_3$, where η is a scaling parameter set to 0.1 in our work. Finally, the distance between the edge point ${}^e\boldsymbol{p}_i^{\mathcal{L}}$ and L_i is defined as follows:

$${}^e e_i = \frac{\left\|\left(\boldsymbol{R}_*^{\mathcal{LG}}\,{}^e\boldsymbol{p}_i^{\mathcal{L}} + \boldsymbol{t}_*^{\mathcal{LG}}\right) - \boldsymbol{p}_s^{L_i}\right) \times \left(\left(\boldsymbol{R}_*^{\mathcal{LG}}\,{}^e\boldsymbol{p}_i^{\mathcal{L}} + \boldsymbol{t}_*^{\mathcal{LG}}\right) - \boldsymbol{p}_e^{L_i}\right)\right\|_2}{\left\|\boldsymbol{p}_s^{L_i} - \boldsymbol{p}_e^{L_i}\right\|_2}. \tag{12}$$

For local points $\{\boldsymbol{p}_k^{p\mathcal{S}}\}_{k=1}^K$ in ${}^p\mathcal{S}$, which usually exhibit a planar structure, it is assumed that these points can be fit to the plane P_j. In this case, the eigenvector $\boldsymbol{v}_1$ corresponding to the minimum eigenvalue λ_1 represents the normal vector of P_j, and it satisfies

$$\left[\left\{\boldsymbol{p}_k^{p\mathcal{S}} - \bar{\boldsymbol{p}}_j^{p\mathcal{S}}\right\}_{k=1}^{K\ \top}\right]\boldsymbol{v}_1 = [\boldsymbol{0}] \in \mathbb{R}^K. \tag{13}$$

Therefore, the distance between the planar point ${}^p\boldsymbol{p}_j^{\mathcal{L}}$ and P_j is defined as

$${}^p e_j = \left|\left(\left(\boldsymbol{R}_*^{\mathcal{LG}}\,{}^p\boldsymbol{p}_j^{\mathcal{L}} + \boldsymbol{t}_*^{\mathcal{LG}}\right) - \bar{\boldsymbol{p}}_j^{p\mathcal{S}}\right)^{\top}\boldsymbol{v}_1\right|. \tag{14}$$

Finally, combining LiDAR observations and IMU preintegration results, the system's pose estimation is transformed into the following nonlinear least-squares form:

$$\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}}{\operatorname{argmin}}\left[{}^e\boldsymbol{e}(\boldsymbol{\xi})^{\top}\boldsymbol{\Sigma}_e^{-1}\,{}^e\boldsymbol{e}(\boldsymbol{\xi}) + {}^p\boldsymbol{e}(\boldsymbol{\xi})^{\top}\boldsymbol{\Sigma}_p^{-1}\,{}^p\boldsymbol{e}(\boldsymbol{\xi}) + \operatorname{Exp}\left(\boldsymbol{\xi}^{-1}\right)\int_{k=1}^{K}(\delta\boldsymbol{R}_k, \delta\boldsymbol{t}_k)\,d_t\right] \tag{15}$$

where ${}^e\boldsymbol{e}(\boldsymbol{\xi}) = [{}^e e_i]_{i=1}^n \in \mathbb{R}^n$ and ${}^p\boldsymbol{e}(\boldsymbol{\xi}) = [{}^p e_j]_{j=1}^m \in \mathbb{R}^m$. The matrices $\boldsymbol{\Sigma}_e^{-1}$ and $\boldsymbol{\Sigma}_p^{-1}$ denote the inverse measurement covariance matrices associated with edge points $\{{}^e\boldsymbol{p}_i^{\mathcal{L}}\}_{i=1}^n$ and planar points $\{{}^p\boldsymbol{p}_j^{\mathcal{L}}\}_{j=1}^m$, respectively. Both covariance matrices are diagonal and adaptively constructed according to feature fitting quality. Specifically,

$$\boldsymbol{\Sigma}_e = \operatorname{diag}\left(\bar{d}_1^e, \ldots, \bar{d}_n^e\right), \quad \boldsymbol{\Sigma}_p = \operatorname{diag}\left(\bar{d}_1^p, \ldots, \bar{d}_m^p\right) \tag{16}$$

where $\bar{d}_i^e$ and $\bar{d}_j^p$ denote the mean distance of the K nearest neighbors [(10)], retrieved via the iKD-tree, to their fit edge line and planar surface, respectively. This adaptive weighting mechanism assigns larger optimization weights to geometrically stable features, thereby improving robustness and convergence stability. The symbol $\boldsymbol{\xi} \in \mathfrak{se}(3)$ is the six-DoF motion parameter vector between adjacent scans, and the conversion process between the $\boldsymbol{\xi}$ and the system transformation matrix is detailed in the Appendix. Additionally, the IMU preintegration ($\int_{k=1}^{K}(\delta\boldsymbol{R}_k, \delta\boldsymbol{t}_k)d_t$) is described in Section III-B1, where K is the number of IMU measurements within the LiDAR scan interval, and d_t is the time interval of IMU measurements.

Different from traditional factor-graph-based parallel fusion or filter-based serial fusion, the proposed multiconstraint optimization framework achieves residual-level deep fusion of LiDAR and IMU measurements. Specifically, LiDAR edge/planar geometric residuals and IMU preintegration residuals are integrated into a unified objective function and jointly optimized within the same nonlinear least-squares formulation at each iteration. This design enables inertial constraints to directly regulate the pose estimation process throughout the optimization, achieving tighter LiDAR-inertial coupling and more comprehensive cross-modal information fusion for a universal LiDAR-inertial system. The details of pose estimation for the UD-LIOM are summarized in Algorithm 1.

Algorithm 1 : Pose Estimation for UD-LIOM

Input: Edge point set ($\{{}^{e}\boldsymbol{p}_i^{\mathcal{L}}\}_{i=1}^{n}$), planar point set ($\{{}^{p}\boldsymbol{p}_j^{\mathcal{L}}\}_{j=1}^{m}$), edge submap (${}^{e}\mathcal{S}$), and planar submap (${}^{p}\mathcal{S}$), IMU preintegration (${}^{\mathcal{I}}\boldsymbol{\xi}$), the estimated pose from the previous scan ($\boldsymbol{\xi}^{\mathcal{P}}$), and the pose transformation between the LiDAR and IMU ($\boldsymbol{\xi}^{\mathcal{LI}}$)

Output: The estimated pose $\boldsymbol{\xi}^{*}$ of the current system **#Parallel**

for each edge point ${}^{e}\boldsymbol{p}_i^{\mathcal{L}}$ in $\{{}^{e}\boldsymbol{p}_i^{\mathcal{L}}\}_{i=1}^{n}$ **do**
  ${}^{e}\boldsymbol{p}_i^{\mathcal{G}} = \boldsymbol{\xi}^{\mathcal{P}} \cdot \boldsymbol{\xi}^{\mathcal{LI}} \cdot {}^{\mathcal{I}}\boldsymbol{\xi} \cdot {}^{e}\boldsymbol{p}_i^{\mathcal{L}}$
  Register ${}^{e}\boldsymbol{p}_i^{\mathcal{G}}$ to the edge submap ${}^{e}\mathcal{S}$ and compute residual term ${}^{e}e_i$
  ${}^{e}E$ += ${}^{e}e_i$
end for
for each planar point ${}^{p}\boldsymbol{p}_j^{\mathcal{L}}$ in $\{{}^{p}\boldsymbol{p}_j^{\mathcal{L}}\}_{j=1}^{m}$ **do**
  ${}^{p}\boldsymbol{p}_j^{\mathcal{G}} = \boldsymbol{\xi}^{\mathcal{P}} \cdot \boldsymbol{\xi}^{\mathcal{LI}} \cdot {}^{\mathcal{I}}\boldsymbol{\xi} \cdot {}^{p}\boldsymbol{p}_j^{\mathcal{L}}$
  Register ${}^{p}\boldsymbol{p}_j^{\mathcal{G}}$ to the planar submap ${}^{p}\mathcal{S}$ and compute residual term ${}^{p}e_j$
  ${}^{p}E$ += ${}^{p}e_j$
  end for#End Parallel
Estimate the optimal pose $\boldsymbol{\xi}^{*}$ by solving (15)
return $\boldsymbol{\xi}^{*}$

