LIVOX-CAM: Adaptive Coarse-to-Fine Visual-assisted LiDAR Odometry for Solid-State LiDAR

Xiaolong Cheng[1], Keke Geng[1], Zhichao Liu[1], Tianxiao Ma[1], and Ye Sun[1]

***Abstract*—The application of solid-state LiDAR is expanding across diverse scenarios. However, most existing methods rely on IMU data fusion to achieve stable performance. This paper presents LIVOX-CAM, a visual-assisted LiDAR odometry based on KISS-ICP, specifically tailored for small field-of-view (FoV) solid-state LiDAR. The system adopts a two-stage architecture comprising a front-end for data pre-processing and a back-end for coarse-to-fine iterative pose optimization. The system is designed to significantly broaden its application scenarios by incorporating a spatial adaptive module and visual assistance. Extensive experiments on public and private datasets show that, even without IMU input, the proposed method achieves robust and accurate performance in challenging scenes, including autonomous driving, degraded scenarios, unstructured environments, and aerial mapping, exhibiting strong competitiveness against state-of-the-art approaches. To encourage reproducibility, the code is available at *https://github.com/huashu996/LIVOX-CAM*.**

***Index Terms*—solid-state LiDAR, localization, odometry.**

I. Introduction

SIMULTANEOUS localization and mapping (SLAM) has been a popular research problem for intelligent agents [1]. After decades of development, there are various SLAM systems based on visible-light camera [2], [3], RGB-D camera [4], LiDAR [5], [6], multi-sensor [7], [8]. The ability of LiDAR to create precise 3D maps makes it a fundamental sensor for mobile intelligent agents. Nevertheless, most existing LiDAR frameworks [9], [10], [5] are tailored for mechanical spinning LiDARs and not suitable for solid-state or non-repetitive scanning LiDARs [11].

Solid-state LiDAR has gained increasing attention due to its advantages in lightweight design, compact form factor, cost-effectiveness, and high-resolution performance, making it a promising solution for a wide range of applications, including autonomous driving [6], aerial robotics [12], and 3D reconstruction [8]. Despite these benefits, adopting novel scanning mechanisms and the inherently narrow field of view (FoV) pose new challenges for LiDAR odometry. Non-repetitive scanning makes most existing point cloud feature extraction algorithms [10], [13] incompatible. Furthermore, the limited FoV significantly reduces the overlap between consecutive scans, exacerbating the risk of drift and degeneration in odometry. Contemporary LiDAR SLAM systems [14], [15] often involve numerous manually adjustable parameters, requiring user intervention to adapt to varying environmental conditions. However, in long-range and long-duration tasks, the surrounding environment usually undergoes significant changes, introducing additional challenges for localization stability. This problem is further exacerbated when using solid-state LiDARs with inherently narrow fields of view, where limited spatial coverage increases sensitivity to scene variation and occlusion. Most solid-state LiDAR odometry [12], [21], [22], [27] uses tightly coupled frameworks that fuse IMU and LiDAR data. Although this helps address narrow FoV and non-repetitive scanning issues, the performance heavily relies on accurate calibration and low-noise IMU data, which are hard to ensure in real-world conditions like vibration, thermal drift, or long-term use [16]. Accordingly, this work aims to develop a solid-state LiDAR odometry method that does not rely on IMU data. The challenges posed by solid-state LiDAR include: (a) reliable point cloud feature extraction under non-repetitive scanning patterns, (b) narrow FoV that increases the risk of degeneration (c) the limited FoV results in fluctuations in the spatial extent of each scan. To address the above problems, we propose a visual-assisted LiDAR

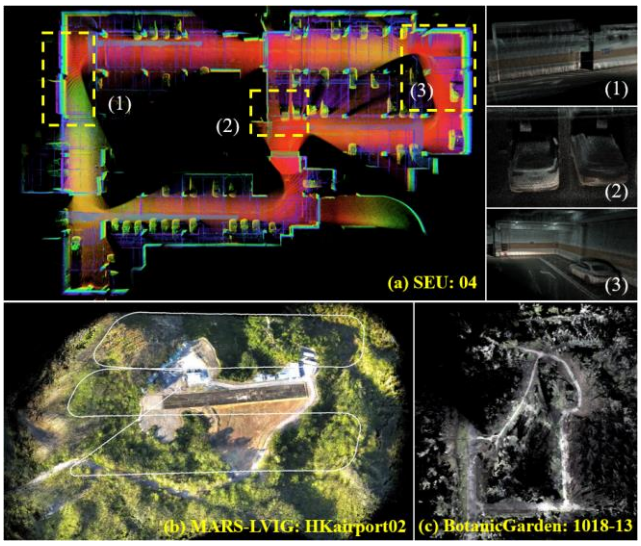

Fig. 1. Some colored map results constructed by LIVOX-CAM. (a) The underground garage scene. (b) The high-altitude mapping scene. (c) The natural garden scene.

Manuscript received: May, 1, 2025; Revised: July, 28, 2025; Accepted: August, 20, 2025.

This paper was recommended for publication by Editor Giuseppe Loianno upon evaluation of the Associate Editor and Reviewers' comments. This work is supported in part by the National Natural Science Foundation of China under Grant no. 52272414 and 51905095, Jiangsu Graduate Innovative Research Program under Grant no. SJCX24_0072, National Key R&D Program of China 2023YFD2000303, and Yangtze River Delta Science and Technology Innovation Alliance Collaborative Research Project 2023C51GG1600. (Corresponding author: Keke Geng.)

[1]The authors are with the Department of Mechanical Engineering, Southeast University, Nanjing 201804, China (e-mail: 230238059@seu.edu.cn; jsgengke @seu.edu.cn; 230248037@seu.edu.cn; 220230347@seu.edu.cn; 220240374@seu.edu.cn).

Digital Object Identifier (DOI): see top of this page.

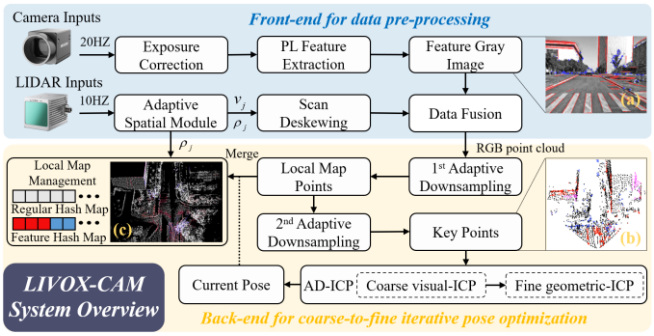

Fig. 2. The system overview of LIVOX-CAM, which contains the data pre-processing (in blue), the iterative position optimization block (in yellow). The v_j and ρ_j are spatial factors of the current scan. (a) Extracting PL features from a gray image. (b) RGB point cloud. (c) Local map incorporating visual features.

odometry method named "LIVOX-CAM", which is built upon and extended from the KISS-ICP framework. The system fully leverages the inherent properties of solid-state sensors, making it highly suitable for complex and variable scenarios.

In summary, the main contributions of this work are summarized as follows: 1) Building upon KISS-ICP [19], we propose LIVOX-CAM, a novel vision-assisted LiDAR odometry framework tailored for solid-state LiDARs, specifically designed to address challenges such as narrow field of view and point cloud degradation; 2) we design an adaptive spatial module that dynamically estimates spatial factors for each frame, enabling the system to handle complex scene changes; 3) we introduce visual features into the odometry pipeline and propose Adaptive Double Iterative Closest Point (AD-ICP), a coarse-to-fine registration strategy that leverages both geometric and visual information to improve pose estimation under degraded LiDAR conditions; 4) we perform extensive experiments on multiple challenging datasets and demonstrate the real-time performance and robustness of LIVOX-CAM. As shown in Fig. 1, the resulting fine-grained texture alignment demonstrates the high localization accuracy achieved by our system.

II. Related Work

A. Mechanical LiDAR Odometry

Early LiDAR odometry systems were primarily developed for mechanical LiDAR sensors, which rely on rotating laser emitters to capture dense 3D point clouds of the surrounding environment. Consequently, many existing methods are closely tied to the scanning characteristics of mechanical LiDAR. The widely known LOAM algorithm [17] contains a point cloud feature extraction method for mechanical LiDAR. This method extracts surface and corner points by calculating the smoothness between each point in the point cloud and its neighboring beam points. Building upon this, LeGO-LOAM [10] introduces ground point segmentation by computing height differences across adjacent scan lines and incorporates ground optimization for improved mapping accuracy. To enhance robustness across diverse environments, Pan et al. [15] proposed a multi-scale, multi-class LiDAR odometry system that achieves stable operation in both indoor and outdoor environments by leveraging a large number of parameters and feature types. In pursuit of higher computational efficiency, F-LOAM [18] introduces a feature weighting strategy during nonlinear optimization, assigning weights based on point smoothness to accelerate pose estimation convergence.

In recent years, LiDAR odometry has seen a growing shift from feature extraction toward direct processing of raw point clouds. Compared to feature-based methods, raw point-based approaches demonstrate stronger adaptability across diverse environments and LiDAR modalities. Dellenbach et al. [9] proposed CT-ICP, a continuous-time framework combined with a sparse voxelized representation, enabling robust and real-time pose estimation. Subsequently, Vizzo et al. [19] proposed KISS-ICP, further exploring the application limits of LiDAR odometry. The KISS-ICP has only 7 parameters compared to other methods that have dozens of parameters. KISS-ICP exhibits strong generalization across UAVs, ground robots, and autonomous vehicles, and is among the few methods explicitly supporting solid-state LiDAR. Nevertheless, its reliance on fixed parameter settings during operation limits its adaptability in dynamically changing environments. This highlights the need for odometry systems that can adjust in real time to scene variations, particularly under the constraints of solid-state LiDAR.

B. Solid-State LiDAR Odometry

Most feature extraction methods are tailored to the scanning patterns of mechanical LiDAR, making them ill-suited for solid-state LiDAR, which typically features a narrow field of view and non-repetitive sampling. Moreover, as the number of manually defined feature types increases, the generalization ability of LiDAR odometry systems tends to decrease, limiting their applicability across varying environments. To address these challenges, Lin et al. proposed Loam_livox [11], a variant of LOAM adapted for solid-state LiDAR. This method filters high-quality points and performs iterative pose optimization. However, its effectiveness is restricted to low-speed motion and requires manual tuning of parameters based on environmental scale and LiDAR types. To overcome the limitations of single-LiDAR systems in high-speed scenarios, Zhang et al. proposed Traj-LO [16], which tightly couples LiDAR-derived geometric features with kinematic constraints on trajectory smoothness. This integration enables the system to achieve performance comparable to LiDAR-inertial odometry, even without inertial sensors.

To address the inherent limitations of solid-state LiDAR, such as narrow FoV and non-uniform sampling, recent research has explored sensor fusion strategies to improve system robustness. The unique characteristics of emerging solid-state LiDARs have inspired new directions in odometry design. Zhu et al. [20] replaced the depth sensor in traditional visual SLAM frameworks with a Livox LiDAR, leveraging its advantages to construct a LiDAR-visual SLAM pipeline. Xu et al. [21] propose the FAST-LIO2, a tightly coupled LiDAR-inertial odometry system based on an iterated Kalman filter framework. It directly aligns raw point clouds to the global map using an efficient incremental k-d tree (ikd-Tree), achieving both accuracy and real-time performance. Building upon this, Zheng et al. [22], [8] introduced the visual-inertial odometry (VIO) system into the FAST-LIO2 framework to further enhance pose estimation stability under degraded LiDAR conditions. These tightly coupled multi-sensor approaches perform well but increase system complexity and computational load, making real-time

TABLE I
ALL PARAMETERS OF OUR APPROACH

	Parameter	Value
Odometry	Expected number of key points n_{key}	1000
	Voxel size of local map v_m	1 m
	Expansion factor for subsampling α	0.3
	Maximum distance $r_{\max}$	100 m
	deviation threshold $\sigma_{\min}$	0.01 m
	ICP convergence criterion γ	10^{-4}
Vision	Line feature length L	30 pixels
	Line feature width W	2 pixels
	ORB feature point count N_{orb}	500
	ORB feature point radius R	2 pixels

deployment on resource-limited platforms challenging.