2) Jacobian of the LiDAR Measurements: To determine the system's pose by solving (15), the Jacobian matrix of the LiDAR observation residuals is required. Specifically, the Jacobian matrices of the edge point residual ${}^{e}e_i$ and planar point residual ${}^{p}e_j$ [given in (12) and (14), respectively] can be uniformly expressed as follows:

$$
\begin{aligned}
{}^{e/p}\boldsymbol{J}_{i/j} &= \frac{\partial {}^{e/p}e_{i/j}}{\partial \left(r_x, r_y, r_z, t_x, t_y, t_z\right)^{\top}} \\
&= \left(\frac{\partial {}^{e/p}e_{i/j}}{\partial r_x}, \frac{\partial {}^{e/p}e_{i/j}}{\partial r_y}, \frac{\partial {}^{e/p}e_{i/j}}{\partial r_z}, \frac{\partial {}^{e/p}e_{i/j}}{\partial t_x}, \frac{\partial {}^{e/p}e_{i/j}}{\partial t_y}, \frac{\partial {}^{e/p}e_{i/j}}{\partial t_z}\right)^{\top}
\end{aligned} \tag{17}
$$

where (r_x, r_y, r_z) and (t_x, t_y, t_z) represent the rotational parameters and translational parameters of the current scan in the global frame, respectively.

Taking the Jacobian matrix of the point residual error ${}^{e/p}e_{i/j}$ with respect to rotation around the x-axis, $(\partial {}^{e/p}e_{i/j})/(\partial r_x)$, as an example, it is derived as

$$
\begin{aligned}
\frac{\partial {}^{e/p}e_{i/j}}{\partial r_x} &= \frac{\partial {}^{e/p}e_{i/j}}{\partial {}^{e/p}\boldsymbol{p}_{i/j}^{\mathcal{G}\top}} \frac{\partial {}^{e/p}\boldsymbol{p}_{i/j}^{\mathcal{G}}}{\partial r_x} \\
&= \left(d_x, d_y, d_z\right) \left(\frac{\partial \boldsymbol{R}_{*}^{\mathcal{LG}}}{\partial r_x} {}^{e/p}\boldsymbol{p}_{i/j}^{\mathcal{L}} + \frac{\partial \boldsymbol{t}_{*}^{\mathcal{LG}}}{\partial r_x}\right) \\
&= \left(d_x, d_y, d_z\right) \frac{\partial \boldsymbol{R}_{*}^{\mathcal{LG}}}{\partial r_x} {}^{e/p}\boldsymbol{p}_{i/j}^{\mathcal{L}}
\end{aligned} \tag{18}
$$

where (d_x, d_y, d_z) is the gradient direction. For edge point residual error ${}^{e}e_i$, this gradient direction is perpendicular to the fit line L_i

$$
\left(d_x, d_y, d_z\right) = \left(\left({}^{e}\boldsymbol{p}_i^{\mathcal{G}} - \boldsymbol{p}_s^{L_i}\right) \times \left({}^{e}\boldsymbol{p}_i^{\mathcal{G}} - \boldsymbol{p}_e^{L_i}\right)\right) \times \left(\boldsymbol{p}_s^{L_i} - \boldsymbol{p}_e^{L_i}\right). \tag{19}
$$

For planar point ${}^{p}p_j^{\mathcal{L}}$, this gradient direction vector is the normal vector $\boldsymbol{v}_1$ of plane P_j. Here, $\boldsymbol{p}^{\mathcal{G}} = \boldsymbol{R}_{*}^{\mathcal{LG}} \boldsymbol{p}^{\mathcal{L}} + \boldsymbol{t}_{*}^{\mathcal{LG}}$ in (18) and (19). Since the rotation matrix $\boldsymbol{R}_{*}^{\mathcal{LG}}$ is parameterized by Euler angles (r_x, r_y, r_z)

$$
\boldsymbol{R}_{*}^{\mathcal{LG}} = \begin{bmatrix} c_y c_z + s_x s_y s_z & s_x s_y c_z - c_y s_z & s_y c_x \\ c_x s_z & c_x c_z & -s_x \\ s_x c_y s_z - s_y c_z & s_y s_z + s_x c_y c_z & c_x c_y \end{bmatrix} \tag{20}
$$

where (c_x, c_y, c_z) and (s_x, s_y, s_z) denote the cosines and sines of (r_x, r_y, r_z), respectively. Accordingly, the following expression is obtained:

$$
\frac{\partial \boldsymbol{R}_{*}^{\mathcal{LG}}}{\partial r_x} = \begin{bmatrix} c_x s_y s_z & c_x s_y c_z & -s_y s_x \\ -s_x s_z & -s_x c_z & -c_x \\ c_x c_y s_z & c_x c_y c_z & -s_x c_y \end{bmatrix}. \tag{21}
$$

Similarly, the Jacobian matrix of ${}^{e/p}e_{i/j}$ with respect to the translational component t_x can be expressed as

$$
\begin{aligned}
\frac{\partial {}^{e/p}e_{i/j}}{\partial t_x} &= \frac{\partial {}^{e/p}e_{i/j}}{\partial {}^{e/p}\boldsymbol{p}_{i/j}^{\mathcal{G}\top}} \frac{\partial {}^{e/p}\boldsymbol{p}_{i/j}^{\mathcal{G}}}{\partial t_x} \\
&= \left(d_x, d_y, d_z\right) \left(\frac{\partial \boldsymbol{R}_{*}^{\mathcal{LG}}}{\partial t_x} {}^{e/p}\boldsymbol{p}_{i/j}^{\mathcal{L}} + \frac{\partial \boldsymbol{t}_{*}^{\mathcal{LG}}}{\partial t_x}\right) \\
&= \left(d_x, d_y, d_z\right) \frac{\partial \boldsymbol{t}_{*}^{\mathcal{LG}}}{\partial t_x} \\
&= \left(d_x, d_y, d_z\right) (1, 0, 0)^{\top}.
\end{aligned} \tag{22}
$$

Finally, according to (18)–(22), the Jacobian of LiDAR residuals ${}^{e}e_i$ and ${}^{p}e_j$ are obtained.

3) Jacobian of the IMU Measurements: To support the residual-level deep fusion of LiDAR and IMU in the universal processing pipeline, we derive the Jacobian matrix of the IMU residual for solving (15), which ensures the numerical stability and computational efficiency of the LiDAR geometric residuals and IMU preintegration residuals during the iterative optimization process. Specifically, leveraging the IMU preintegration method elaborated in Section III-B1, the pose transformation ${}^{\mathcal{I}}\boldsymbol{\xi} \in \mathfrak{se}(3)$ between two adjacent LiDAR scans is calculated, and the IMU residual term ${}^{\mathcal{I}}\boldsymbol{e}$ is defined by the IMU measurements ${}^{\mathcal{I}}\boldsymbol{\xi}$ and system pose increment $\delta\boldsymbol{\xi} \in \mathfrak{se}(3)$

$$
{}^{\mathcal{I}}\boldsymbol{e} = {}^{\mathcal{I}}\boldsymbol{\xi}^{\top} - \delta\boldsymbol{\xi}^{\top}. \tag{23}
$$

Therefore, the Jacobian matrix ${}^{\mathcal{I}}\boldsymbol{J}$ of the residual term ${}^{\mathcal{I}}\boldsymbol{e}$ is given as

$$
{}^{\mathcal{I}}\boldsymbol{J} = \frac{\partial {}^{\mathcal{I}}\boldsymbol{e}}{\partial \delta\boldsymbol{\xi}} = \frac{\partial \left({}^{\mathcal{I}}\boldsymbol{\xi}^{\top} - \delta\boldsymbol{\xi}^{\top}\right)}{\partial \delta\boldsymbol{\xi}} = -\boldsymbol{I}_{6\times 6}. \tag{24}
$$

4) Sliding Window Optimization: Pose estimation achieved solely through scan-to-submap matching performs poorly in terms of robustness and accuracy, especially for solid-state LiDARs with a limited FoV. To address this issue, we employ a sliding window optimization framework that incorporates new LiDAR pose nodes and IMU nodes as optimization variables while marginalizing older ones, thereby maintaining a fixed window size for pose constraints. Subsequently, the pose transformation relationship between LiDAR measurements and IMU preintegration is utilized to establish pose constraints within the factor graph. This local pose optimization graph iteratively refines each pose estimate, significantly enhancing the pose accuracy of the system.