Leveraging the unique characteristics of solid-state LiDAR, we present a novel visual-assisted LiDAR odometry framework that seamlessly integrates LiDAR and visual information to enhance overall system performance. The proposed LIVOX-CAM addresses longstanding challenges in LiDAR odometry, such as sensitivity to degradation and lack of adaptability, while exploiting the compactness and precision of modern solid-state sensors. LIVOX-CAM can adaptively adjust system parameters according to changing environments, as well as incorporate visual features to overcome the LiDAR degradation problem. Built upon the lightweight KISS-ICP framework [19], our method retains the simplicity and efficiency of point-to-point registration, without relying on auxiliary sub-odometry from visual or inertial systems.

III. PROPOSED SYSTEM

A. System Overview

The overview of the LIVOX-CAM system is illustrated in Fig. 2. The framework comprises two core components: (1) a front-end for data preprocessing, and (2) a back-end for iterative pose optimization. The system first synchronizes LiDAR and camera data. Images are exposure-corrected, and point-line (PL) features are extracted, while the LiDAR point cloud undergoes motion compensation using a constant velocity model. Adaptive spatial factors are then computed to support dynamic parameter adjustment. The extracted visual features are projected onto the point cloud to generate a colorized, fused point cloud. This is followed by a two-stage downsampling to obtain the map and key points. Finally, pose estimation is performed using the proposed Adaptive Double ICP (AD-ICP) algorithm for robust coarse-to-fine registration. System parameters are summarized in Table I, with detailed component descriptions provided in the following sections.

B. Adaptive Spatial Module

The effective scanning space of solid-state LiDAR continuously changes during operation, often causing system parameters to become mismatched with the current scene. For example, using a fixed downsampling size results in insufficient sampling in confined spaces and excessive sampling in large-scale environments. To solve this, we propose an adaptive spatial module that computes spatial factors in real time, allowing dynamic adjustment of key parameters. This ensures sufficient key point density and efficient performance across diverse environments.

We first make the following definitions, P_j denotes the j th frame point cloud, which contains n_j points, $p_i \in P_j$ denotes the i th point of P_j. To determine a suitable voxel size for downsampling, we introduce the expected number of key points as a guiding metric. Taking into account both localization accuracy and computational cost, we set this value to 1000. To extract n_{key} key points, it is necessary to retain approximately n_{key} occupied voxels after downsampling, with each voxel contributing one key point candidate. The expected average number of points per voxel is

$$\rho_{\exp} = n_j / n_{key} . \tag{1}$$

The point cloud is subsequently partitioned into a voxel grid with a resolution of v_m. A voxel is marked as occupied if it contains more than three points. The total number of occupied voxels is N_j, and the average number of points per voxel ρ_j can be obtained. An appropriate downsampling voxel size can be estimated based on the ratio between the spatial volume and the number of key points, which can be expressed as

$$\frac{v_j^3}{v_m^3} = \frac{\rho_{\exp}}{\rho_j}, \qquad v_j = \sqrt[3]{\frac{\rho_{\exp} v_m^3}{\rho_j}} . \tag{2}$$

We define v_j as the scale factor, ρ_j as the resolution factor. They are referred to as spatial factors, which describe the geometric scene scale and the point distribution, respectively. The spatial factors of the current frame are used to dynamically adjust system parameters. Afterwards, the same as in KISS-ICP, the constant velocity model is applied to obtain the deskewed point cloud P_j^*.

C. Visual Feature Extraction and Fusion

For image data processing, we first apply Drago tone mapping followed by histogram averaging to enhance the dynamic range and visual quality, producing a high-resolution image suitable for map visualization. To reduce computational overhead during feature extraction, we downsampled the image to extract both ORB [23] point features and LSD [24] line features from the low-resolution image. We adopt point-line (PL) features due to their broad applicability across diverse environments, including open-field and low-texture scenes. The extraction of visual features is governed by four parameters listed in Table I, which collectively control the number of features, detection thresholds. As shown in Fig. 2(a), the extracted grayscale feature image is fused with the deskewed LiDAR point cloud P_j^* to generate an RGB point cloud P_j^c shown in Fig.2(b). The feature label of points is stored in the RGB channels: line features are marked with R=255, B=0; point features with R=0, B=255; and regular points with R=0, B=0. The G channel encodes reflection intensity. Based on whether a LiDAR point is projected by visual features, the LiDAR points is divided into two categories: visual feature points and regular points.

D. Adaptive Downsampling

Extracting key points for data association is a widely adopted strategy in odometry systems, as it significantly reduces computational complexity while preserving the accuracy required for robust pose estimation. KISS-ICP

employs voxel grid downsampling to extract key points, with each voxel having a predefined size and containing only one representative point, thereby ensuring uniform spatial distribution. The key points obtained in that way cannot be adaptively adjusted, so we propose adaptive downsampling with a scale factor v_j. For each RGB point cloud P_j^c, we first subsample for the first time to get a local map point cloud P_j^{map} for merging with the voxel size αv_j and the second downsampling is to get the key points P_j^{key} with the voxel size v_j. The expansion factor α is introduced to ensure that the map resolution remains higher than that of the key points, which facilitates more effective iterative optimization.