TABLE I
PUBLIC DATASET DETAILS

Dataset	Environment	Sensor	Platform
KITTI [28]	Urban (Struct)	**LiDAR** Velodyne HDL-64E (10Hz)	Vehicle
BotanicGarden [29]	Natural (Unstruct)	**LiDAR** Velodyne VLP16 (10Hz) Livox AVIA (10Hz) **IMU** Xsens Mti-680G (400Hz)	Ground Robot
M2DGR [30]	Campus (Struct)	**LiDAR** Velodyne VLP-32C (10Hz) **IMU** Handsfree A9 (150Hz)	Ground Robot
NCLT [31]	Campus (Struct)	**LiDAR** Velodyne HDL-32E (10Hz) **IMU** Microstrain GX3 (100Hz)	Ground Robot

5) Mapping Update: Through the above steps, the system's current high-precision pose information is obtained. Timely updating of the system state is crucial for the efficient processing of subsequent point cloud and inertial data. In this method, the local submap is divided into two parts: an edge submap and a planar submap, which, respectively, establish constraints with acquired edge points and planar points. Therefore, it is necessary to update both the edge and planar submaps after obtaining the current system pose. Specifically, based on the global pose of the system, the edge and planar points of the current scan are first incrementally integrated into their corresponding maps. Then, a fixed-size cubic region is constructed in the global coordinate frame, centered at the current system pose. The edge and planar maps within this region serve as the updated edge and planar submaps, providing constraint support for the registration and optimization of subsequent point cloud data. More related details can be found in [6].

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Datasets and Implementation

To evaluate the performance of UD-LIOM across diverse environments and LiDAR types, we conducted comprehensive qualitative and quantitative analyses focusing on robustness, accuracy, and efficiency. Experiments were performed on publicly available datasets—KITTI [28], BotanicGarden [29], M2DGR [30], and NCLT [31]—which cover varied scenarios and include multisensor data. Detailed dataset specifications are provided in Table I. Additionally, to further validate UD-LIOM's robustness and applicability, we also tested it on self-recorded datasets using a Velodyne VLP16 (mechanical LiDAR) and an RSM1 (solid-state LiDAR), alongside a 100 Hz IMU. Our handheld data collection platform is depicted in Fig. 4. Implementation-wise, UD-LIOM was developed in C++ within the ROS (Robot Operating System). Experiments were conducted on a laptop with Ubuntu 18.04, equipped with an Intel Core i7-6820EQ ×8 and 32 GB RAM, without GPU acceleration. For comparative analysis, we evaluated UD-LIOM against SOTA LiDAR SLAM methods: A-LOAM [13], LIO-mapping [32], LINS [25], LeGO-LOAM [9], LIO-SAM [15], LOAM-Livox [20], LiLi-OM [10], and FAST-LIO2 [6].

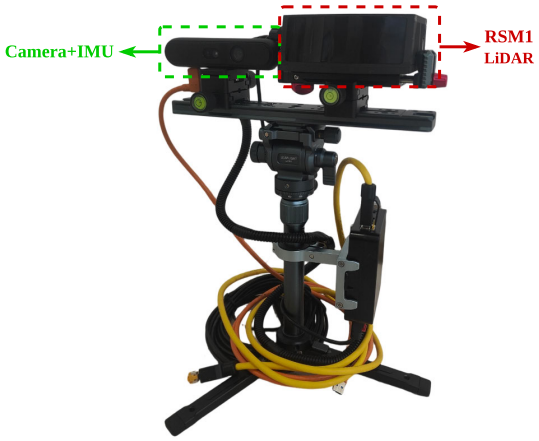

Fig. 4. Our handheld data collection platform.

TABLE II
ADAPTABILITY OF UD-LIOM AND COMPETING METHODS ACROSS VARIOUS LIDAR TYPES

Method	Mechanical LiDAR	Solid-state LiDAR	
		Livox AVIA	RSM1
A-LOAM [13]	✓	✗	✗
LIO-mapping [32]	✓	✗	✗
LINS [25]	✓	✗	✗
LeGO-LOAM [9]	✓	✗	✗
LIO-SAM [15]	✓	✗	✗
LOAM-Livox [33]	✗	✓	✗
LiLi-OM [10]	✓	✓	✗
FAST-LIO2 [6]	✓	✓	✗
UD-LIOM	✓	✓	✓

B. Comparative Analysis of Universality

To evaluate the compatibility of UD-LIOM with LiDARs employing different scanning patterns, this section presents a comparative analysis between UD-LIOM and other SOTA LiDAR SLAM methods, with a particular focus on their adaptability to diverse LiDAR types. As shown in Table II, UD-LIOM exhibits strong compatibility not only with traditional mechanical LiDARs but also with the widely adopted Livox AVIA. Furthermore, UD-LIOM demonstrates effective adaptation to the novel RSM1 solid-state LiDAR. The RSM1 is a lightweight, high-precision solid-state LiDAR characterized by its repetitive scanning pattern and small FoV (as shown in Fig. 1). It has been successfully deployed across various platforms, including automotive applications that require reliable operation under challenging conditions such as high-frequency vibrations and extremely low temperatures. To the best of our knowledge, UD-LIOM is the first published LiDAR SLAM framework verified to be compatible with RSM1, fully demonstrating its exceptional universality across LiDARs with different scanning patterns.

C. Ablation Analysis of Core Modules

The core design of UD-LIOM comprises two synergistic modules: 1) ring-structure-based point cloud classification

TABLE III
COMPARISON OF MEAN APE AND RPE RMSE IN THE KITTI DATASET

Metric	Method	00	01	02	03	04	05	06	07	08	09	10
APE	A-LOAM [13]	1.051	1.873	**1.779**	**1.056**	0.805	**0.784**	0.744	0.964	1.365	**1.591**	1.031
	LeGO-LOAM [9]	1.408	**1.849**	1.960	1.532	**0.555**	1.123	**0.615**	0.758	1.398	1.616	1.403
	LIO-mapping [32]	1.233	2.278	2.067	1.988	1.209	0.925	0.998	0.907	1.602	2.074	1.099
	UD-LIOM	**1.005**	2.250	1.905	1.348	1.549	0.896	0.679	**0.650**	**1.354**	1.721	**0.962**
	w/o Planar Points	2.090	2.299	2.158	1.966	1.939	1.381	0.926	1.810	2.085	2.112	1.116
	w/o Edge Points	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail
RPE	A-LOAM [13]	1.170	1.163	1.155	1.170	1.171	1.173	1.171	1.173	1.168	1.172	1.173
	LeGO-LOAM [9]	1.224	1.550	1.425	1.247	1.109	1.176	1.221	1.105	1.173	1.250	1.125
	LIO-mapping [32]	**0.848**	1.273	0.856	1.072	1.395	0.893	1.266	1.159	1.291	1.226	1.026
	UD-LIOM	0.986	**0.635**	**0.764**	**0.913**	1.018	**0.786**	**1.151**	**1.030**	**0.925**	**0.949**	**0.884**
	w/o Planar Points	0.917	0.647	0.842	0.956	**0.795**	0.925	1.218	1.179	0.968	1.447	0.939
	w/o Edge Points	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail

The best results were in **bold**.