To enhance the system's robustness in geometrically sparse scenes, visual features are projected onto the point cloud. As for each point of the RGB point cloud $p_i^c \in P_j^c$, we determine whether it is the PL feature based on its RGB pixel value. To extract a larger number of visual feature points from limited visual information, a smaller voxel size is adopted to retain more detailed features. The downsampling voxel size is v_j^{pl} for the PL feature point, and v_j^{reg} for the other point, we call it the regular point. The downsampling voxel size can be expressed as

$$v_j^{pl} = \alpha v_j^{reg} = \begin{cases} \alpha v_j, \alpha \in (0.0, 1.0] & 1^{st} \\ v_j & 2^{nd} \end{cases}, \tag{3}$$

where α represents the expansion factor for downsampling, v_j^{reg} denotes the downsampling voxel size of the regular points, v_j^{pl} denotes the downsampling voxel size of the PL feature point. In this manner, the selected key points effectively capture the visual characteristics of the scene, as illustrated in Fig. 2(b). By integrating visual features into the adaptive downsampling strategy, the extracted key points demonstrate enhanced distinctiveness, enabling more robust data association. This visual guidance helps mitigate point cloud degeneration, especially in low-texture or geometrically degraded environments.

E. Local Map Management

The scan-to-map registration is more robust than the scan-to-scan alignment [25]. Therefore, constructing a local map that balances matching accuracy and computational efficiency is critical for ensuring stable and real-time pose estimation. We construct the local map M_{local} using a hash table to efficiently store and manage voxel blocks, where each voxel block serves as a container for storing point cloud data. To enable adaptive spatial representation, we introduce five dynamic parameters that govern the resolution and organization of voxel blocks, allowing the local map to adjust in real time based on environmental complexity and system requirements

$$V_{block}^i \left\{ i, v_m, n_j^{\max}, F, R_{\max} \right\} \quad V_{block}^i \in M_{local} \quad , \tag{4}$$

where the i as an establishment number of the voxel block, v_{map} denotes the grid voxel size of the local map is $v_m \times v_m \times v_m$, $n_j^{\max}$ represents the allowed point number for each voxel to be stored in the j th frame, F is the voxel block feature label that is used to distinguish the feature point, $R_{\max}$ denotes the radius of the local map. For convenient management, the PL feature points $P_{pl} \in P_j^{map}$ and regular points $P_{reg} \in P_j^{map}$ are stored separately in voxel blocks, labeled with F. Due to the sensitivity of visual features to viewpoint and illumination changes, only the 10 frames nearest to the current frame are retained in the feature voxel block to ensure feature consistency and reduce memory consumption. The regular points are removed by distance range $R_{\max}$. The storage points number $n_j^{\max}$ of per voxel block in the local map affects the accuracy of the corresponding estimation. The smaller the space, the denser point cloud, the more points are needed for per voxel block in the local map, as shown in Fig. 3. Therefore, the maximum storage number $n_j^{\max}$ of per voxel block is dynamically adjusted according to the current point cloud resolution ρ_j.

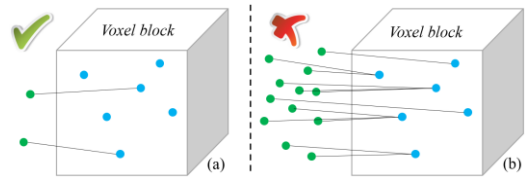

Fig. 3. Correspondence between key points and the local map within a voxel block. In the illustration, green points denote key points, while blue points represent the points within the voxel block. (a) The map resolution is higher than the key points. (b) The map resolution is lower than the key points.

As a result, the size of the local map M_{local} is limited, including feature voxel blocks stored in the latest 10 frames and the regular voxel blocks in the range $R_{\max}$, where the voxel block size is $v_m \times v_m \times v_m$ keep constant and the storage number $n_j^{\max} = 0.5\rho_j$ of per voxel block is dynamic. This design enables our approach to adapt effectively to both large-scale open environments and narrow spaces.

F. Spatial Adaptive Threshold for Data Association

In scan-to-map point cloud search, finding the right target point cloud quickly in the local map is pivotal, with the maximum search distance being a common constraint. KISS-ICP proposes a method to adaptively adjust the threshold by model error σ_j, the detailed formula derivation can be found in the literature [19]. Here, only the modified formulae are explained.

The current model deviation ΔT_j can be obtained from the predicted pose of the constant velocity model and the corrected pose, and the possible offset error distance $\delta(\Delta T_j)$ of the corresponding points could be calculated from Eq. (5)

$$\delta(\Delta T_j) = \delta(\Delta R_j) + \delta(\Delta t_j) . \tag{5}$$

where $\Delta R_j \in SO(3)$ and $\Delta t_j \in \mathbb{R}^3$, $\delta(\Delta R_j)$ is the rotational offset error distance calculated by Eq. (6), $\delta(\Delta t_j)$ is the translation offset error distance calculated by Eq. (7).

$$\delta_{rot}(\Delta R_j) = 2 v_j r_{\max} \sin\left(\frac{1}{2}\arccos\left(\frac{tr(\Delta R_j) - 1}{2}\right)\right), \tag{6}$$

$$\delta_{trans}(\Delta t_j) = \frac{\left\| \Delta t_j \right\|_2}{v_j} . \tag{7}$$

To accommodate the changes in scene scale, the scale factor v_j is incorporated into the offset error, enabling the system to adaptively balance localization accuracy in both large-scale

and confined environments. Notably, a 0.5-meter drift in an open outdoor area may be tolerable, whereas the same magnitude of error in a narrow indoor corridor can lead to significant localization failure. In this way, the offset error $\delta(\Delta T_j)$ for each frame is obtained. And the model error σ_j is calculated from Eq. (8) by superimposing all the offset errors

$$\sigma_j = \sqrt{\frac{1}{N}\sum_{j\in N}\delta(\Delta T_j)^2} \quad , \tag{8}$$

where N denotes the number of model errors higher than the deviation threshold $\sigma_{\min}$. The KISS-ICP used $\tau_j = 3\sigma_j$ as the search boundary, which cannot adapt dynamically to varying scene scales. As a result, the search distance is too large, causing computational consumption, or too small, leading to search failure in scale-varying scenarios. Thus, the scale factor v_j is introduced to enable the search distance to be adjusted with the spatial scale, expressed as

$$\tau_j = \sigma_j v_j \quad . \tag{9}$$

G. Adaptive Double Iterative Optimization

The classic point-to-point ICP has high applicability, but it is sensitive to noise and lacks robustness in the presence of scale variations. To solve this issue, we introduce the spatial factor and the visual feature to the traditional ICP. The adaptive double iterative closest point (AD-ICP) is a coarse-to-fine point cloud registration method. To obtain a robust and accurate pose T_j, twice the correction is used based on the predicted pose $T_{pre,j}$. We first use the vision feature points $P_j^{pl} \in P_j^{key}$ to estimate the visual pose correction $\Delta T_{v,j}$, then use the regular points $P_j^{reg} \in P_j^{key}$ to estimate the ICP pose correction $\Delta T_{g,j}$. The reason for not performing joint iterative optimization is that it causes interference between different key points, leading to increased iteration time. The steps of the AD-ICP are as follows.