TABLE IV
COMPARISON OF MEAN APE AND RPE RMSE IN THE M2DGR DATASET

Metric	Method	street-01	street-02	street-03	street-04	street-05	street-06	street-07	street-08	street-09	street-10
APE	LINS [25]	46.231	71.016	4.653	108.721	39.668	34.447	65.777	13.649	63.819	48.571
	LIO-SAM [15]	3.587	4.86	4.924	2.836	2.756	**2.781**	15.598	3.257	8.582	32.108
	LiLi-OM [10]	6.995	6.573	3.145	3.661	3.249	2.786	16.053	3.401	21.994	15.799
	FAST-LIO2 [6]	2.759	**3.989**	**2.544**	2.780	2.764	2.786	12.717	**2.750**	4.091	4.785
	UD-LIOM	**2.602**	4.152	2.741	**2.624**	**2.445**	2.811	**12.164**	2.752	**3.173**	**4.061**
	w/o IMU factor	2.803	4.215	2.746	2.804	2.785	2.811	12.378	2.895	4.173	5.061
RPE	LINS [25]	0.194	0.151	0.298	0.193	0.201	0.175	0.934	0.139	0.187	0.183
	LIO-SAM [15]	0.315	**0.117**	0.859	0.362	0.365	0.367	1.539	0.491	0.479	0.493
	LiLi-OM [10]	0.517	0.731	0.923	0.707	0.635	0.635	1.899	0.444	0.594	0.555
	FAST-LIO2 [6]	0.136	0.223	0.254	**0.180**	0.183	0.183	0.940	**0.135**	0.179	0.182
	UD-LIOM	**0.130**	0.223	**0.232**	**0.180**	**0.163**	**0.156**	**0.874**	0.138	**0.112**	**0.179**
	w/o IMU factor	0.138	0.224	0.256	0.181	0.183	0.184	0.941	0.138	0.182	0.183

with submap registration and 2) LiDAR-inertial multi-constraint back-end fusion. To validate the necessity and functional contributions of each core component, two ablation experiments are conducted: feature module ablation (removing edge/planar points) and sensor fusion ablation (removing the IMU back-end fusion factor). Notably, the KITTI dataset—an extensively used benchmark for LiDAR SLAM research—lacks valid high-frequency IMU data, which is insufficient to support reliable IMU-related ablation experiments. Thus, feature module ablation results are derived from Table III (KITTI dataset), while sensor fusion ablation experiments are performed on the M2DGR dataset (equipped with complete high-frequency IMU data), with corresponding results quoted from Table IV. The root-mean-square error (RMSE) of both absolute pose error (APE) and relative pose error (RPE) was computed as metrics, defined as follows:

$$\text{APE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left\|\log\left(\boldsymbol{T}_{gt,i}^{-1}\boldsymbol{T}_{est,i}\right)^{\vee}\right\|_2^2} \tag{25}$$

$$\text{RPE} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N-1}\left\|\log\left(\boldsymbol{T}_{gt,i}^{-1}\boldsymbol{T}_{gt,i+1}\right)^{-1}\left(\boldsymbol{T}_{est,i}^{-1}\boldsymbol{T}_{est,i+1}\right)^{\vee}\right\|_2^2}. \tag{26}$$

For the feature module ablation, the ring-structure-based point cloud classification strategy categorizes valid points into edge and planar points. Experimental results indicate that removing planar points leads to an approximately 29.3% decrease in pose estimation accuracy. Additionally, removing edge points directly causes rotational estimation failure: as widely distributed geometric primitives, edge features are highly sensitive to rotational transformations, and their absence results in severe trajectory drift or even complete localization failure, particularly in sparse point cloud scenarios. This confirms the indispensability of edge-planar point collaborative design. For the sensor fusion ablation, the multi-constraint optimization model enhances robustness in complex motion scenarios by fusing LiDAR geometric residuals with IMU preintegration constraints. Experimental results indicate that removing the IMU back-end fusion factor leads to significant performance degradation: the estimation error increases by approximately 10.7%, and adaptability to challenging scenarios (e.g., rapid rotation and feature-sparse environments) is notably reduced. This degradation is more prominent for repetitive-scanning solid-state LiDARs, as their sparse and inconsistent point clouds rely heavily on high-frequency IMU motion constraints to compensate for geometric feature

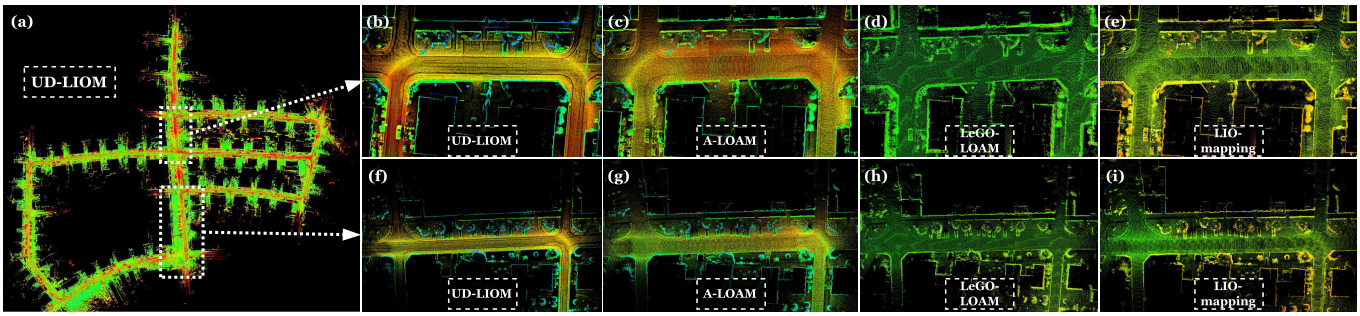

Fig. 5. Mapping result comparison on structured scenes using KITTI-05. (a) Presents the global map reconstructed by UD-LIOM, while (b)–(e) and (f)–(i), respectively, display the mapping results of different methods (UD-LIOM, A-LOAM, LeGO-LOAM, and LIO-mapping) on two distinct road sections in KITTI-05. It can be found that UD-LIOM achieves more refined mapping results, effectively capturing finer details in the environment, such as road surface undulations and roadside features.

deficiencies. Thus, the synergistic optimization of LiDAR and IMU constraints is critical for improving the system's robustness across diverse sensors and scenarios.

In summary, ablation experiments collectively demonstrate that the collaborative design of point cloud classification and LiDAR-inertial multiconstraint fusion is the core guarantee for UD-LIOM to achieve cross-sensor compatibility and high robustness. Sections IV-D and IV-E further verify the framework's practical performance in various complex environments.

D. Mapping in Various Complex Environments

Point cloud classification and the registration between classified points and local submaps provide valuable information for pose optimization. This capability makes UD-LIOM particularly suitable for diverse complex environments. In this section, we validate and analyze the mapping performance of UD-LIOM in both structured and complex unstructured environments to demonstrate its broad applicability across various scenarios.

1) Reconstruction in Structured Environments: We first evaluated UD-LIOM using the KITTI dataset as a benchmark for structured environments, and the results are illustrated in Fig. 5. Specifically, Fig. 5(a) shows the mapping result of UD-LIOM on the KITTI-05 sequence, while Fig. 5(b)–(i) present detailed comparative mapping results from two different road sections, resulting from UD-LIOM, A-LOAM, LeGO-LOAM, and LIO-mapping, respectively. Notably, since the KITTI dataset lacks high-frequency inertial measurement data, our experimental analysis was conducted without IMU measurements. As shown in Fig. 5, UD-LIOM achieves superior accuracy in reconstructing large-scale urban scenes [see Fig. 5(b) and (f)]. This performance advantage primarily arises from the proposed front-endpoint cloud classification and registration framework, which effectively preserves a larger number of geometrically informative points for pose estimation.