The vision feature points P_j^{pl} are first transformed to the global coordinate frame based on the predicted pose $T_{pre,j}$

$$S_j^{pl} = \left\{ s_i = T_{j-1} T_{pre,j}\, p_i \mid p_i \in P_j^{pl} \right\} \quad , \tag{10}$$

where the S_j^{pl} is the source feature points, the T_{j-1} is the last scan pose, p_i is a point of P_j^{pl}. In each iteration k of 1st ICP, each feature source point $s_i \in S_j^{pl}$ is found its corresponding target feature point $q_i \in Q_j^{pl}$ in the local map $M_{local} = \left\{ q_i \mid q_i \in \mathbb{R}^6 \right\}$ with the distance threshold τ_j. The s_i and q_i are 6-dimensional vectors can be expressed as

$$\begin{aligned} s_i &= (x_s, y_s, z_s, r_s, g_s, b_s) \\ q_i &= (x_q, y_q, z_q, r_q, g_q, b_q) \end{aligned} . \tag{11}$$

We define s_i^g, q_i^g as the first three dimensions of the vector, which represent geometric information, and s_i^c, q_i^c as the last three dimensions, which represent color information. Then, we use feature-weighted point-to-point residual sum minimization as the optimization goal

$$\Delta T_{v,k}^{est} = \arg\min_T \sum w_i^c \left(\left\| T s_i^g - q_i^g \right\|_2 \right) , \tag{12}$$

where w_i^c denotes the similarity of feature between each pair of points, can be expressed as

$$w_i^c = \frac{\left\| q_i^c \right\|^2}{\left\| s_i^c - q_i^c \right\|^2 + \left\| q_i^c \right\|^2} , \tag{13}$$

Each iteration, the pose correction $\Delta T_{v,k}^{est}$ is obtained. After the iterative convergence, we get the visually corrected pose $\Delta T_{v,j} = \prod_k \Delta T_{v,k}^{est}$.

Combine the visual corrected pose $\Delta T_{v,j}$ to transform the regular key points P_j^{reg} to the global coordinate frame, and regular source points S_j^{reg} can be expressed as

$$S_j^{reg} = \left\{ s_i = \Delta T_{v,j} T_{j-1} T_{pre,j}\, p_i \mid p_i \in P_j^{reg} \right\} \quad . \tag{14}$$

Then, as in the first iteration, each point $s_i \in S_j^{reg}$ is matched to its corresponding point $q_i \in Q_j^{reg}$. We use geometry-weighted point-to-point residual sum minimization as the optimization goal

$$\Delta T_{g,k}^{est} = \arg\min_T \sum w_i^g \left(\left\| T s_i^g - q_i^g \right\|_2 \right) , \tag{15}$$

where w_i^g denotes the geometric confidence, $\Delta T_{g,k}^{est}$ represents the pose correction at each iteration. To better adapt to multi-scale scenarios, we incorporate spatial factors, and the geometric confidence can be expressed as

$$w_g = \left(\frac{\left\| s_i^g - q_i^g \right\|^2}{\alpha \tau_j + \left\| s_i^g - q_i^g \right\|^2} \right)^2 , \tag{16}$$

After the iterative process converges, we get the geometry correction error $\Delta T_{g,j} = \prod_k \Delta T_{g,k}^{est}$ in 2nd iterative optimization. Finally, the pose deviation ΔT_j at j th frame can be expressed

$$\Delta T_j = (\Delta T_{v,j} T_{j-1} T_{pre,j})^{-1} \Delta T_{g,j} \Delta T_{v,j} T_{j-1} T_{pre,j} \quad . \tag{17}$$

During the iteration process, the correction is smaller than the minimum change threshold γ as a convergence condition. For 1st iterative optimization, the minimum change threshold is 10γ, which uses almost no computational resources while applying visual feature correction. In cases where insufficient visual features are extracted, such as under poor illumination, the system automatically skips the coarse visual matching stage. This design ensures that the algorithm remains robust and insensitive to illumination variations.

IV. EXPERIMENTAL EVALUATION

In this section, we conduct a comprehensive evaluation of the proposed method from the following aspects. Section IV-A introduces the datasets and experimental setup. Section IV-B presents ablation studies to validate the effectiveness of each proposed module. In Section IV-C, we compare our method with several state-of-the-art algorithms, including LiDAR-based methods (KISS-ICP [19], Traj-LO [16] Loam-livox [11], Livox-mapping[1]), LiDAR-inertial method (FAST-LIO2 [21]), LiDAR-inertial-visual method (FAST-LIVO [22]). In Sec. IV-D, the proposed method is further evaluated under unique high-altitude viewpoints to verify its adaptability to unconventional and large-scale observation perspectives. Finally, Section IV-E assesses the real-time performance of the proposed algorithm.

[1] https://github.com/Livox-SDK/livox_mapping

A. Dataset and Setup

Our experiments cover five datasets, including autonomous driving, robotics, handheld, and UAV mapping scenarios, aiming to comprehensively evaluate the applicability and limitations of the proposed method across diverse platforms and environments.

Botanic Garden [26] is a challenging robot navigation dataset collected in a 48,000 m² natural botanic garden, featuring complex unstructured environments such as dense vegetation, narrow trails, riversides, and grasslands. The dataset is collected with stereo cameras, 3D LiDARs, and IMUs.

SEU Dataset is a private collection of eight sequences covering various challenging scenarios, including closed-loop routes, degraded environments, scale variation, and long-range navigation. Data were collected using an AVIA solid-state LiDAR, a HIKVISION color camera, and a GNSS module for ground truth. SEU_04 and SEU_08 represent scale-varying scenes, SEU_02, SEU_05, and SEU_06 contain closed-loop trajectories, while the remaining sequences reflect normal driving conditions.

GEODE [27] is a multi-LiDAR dataset focused on real-world degenerate scenes such as off-road, inland waterways, and metro tunnels, designed to benchmark SLAM performance under feature-poor and ambiguous conditions.

MARS-LVIG [28] is a LiDAR-Visual-Inertial-GNSS dataset, featuring aerial viewpoints captured at altitudes ranging from 80 m to 130 m. The dataset covers a wide range of challenging environments, including airports, islands, rural towns, and valleys.

M3DGR [29] is a sensor-rich benchmark specifically designed to evaluate SLAM systems under diverse scenarios, including visual challenges, LiDAR degeneration, wheel slippage, and GNSS denial. The data collection platform is equipped with RGB-D cameras, 3D LiDAR, GNSS, IMU, and wheel odometry sensors, providing rich multi-modal data for robust SLAM evaluation.

In the Botanic Garden, SEU, GEODE datasets, all methods are evaluated using identical parameter settings to assess their robustness and sensitivity to different scenes. The parameter settings of our system are summarized in the Table. I. For quantitative analysis, we use the evo toolkit [30] to obtain the Root Mean Square Error (RMSE) of Absolute Trajectory Error (ATE) and Relative Trajectory Error (RTE) with a window size of 1 m.

B. Experiment-1: Ablation Study

To demonstrate the effectiveness of visual assistance and the adaptive spatial module, we conduct ablation studies. The baseline uses a fixed downsampling size without visual input. Then, the adaptive spatial module (A) and visual assistance (V) are individually incorporated to evaluate their respective contributions. To enhance the credibility of the experimental results, we evaluate our method on three distinct datasets with varying scene characteristics. The quantitative results are summarized in Table II, while the corresponding trajectories are illustrated in Fig. 4.

SEU 04 (underground garage), M3DGR Corridor01 (indoor corridor), and GEODE Tunnel5 (metro tunnel) represent environments with varying degrees of point cloud degradation, primarily caused by confined structures and low geometric diversity. When the LiDAR gets close to planar walls, the system is prone to failure, exhibiting noticeable trajectory drift. This is largely due to the use of a fixed downsampling voxel size, which results in sparse point clouds with limited structural information, impairing accurate alignment. By contrast, the proposed adaptive downsampling strategy alleviates this issue by preserving more informative points. Even with a sufficient number of points, such drift is common in LiDAR odometry when operating in environments with low geometric variability, such as smooth or featureless surfaces. With the integration of visual constraints alone, the system may still fail in degraded scenes due to an insufficient number of points. Only when the spatial adaptive module is combined with visual assistance can these challenging scenarios be effectively overcome, as illustrated in Fig. 4. To evaluate the sensitivity of the coarse-to-fine iterative strategy to illumination changes, we conduct experiments in a dark environment sequence M3DGR Dark01. As shown in Table II, the performance degradation is minimal, indicating strong robustness to low-light conditions.

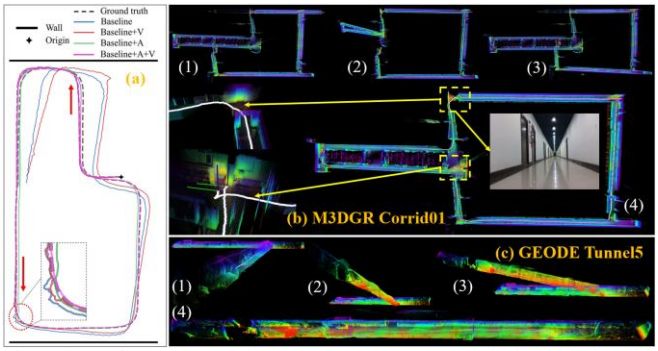

Fig. 4. (a) Trajectory comparison on SEU_04 sequence. The bold black solid lines indicate walls, and the red arrows indicate the moving direction. (b) The point cloud maps on M3DGR Corrid01 sequence. (c) The point cloud maps on the GEODE Tunnel5 sequence. Subfigures (1), (2), (3), and (4) respectively illustrate the results of Baseline, Baseline+V, Baseline+A, and Baseline+AV methods.

TABLE II
THE ATE/RTE OF OUR METHOD ON ABLATION EXPERIMENT.

Method	SEU 04		M3DGR Corrid01		M3DGR Dark01		GEODE Tunnel5	
	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE
Baseline	4.531	0.089	20.819	29.443	0.185	0.110	24.605	0.226
+V	3.962	0.098	17.795	25.166	0.189	0.113	31.816	0.227
+A	0.879	0.043	11.043	15.61	0.169	0.110	4.455	0.047
+AV	0.380	0.042	1.119	1.582	0.172	0.108	0.373	0.046

Ablation studies confirm that the integration of the adaptive spatial module and visual assistance significantly enhances system robustness, especially in degraded or challenging environments. Moreover, our system is less affected by illumination changes, further improving its reliability across diverse scenarios.

C. Experiment-2: Evaluation of Localization Accuracy

In this section, we compare our method against state-of-the-art odometry algorithms on 3 datasets representing different environments, including indoor confined spaces, urban outdoor areas, natural unstructured scenes, and degraded scenes. The comparison results are summarized in Table III, where our method achieves competitive performance across all evaluated scenarios. FAST-LIO, FAST-LIVO, and Traj-LO achieve lower Relative Trajectory Error (RTE) compared to our method. This is because using IMU and continuous-time models

TABLE III
THE COMPARISON OF ABSOLUTE/RELATIVE TRAJECTORY ERRORS (ATE/RTE, METERS). THE -- INDICATES METHOD FAIL IN THIS SEQUENCE.