In contrast, the feature extraction schemes employed in A-LOAM, LeGO-LOAM, and LIO-mapping are proven inadequate for obtaining sufficient effective features in these scenarios, resulting in degraded performance. Specifically, LeGO-LOAM uses segmented ground points as structural features for pose optimization. However, as shown in Fig. 5(d) and (h), this approach yielded unsatisfactory mapping results. The primary reason lies in the ground point cloud segmentation methodology, which depends critically on the imaging angle between the LiDAR and point clouds for ground identification. System motion often causes misclassification of nonground points as ground points, adversely affecting pose estimation accuracy. Furthermore, the incomplete constraint relationships resulting from this segmentation approach lead to insufficient final accuracy, despite LeGO-LOAM's efficiency gains from decoupling the six-DoF pose estimation into two components.

To further evaluate the map reconstruction performance of UD-LIOM in the structured environments, we conducted comparative experiments on the NCLT dataset against LINS, LIO-SAM, LiLi-OM, and FAST-LIO2. As shown in Fig. 6, as SOTA LiDAR-inertial fusion methods, both LiLi-OM and FAST-LIO2 exhibit consistently impressive performance across all scenes. Furthermore, Fig. 6(p)–(r) reveals that the map reconstruction results of UD-LIOM are comparable to these methods, and even exhibit better performance in certain scenes. For instance, in the second scene (second column), where compact buildings imposed stringent requirements on the method, UD-LIOM outperforms other methods in precisely reconstructing pathways and surrounding vegetation, as visible in the central region of Fig. 6(q). For LiLi-OM, its stringent feature selection criteria limit the number of effective features available in such environments, adversely affecting its reconstruction accuracy [as shown in Fig. 6(j)–(l)]. Similarly, FAST-LIO2's lack of point cloud classification results in uniform processing of each point, leading to blurred scene details and reduced reconstruction clarity [Fig. 6(m)–(o)]. These findings further validate the exceptional map reconstruction performance of UD-LIOM in structured environments and prove the effectiveness of the proposed point classification scheme.

2) Reconstruction in Unstructured Environments: Although existing methods primarily address simple unstructured environments such as forests and grasslands, real off-road scenes exhibit more intricate and unpredictable characteristics. Natural terrain features such as irregular gullies, stone piles, and random shrubs pose significant challenges for conventional LiDAR SLAM systems. The map reconstruction results of

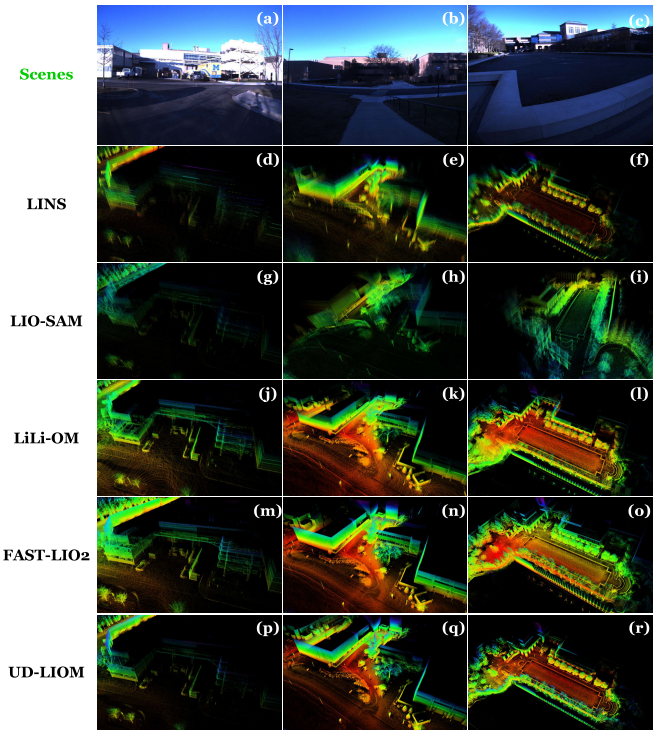

Fig. 6. Mapping result comparison on campus building scenes using NCLT dataset. The first row [(a)–(c)] illustrates three representative scenes in the NCLT dataset, while subsequent rows present the corresponding mapping results obtained by different methods (LINS [(d)-f)], LIO-SAM [(g)-(i)], LiLi- OM [(j)-(l)], FAST-LIO2 [(m)-(o)], and the proposed UD-LIOM [(p)â(r)]). It can be found that UD-LIOM achieves superior reconstruction quality, particularly evident in precisely reconstructing pathways and surrounding vegetation in the second test case (second column).

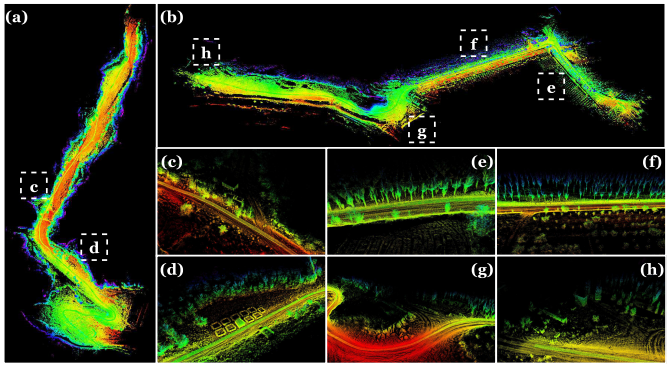

Fig. 7. Mapping results of UD-LIOM in unstructured scenes. (a) and (b) present the complete maps reconstructed by UD-LIOM in two unstructured off-road scenes, respectively. Additionally, (c)–(h) provides detailed views of the labeled regions in (a) and (b). The results demonstrate that UD-LIOM achieves accurate mapping in complex unstructured environments, exhibiting remarkable robustness in complex environments.

UD-LIOM in such challenging environments are depicted in Fig. 7, with Fig. 7(a) and (b) showing the maps reconstructed by UD-LIOM in two distinct off-road scenes. Additionally, Fig. 7(c)–(h) presents detailed representations of various areas in these maps as labeled in Fig. 7(a) and (b), including rugged mountain roads, forests, soil mounds, and interspersed artificial structures. The results indicate that UD-LIOM successfully handles complex off-road environments, accurately reconstructing diverse elements such as forests, slopes, and man-made objects. This robustness from UD-LIOM's innovative point cloud classification scheme and advanced map construction capabilities demonstrates its robustness as a universal LiDAR odometer across various complex environments.

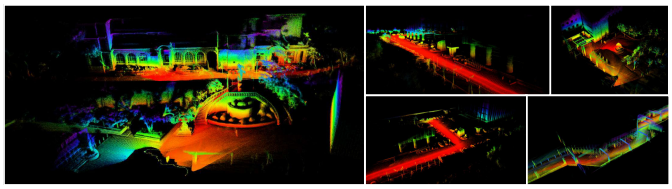

Fig. 8. Mapping results of UD-LIOM using a Livox AVIA LiDAR. The results demonstrate that UD-LIOM achieves robust compatibility with the Livox AVIA solid-state LiDAR.

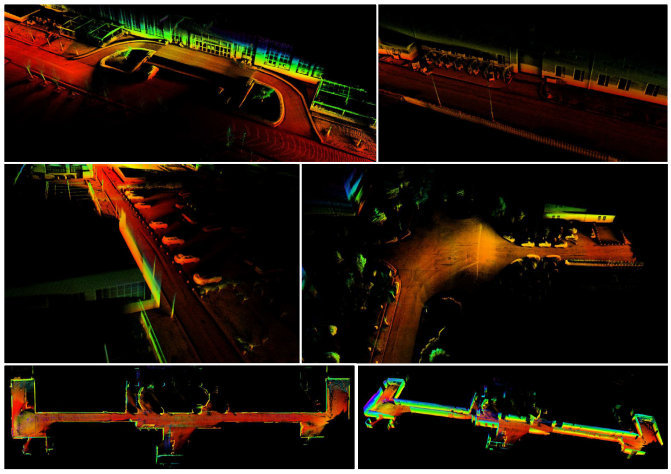

Fig. 9. Mapping results of UD-LIOM using an RSM1 LiDAR. The results demonstrate that UD-LIOM achieves robust compatibility with the RSM1 solid-state LiDAR, showing its compatibility with the repetitive scanning solid-state LiDAR.