Dataset	Sequence	NO IMU										USE IMU			
		Ours		KISS-ICP [19]		Traj-LO [16]		Loam-livox [11]		Livox-Mapping[1]		FAST-LIO2 [21]		FAST-LIVO [22]	
		RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE
Botanic Garden (Robot)	1005-00	0.047	2.26	0.058	4.70	0.022	2.41	0.061	5.47	0.185	23.65	0.018	2.305	0.024	2.54
	1005-01	0.045	2.15	0.066	2.82	0.018	2.04	0.067	13.61	0.144	3.81	0.018	2.470	0.026	1.46
	1005-07	0.051	5.88	0.098	14.52	0.026	3.39	0.074	14.02	0.162	24.67	0.023	4.438	0.027	3.23
	1006-01	0.050	5.77	0.073	9.04	0.028	3.46	0.093	26.71	0.071	11.75	0.065	39.733	0.028	9.54
	1008-03	0.059	3.31	0.089	7.41	0.025	3.41	0.095	11.83	0.095	19.77	0.024	4.019	0.042	6.14
	1018-00	0.046	0.54	0.058	0.88	0.021	0.47	0.056	0.71	0.062	0.73	0.027	2.154	0.043	0.66
	1018-13	0.042	1.01	0.065	1.31	0.018	1.11	0.073	8.11	0.074	2.37	0.025	2.390	0.028	1.28
SEU (Vehicle)	01	0.052	1.07	0.056	1.12	0.023	4.99	7.463	43.47	0.148	77.52	0.020	4.83	0.036	18.86
	02	0.059	0.64	0.064	0.61	0.072	1.13	8.397	27.75	0.097	18.26	0.037	0.94	0.054	3.54
	03	0.051	1.72	0.061	2.69	0.025	3.17	0.413	8.78	0.855	34.07	0.024	5.28	0.046	9.96
	04	0.035	0.32	0.293	24.97	0.028	0.81	0.958	4.51	0.025	0.65	0.012	0.29	0.024	1.82
	05	0.048	1.08	0.049	1.61	0.011	0.48	0.828	11.99	0.117	40.49	0.019	3.06	0.019	2.54
	06	0.059	1.14	0.065	1.69	0.019	0.39	6.027	40.88	13.983	102.73	0.022	2.55	0.024	2.10
	07	0.065	2.89	0.053	4.42	0.021	3.94	--	--	--	--	0.021	7.79	--	--
	08	0.039	0.71	0.227	53.14	0.026	1.05	0.792	4.95	0.028	2.35	0.011	0.21	0.023	7.83
GEODE (Handle)	Tunnel1	0.045	0.56	0.046	0.35	0.014	0.14	2.65	30.20	4.53	1.60	0.032	0.23	0.045	0.40
	Tunnel2	0.047	1.04	0.051	1.79	0.049	1.55	2.98	23.53	2.47	1.32	0.025	1.88	0.024	1.71
	Tunnel3	0.057	0.56	0.054	6.12	0.078	22.0	2.91	16.98	2.44	3.57	0.035	0.51	0.066	3.99
	Tunnel4	0.043	0.32	0.081	0.33	0.046	0.31	2.88	18.50	3.68	0.70	0.038	0.35	0.021	0.35
	Tunnel5	0.046	0.37	0.073	5.79	0.028	0.67	2.61	7.57	2.27	2.26	0.022	0.43	0.042	9.37
Avg		0.049	1.66	0.084	7.26	0.029	2.84	2.07	16.82	1.65	19.59	0.025	4.29	0.033	4.59

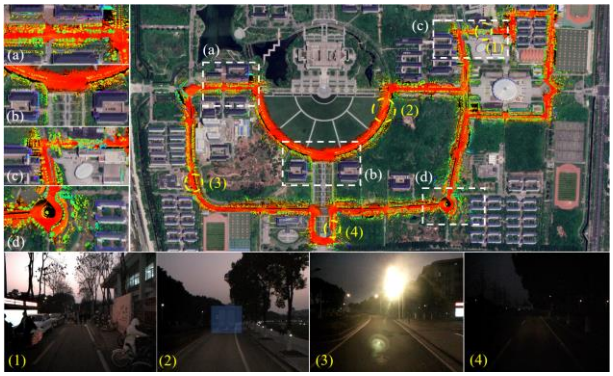

Fig. 5. Project point cloud map onto satellite image. The zoom-in images at the left reflect the mapping accuracy and the pictures at the bottom reflect the scene situation. (1) Dynamic object scene. (2) Follow the van scene. (3) Dazzling scene. (4) Low recognition scene.

enable more accurate motion compensation for LiDAR point clouds, while both filtering-based frameworks (FAST-LIO, FAST-LIVO) and the continuous-time model (Traj-LO) further incorporate pose constraints from historical frames, leading to smoother trajectories. The performance of methods Loam-livox and Livox-mapping is highly sensitive to parameter settings under varying environments, resulting in notable instability across different scenarios.

In the Garden Botanic dataset, PL features enhance the robustness of our method, allowing it to operate reliably in unstructured and feature-sparse scenarios. In the 1006-01 sequence, strong platform vibrations caused by uneven surface transitions lead to degraded IMU signals, resulting in significant Z-axis drift for IMU-dependent methods like FAST-LIO and FAST-LIVO.

KISS-ICP fails on SEU_04 and SEU_08 due to its fixed voxel size, which limits adaptability in scale-varying scenes and reduces accuracy. SEU_07 covers 4.627 km with dynamic and low-visibility conditions, causing instability in many methods. In such scenes, visual odometry is unreliable due to occlusions and moving objects. As a result, FAST-LIVO fails on SEU_07, which includes a long vehicle-following segment, with the failure point marked in Fig. 5(2). To evaluate global consistency, we align the LIVOX-CAM map with the satellite image in Fig. 5.

TABLE IV
THE COMPARISON OF ABSOLUTE TRAJECTORY ERRORS (ATE, METERS) ON MARS-LVIG DATASET. THE -- INDICATES THAT THE METHOD FAILED.