E. Reconstruction With Diverse Solid-State LiDARs

This section focuses on evaluating UD-LIOM's compatibility with diverse solid-state LiDARs. Fig. 8 shows the mapping reconstructions in typical scenes using Livox AVIA LiDAR from public datasets. The results demonstrate that UD-LIOM performs robustly with Livox AVIA in complex buildings, open outdoor scenes, and indoor environments. Additionally, Figs. 1(a) and 9 depict the mapping results with RSM1 LiDAR, where Fig. 9 illustrates outdoor and indoor reconstructions from multiple views. Notably, narrow indoor scenes posed significant challenges for solid-state LiDARs due to their limited FoV, which hindered pose optimization convergence under feature-sparse conditions. Nevertheless, UD-LIOM addresses this issue by directly integrating point cloud classification into the optimization framework, yielding superior results. Collectively, these results validate UD-LIOM's adaptability to diverse solid-state LiDARs, particularly its practical applicability with repetitive scanning LiDARs in SLAM—a capability previously overlooked by existing SOTA methods.

F. Localization Results

In this experiment, we conducted a comprehensive quantitative evaluation of UD-LIOM across multiple datasets. Using

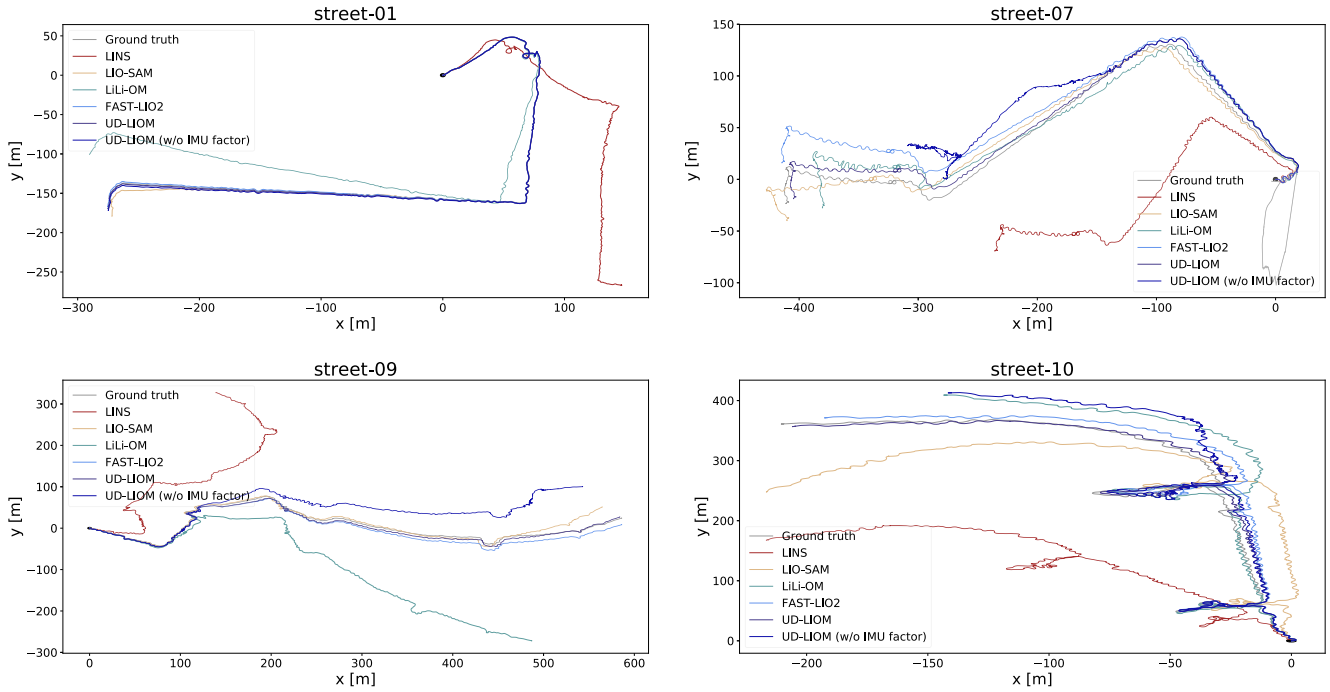

Fig. 10. Comparison of trajectories obtained by different methods on the M2DGR dataset. The results show that UD-LIOM's trajectory aligns most closely with the ground truth. However, its performance deteriorates significantly when IMU factors are excluded, as confirmed by the ablation study. This demonstrates the importance of the proposed IMU fusion strategy in the multiconstraint optimization framework.

the KITTI benchmark, UD-LIOM was compared with several representative LiDAR SLAM methods, including A-LOAM, LeGO-LOAM, and LIO-mapping, and the quantitative results are summarized in Table III. Additionally, to further validate the effectiveness of the proposed point cloud classification strategy, we also conducted ablation experiments that separately evaluated the contributions of edge and planar points; the corresponding results are also presented in Table III. Note that IMU data was excluded in KITTI experiments due to its low frequency, and loop closure was disabled for fairness. As shown in Table III, UD-LIOM consistently achieves superior accuracy across nearly all sequences, with RPE improvements of 24.%, 27.2%, and 18.3% compared to A-LOAM, LeGO-LOAM, and LIO-mapping, respectively.

To further verify the exceptional performance of UD-LIOM in terms of localization accuracy within park-like environments and investigate the influence of the back-end IMU factor on pose optimization, we conducted comparative experiments using LINS, LIO-SAM, LiLi-OM, and FAST-LIO2 as benchmark methods on the M2DGR dataset. The experimental findings are presented in Fig. 10 and Table IV. As illustrated in Fig. 10, UD-LIOM's trajectory shows the closest alignment with ground truth across all sequences, exhibiting strong robustness in rapid rotation scenarios. However, the performance of UD-LIOM significantly decreases when back-end IMU factors are excluded. In fact, although the initial localization results of the sequence are comparable to UD-LIOM, the absence of IMU factor optimization leads to notable deviations in scenarios with rapid rotation, ultimately resulting in substantial cumulative errors. Additionally, the results in Table IV demonstrate that UD-LIOM achieves RPE improvements of approximately 46.0%, 39.7%, 54.1%, and 6.7% compared to LINS, LIO-SAM, LiLi-OM, and FAST-LIO2, respectively. These improvements can be attributed to the point cloud classification scheme proposed in UD-LIOM, which ensures adequate feature points for pose optimization.

TABLE V

RMSE APE FOR DIFFERENT METHODS FOR LIVOX AVIA LIDAR MEASUREMENTS ON THE BOTANICGARDEN DATASET

Seq.	LOAM-Livox [33]	LiLi-OM [10]	FAST-LIO2 [6]	UD-LIOM
1005-00	Fail	Fail	**1.567**	2.496
1005-01	Fail	Fail	**1.639**	2.092
1005-07	Fail	Fail	3.192	**2.532**
1006-01	Fail	Fail	**12.266**	18.280
1008-03	Fail	Fail	2.344	**2.249**
1018-00	Fail	Fail	**1.492**	2.660
1018-13	Fail	Fail	1.387	**1.345**

The BotanicGarden dataset records unstructured scenes within botanical gardens, containing point clouds from both the Velodyne mechanical LiDAR and the Livox AVIA solid-state LiDAR, along with high-frequency IMU measurements and ground truth data. In this experiment, we utilized only the solid-state LiDAR points and IMU measurements from this dataset to conduct comparative experiments with LOAM-Livox, LiLi-OM, FAST-LIO2, and our UD-LIOM. From the results summarized in Table V, it can be seen that the complexity of scenes in the BotanicGarden dataset presents challenges for restrictive feature extraction approaches employed in LOAM-Livox and LiLi-OM, resulting in failures to obtain sufficient numbers of effective features for pose optimization. The point cloud filtering in LOAM-Livox exacerbates the issue of feature scarcity. Notably, FAST-LIO2, currently recognized as a top solid-state LiDAR SLAM method, consistently

TABLE VI
RUNNING TIME [MS] ANALYSIS

LiDAR		Pre-processing	Point Classification			State Optimization	Map Updating
Type	Beam NUM		Time	Edge Point NUM	Planar Point NUM		
Mechanical	16	0.234	**0.368**	16033	6398	26.732	**0.203**
	32	0.936	0.517	22598	7093	55.125	0.440
	64	2.072	1.248	**40931**	**20116**	83.246	0.650
Solid-state	Livox AVIA	0.211	0.487	23941	7412	62.705	1.02
	RSM1	**0.123**	0.373	12076	3767	**19.458**	0.252

demonstrated exceptional performance on this dataset. Our UD-LIOM also achieves competitive results, even outperforming FAST-LIO2 in certain sequences (e.g., 1005-07, 1008-03, and 1018-13) with higher accuracy. The superior performance of both FAST-LIO2 and UD-LIOM can be attributed to their direct use of raw LiDAR points in registration, effectively overcoming the limitations caused by insufficient features in solid-state LiDAR point clouds within complex environments.

G. Running Time Analysis

To evaluate the real-time performance of UD-LIOM, we analyzed the running time and results of each module within the framework, and the findings are presented in Table VI. Furthermore, to test the running time with various scanning beams and LiDAR types, we selected KITTI (with Velodyne HDL-64E LiDAR), BotanicGarden (with Velodyne VLP16 and Livox AVIA LiDAR), and M2DGR (with Velodyne VLP-32C LiDAR) datasets, and a self-recorded dataset (with RSM1 LiDAR) for comparisons. Table VI summarizes the running time of each module in the system with different LiDARs, along with the number of feature points per-scan. Notably, the "Preprocessing" entry in Table VI denotes the time allocated for point cloud and IMU alignment, and the duration required for distortion removal based on IMU preintegration. The results in Table VI show that for mechanical LiDAR, more beams obtain more points, leading to a linear increase in the running time of each module. As for solid-state LiDARs, a Livox AVIA obtains more points than an RSM1 LiDAR, resulting in a longer running time for each module. Overall, the number of points obtained by LiDAR is the primary factor influencing running time. Additionally, in UD-LIOM, we have implemented parallel processing for all modules within the framework. Consequently, the processing time of each scan in UD-LIOM is approximately equal to the maximum running time of the modules, specifically the pose optimization. As shown in Table VI, UD-LIOM maintains a per-scan processing time below 100 ms across all LiDAR types, thereby satisfying real-time performance requirements.

V. CONCLUSION

This article proposed UD-LIOM, a novel framework designed for universal compatibility with both mechanical and various solid-state LiDARs. In the front-end, we introduced a range-smoothness-based universal point cloud classification pipeline via structural reconstruction of traditional feature processing logic (inspired by LOAM). Abandoning sparse feature selection, we retain all effective edge/planar points with a dual-threshold strategy and integrate the classification with an innovative classified submap registration mechanism. This system-level design enables UD-LIOM to adapt to various LiDAR types (including repetitive-scanning solid-state LiDARs such as RSM1) and establish more effective feature correspondences between LiDAR scans and submaps. Consequently, the proposed framework effectively mitigates the performance degradation or failure commonly observed in existing methods, particularly when handling solid-state LiDARs with limited FoV. Additionally, in the back-end, we systematically integrate residual-level deep LiDAR-IMU fusion into the universal LiDAR processing pipeline, fusing IMU preintegration residuals with LiDAR edge/planar geometric residuals into a unified multiconstraint optimization model. Different from traditional module-level fusion, the two types of residuals are jointly optimized on an equal footing at each iteration, realizing tighter LiDAR-inertial coupling. The IMU's high-precision local motion estimation capability is fully utilized to compensate for the geometric feature deficiency of diverse LiDARs, which effectively improves the ability of UD-LIOM to cope with extreme motion and feature-sparse scenarios. To validate our approach, we conducted quantitative and qualitative evaluations, demonstrating UD-LIOM's superior performance across structured/unstructured environments, diverse solid-state LiDAR compatibility, and real-time performance. Future work will focus on integrating visual information to further enhance the system's performance in complex scenes, while proposing a more effective loop closure strategy to balance loop closure recognition and system efficiency.

APPENDIX

In rigid body motion and Lie group theory, the relationship between the transformation matrix $\boldsymbol{T}$ (representing a finite displacement in $SE(3)$) and the six-DoF motion parameter vector $\boldsymbol{\xi}$ [an element of the Lie algebra $\mathfrak{se}(3)$] is governed by the exponential map and logarithmic map. This section formalizes this transformation.

The rigid body transformation matrix $\boldsymbol{T} \in SE(3)$ is a 4×4 homogeneous matrix that combines rotation and translation

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}^\top & 1 \end{bmatrix}$$

where $\boldsymbol{R} \in SO(3)$ is a 3×3 rotation matrix (orthogonal, $\boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{I}$), and $\boldsymbol{t} \in \mathbb{R}^3$ is the translation vector.

The six-DoF motion parameter vector $\boldsymbol{\xi} \in \mathbb{R}^6$ encodes infinitesimal rotation and translation in the Lie algebra $\mathfrak{se}(3)$. It is decomposed as

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{\nu} \end{bmatrix}$$

where $\boldsymbol{\omega} \in \mathbb{R}^3$ represents the rotational component (axis-angle representation, where the direction is the rotation axis and the magnitude is the rotation angle), and $\boldsymbol{\nu} \in \mathbb{R}^3$ represents the translational component (related to the Lie algebra generator).

A. From $\boldsymbol{\xi}$ to $\boldsymbol{T}$: Exponential Map

The exponential map converts the $\boldsymbol{\xi}$ to $\boldsymbol{T}$

$$\boldsymbol{T} = \exp\left(\boldsymbol{\xi}^{\wedge}\right)$$

where $\boldsymbol{\xi}^{\wedge}$ is the wedge operator, that converts the six-DoF vector $\boldsymbol{\xi}$ into a 4×4 matrix

$$\boldsymbol{\xi}^{\wedge} = \begin{bmatrix} \boldsymbol{\omega}^{\wedge} & \boldsymbol{\nu} \\ \mathbf{0}^{\top} & 0 \end{bmatrix}, \quad \boldsymbol{\omega}^{\wedge} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}.$$

The closed-form solution (generalized Rodrigues' formula) is given by

$$\exp\left(\boldsymbol{\xi}^{\wedge}\right) = \begin{bmatrix} \exp\left(\boldsymbol{\omega}^{\wedge}\right) & \boldsymbol{V}\boldsymbol{\nu} \\ \mathbf{0}^{\top} & 1 \end{bmatrix}$$

where $\exp(\boldsymbol{\omega}^{\wedge})$ represents the exponential map for $SO(3)$ (converting axis-angle to a rotation matrix), and $\boldsymbol{V}$ is the translation coupling matrix

$$\boldsymbol{V} = \boldsymbol{I} + \frac{1 - \cos\theta}{\theta^2}\boldsymbol{\omega}^{\wedge} + \frac{\theta - \sin\theta}{\theta^3}\left(\boldsymbol{\omega}^{\wedge}\right)^2$$

where $\theta = |\boldsymbol{\omega}|$ is the magnitude of the angular velocity vector $\boldsymbol{\omega}$.

B. From $\boldsymbol{T}$ to $\boldsymbol{\xi}$: Logarithmic Map

The logarithmic map extracts the motion parameters from $\boldsymbol{T}$

$$\boldsymbol{\xi}^{\wedge} = \log\left(\boldsymbol{T}\right).$$

where the rotational component $\boldsymbol{\omega}$ is obtained via the $SO(3)$ logarithmic map (converting $\boldsymbol{R}$ to axis-angle). The translational component $\boldsymbol{\nu}$ is computed as

$$\boldsymbol{\nu} = \boldsymbol{V}^{-1}\boldsymbol{t}$$

where $\boldsymbol{V}^{-1}$ is the inverse of the coupling matrix.

REFERENCES

[1] Z. Chen et al., "IGE-LIO: Intensity gradient enhanced tightly coupled LiDAR-inertial odometry," *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–11, 2024.

[2] Z. Song et al., "FI-SLAM: Feature information-based robust and efficient LiDAR SLAM," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–12, 2025.

[3] J. Zhu et al., "SPLIN: A structured plane-based LiDAR–inertial SLAM with dynamic object removal," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–15, 2025.

[4] K. Wang, K. Chen, J. Guo, and J. Lu, "Saliency-LOAM: Saliency-based LiDAR odometry and mapping," *IEEE Trans. Instrum. Meas.*, vol. 75, pp. 1–9, 2026.

[5] S. Wang, A. Song, T. Miao, Q. Ji, and H. Li, "A LiDAR SLAM based on clustering features and constraint separation," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–18, 2025.

[6] 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.

[7] S. Liu, J. Huai, B. Wang, Y. Han, and Y. Zhuang, "GBA-LIOM: Global bundle adjustment enhanced LiDAR–inertial odometry and mapping," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–13, 2025.

[8] H. Li, B. Tian, H. Shen, and J. Lu, "An intensity-augmented LiDAR-inertial SLAM for solid-state LiDARs in degenerated environments," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–10, 2022.

[9] 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. (IROS)*, Oct. 2018, pp. 4758–4765.

[10] K. Li, M. Li, and U. D. Hanebeck, "Towards high-performance solid-state-LiDAR-inertial odometry and mapping," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 5167–5174, Jul. 2021.

[11] J. Liu, Y. Zhang, X. Zhao, Z. He, W. Liu, and X. Lv, "Fast and robust LiDAR-inertial odometry by tightly-coupled iterated Kalman smoother and robocentric voxels," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 10, pp. 14486–14496, Oct. 2024.

[12] Z. Wang, L. Zhang, Y. Shen, and Y. Zhou, "D-LIOM: Tightly-coupled direct LiDAR-inertial odometry and mapping," *IEEE Trans. Multimedia*, vol. 25, pp. 3905–3920, 2023.

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

[14] Z. Wang, L. Yang, F. Gao, and L. Wang, "FEVO-LOAM: Feature extraction and vertical optimized LiDAR odometry and mapping," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 12086–12093, Oct. 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 *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 5135–5142.

[16] T. Shan, B. Englot, C. Ratti, and D. Rus, "LVI-SAM: Tightly-coupled LiDAR-visual-inertial odometry via smoothing and mapping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 5692–5698.

[17] H. Wang, C. Wang, C.-L. Chen, and L. Xie, "F-LOAM: Fast LiDAR odometry and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 4390–4396.

[18] J. Wang, M. Xu, G. Zhao, and Z. Chen, "Feature- and distribution-based LiDAR SLAM with generalized feature representation and heuristic nonlinear optimization," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–15, 2023.

[19] J. Xu, T. Li, H. Wang, Z. Wang, T. Bai, and X. Hou, "Intermittent VIO-assisted LiDAR SLAM against degeneracy: Recognition and mitigation," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–13, 2025.

[20] J. Lin and F. Zhang, "Loam livox: A fast, robust, high-precision LiDAR odometry and mapping package for LiDARs of small FoV," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 3126–3131.

[21] W. Xu and F. Zhang, "FAST-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021.

[22] D. Zhang, W. Tan, J. Zelek, L. Ma, and J. Li, "SLAM-TSM: Enhanced indoor LiDAR SLAM with total station measurements for accurate trajectory estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 2, pp. 1743–1753, Feb. 2025.

[23] Y. Wang, S. Fang, K. Shen, and Y. Liu, "A tightly coupled LiDAR-inertial-visual SLAM with enhanced dual subsystems for limited FoV LiDAR," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–12, 2025.

[24] L. Zhang, W. Ye, J. Yan, H. Zhang, J. Betz, and H. Yin, "Loosely coupled stereo VINS based on point-line features tracking with feedback loops," *IEEE Trans. Veh. Technol.*, vol. 73, no. 8, pp. 10916–10931, Aug. 2024.

[25] 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. (ICRA)*, May 2020, pp. 8899–8906.

[26] J. Lin, C. Zheng, W. Xu, and F. Zhang, "R^{2}LIVE: A robust, real-time, LiDAR-inertial-visual tightly-coupled state estimator and mapping," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7469–7476, Jul. 2021.

[27] C. Zheng, Q. Zhu, W. Xu, X. Liu, Q. Guo, and F. Zhang, "FAST-LIVO: Fast and tightly-coupled sparse-direct LiDAR-inertial-visual odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 4003–4009.

[28] 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.*, Jun. 2012, pp. 3354–3361.

[29] Y. Liu et al., "BotanicGarden: A high-quality dataset for robot navigation in unstructured natural environments," *IEEE Robot. Autom. Lett.*, vol. 9, no. 3, pp. 2798–2805, Mar. 2024.

[30] J. Yin, A. Li, T. Li, W. Yu, and D. Zou, "M2DGR: A multi-sensor and multi-scenario SLAM dataset for ground robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2266–2273, Apr. 2022.
[31] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan north campus long-term vision and LiDAR dataset," *Int. J. Robot. Res.*, vol. 35, no. 9, pp. 1023–1035, Aug. 2016.
[32] H. Ye, Y. Chen, and M. Liu, "Tightly coupled 3D LiDAR inertial odometry and mapping," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 3144–3150.

Baosheng Zhang received the B.S. degree in automation from Dalian Minzu University, Dalian, China, in 2019, and the M.S. degree in control science and engineering from Northeastern University, Shenyang, China, in 2022. He is currently pursuing the Ph.D. degree with Tongji University, Shanghai, China.

His research interests include visual and LiDAR SLAM, unmanned intelligent systems, and autonomous driving.

Lin Zhang (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2003 and 2006, respectively, and the Ph.D. degree from the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, in 2011.

From March 2011 to August 2011, he was a Research Associate with the Department of Computing, The Hong Kong Polytechnic University. In August 2011, he joined with the School of Software Engineering, Tongji University, Shanghai. He is currently a Full Professor with the School of Computer Science and Technology, Tongji University. His current research interests include machine vision and unmanned intelligent systems.

Dr. Zhang was awarded as a Young Scholar of Changjiang Scholars Program, Ministry of Education, China. He serves as an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS, and *Journal of Visual Communication and Image Representation*.

Shengjie Zhao (Senior Member, IEEE) received the B.S. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 1988, the M.S. degree in electrical and computer engineering from China Aerospace Institute, Beijing, China, in 1991, and the Ph.D. degree in electrical and computer engineering from Texas AM University, College Station, TX, USA, in 2004.

He is currently a Professor with the School of Computer Science and Technology and the College of Electronics and Information Engineering, Tongji University, Shanghai, China. In previous postings, he conducted research at Lucent Technologies, Whippany, NJ, USA, and China Aerospace Science and Industry Corporation, Beijing. His research interests include artificial intelligence, big data, wireless communications, image processing, and signal processing.

Dr. Zhao is a fellow of the Thousand Talents Program of China and an Academician of the International Eurasian Academy of Sciences.

Yicong Zhou (Senior Member, IEEE) received the B.S. degree in electrical engineering from Hunan University, Changsha, China, in 1992, and the M.S. and Ph.D. degrees in electrical engineering from Tufts University, Medford, MA, USA, in 2008 and 2010, respectively.

He is currently a Full Professor and the Director of the Vision and Image Processing Laboratory, Department of Computer and Information Science, University of Macau, Macau, China. His research interests include chaotic systems, multimedia security, computer vision, and machine learning.

Dr. Zhou was a recipient of the Third Prize of the Macau Natural Science Award in 2014. He serves as an Associate Editor for *Neurocomputing*, *Journal of Visual Communication and Image Representation*, and *Signal Processing: Image Communication*. He is the Co-Chair of the Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He is a Senior Member of the International Society for Optical Engineering (SPIE).