Sequence	Ours	KISS-ICP [19]	Traj-LO [16]	FAST-LIO2 [21]
HKairport01	3.85	10.63	9.12	0.66
HKairport02	8.84	18.16	17.95	1.14
HKisland01	0.61	1.021	0.35	0.64
HKisland02	4.28	--	1.94	2.13
AMtown01	3.24	12.59	2.46	2.28
AMtown02	10.59	--	4.30	3.24
AMvalley01	11.18	--	--	4.54
AMvalley02	42.86	--	--	8.12

On the GEODE degradation dataset, characterized by frequent LiDAR degeneration and challenging lighting conditions, our method maintains stable performance across all sequences. This robustness stems from the mutual complementarity between visual and geometric features. Visual cues compensate for LiDAR degeneration in texture-rich regions, while geometric information reinforces alignment in areas where visual data is unreliable. The coarse-to-fine registration strategy further enhances system stability in challenging environments. Although FAST-LIVO adopts a tightly coupled LiDAR-inertial-visual framework, its performance is highly sensitive to illumination changes. Under severe lighting variations, the visual feature becomes unreliable, causing FAST-LIVO to perform worse than FAST-LIO2.

D. Experiment-3: Evaluation on High-Altitude Scene

To further explore the limits of our system, high-altitude mapping scenarios are introduced. The MARS-LVIG dataset has unique challenges for LiDAR odometry and often renders many feature-based algorithms ineffective. Hence, we selected representative methods for comparison, with results shown in Table IV. The high-speed flight and turbulence of the UAV in the MARS-LVIG dataset bring significant challenges to the constant-velocity motion model.

The results show that our method is the only one that did not fail on any sequence without using IMU data. Traj-LO divides each LiDAR frame into multiple segments and incorporates trajectory smoothing constraints, achieving good performance on sequences without degeneration. In the AMvalley01 and AMvalley02 sequences, the UAV gradually approaches the mountain, causing a rapid reduction in the

TABLE V
PERFORMANCE EVALUATION

	SEU	Botanic Garden	GEODE	MARS-LVIG
Time (ms)	69.4	70.8	57.6	93.4
Memory (MB)	552	497	256	833

scanning area. This leads to point cloud degeneration and results in the failure of methods KISS-ICP and Traj-LO. In turning regions with abrupt UAV motion, the constant-velocity model causes larger pose errors. At high altitudes, these errors are amplified due to long-range projection, reducing scan-to-map overlap. While our method may not surpass tightly coupled systems like FAST-LIO2 in such scenarios, it achieves significant improvements over pure LiDAR odometry methods. A colored map of this dataset constructed by our algorithm is shown in Fig. 1(b).

E. Runtime Analysis

Our algorithm can maintain real-time performance that meets the 10 Hz LiDAR input frequency, running on a system equipped with an Intel Core i7-14700K CPU and 64 GB RAM, without requiring GPU acceleration. The mean processing time and maximum usage memory on different datasets are presented in Table V.

V. CONCLUSION

This paper introduces LIVOX-CAM, a vision-assisted LiDAR odometry framework designed for solid-state LiDARs. The system enhances robustness by integrating an adaptive spatial module and visual assistance, effectively mitigating the impact of LiDAR degradation in challenging environments. Extensive experimental results validate that our system maintains robustness across challenging scenarios, including scale-varying scenes, degenerate scenes, low-light scenes, unstructured scenes, and high-altitude mapping scenes. However, in high-speed and vibration-intensive scenarios, our system performs less favorably compared to continuous time models, due to the limitations of the constant-velocity assumption. Future work will explore adaptive motion models and visual loop closures to improve global consistency and long-term robustness.

REFERENCES

[1] P.-Y. Lajoie and G. Beltrame, "Swarm-SLAM: Sparse decentralized collaborative simultaneous localization and mapping framework for multi-robot systems," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 475–482, 2024.

[2] S. Boche, X. Zuo, S. Schaefer, and S. Leutenegger, "Visual-inertial SLAM with tightly-coupled dropout-tolerant GPS fusion," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2022, pp. 7020–7027.

[3] C. Campos, R. Elvira, J. J. G. Rodriguez, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, 2021.

[4] J. Liu, X. Li, Y. Liu, and H. Chen, "RGB-D inertial odometry for a resource-restricted robot in dynamic environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9573–9580, 2022.

[5] S. Yi, Y. Lyu, L. Hua, Q. Pan, and C. Zhao, "Light-LOAM: A lightweight LiDAR odometry and mapping based on graph-matching," *IEEE Robotics and Automation Letters*, vol. 9, no. 4, pp. 3219–3226, 2024.

[6] X. Cheng, T. Ma, K. Geng, Z. Liu, Z. Wang, and G. Yin, "SVM-LO: An accurate, robust, real-time LiDAR odometry with segmentation voxel map for autonomous vehicles," in *Proc. IEEE Int. Conf. on Intelligent Transportation Systems (ITSC)*, 2024, pp. 1870–1877.

[7] J. Lin and F. Zhang, "R(3)LIVE++: A robust, real-time, radiance reconstruction package with a tightly-coupled LiDAR–inertial–visual state estimator," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 11168–11185, Dec. 2024.

[8] C. Zheng et al., "FAST-LIVO2: Fast, direct LiDAR–inertial–visual odometry," *IEEE Trans. Robot.*, vol. 41, pp. 326–346, 2025.

[9] P. Dellenbach, J.-E. Deschaud, B. Jacquet, and F. Goulette, "CT-ICP: Real-time elastic LiDAR odometry with loop closure," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2022, pp. 5580–5586.

[10] T. X. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized LiDAR odometry and mapping on variable terrain," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2018, pp. 4758–4765.

[11] J. R. 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. on Robotics and Automation (ICRA)*, 2020, pp. 3126–3131.

[12] W. X. Dongjiao He, N. Chen, F. Kong, C. Yuan, and F. Zhang, "Point-LIO: Robust high-bandwidth light detection and ranging inertial odometry," *Adv. Intell. Syst.*, vol. 5, no. 1, p. 2200459, 2023.

[13] K. Honda, K. Koide, M. Yokozuka, S. Oishi, and A. Banno, "Generalized LOAM: LiDAR odometry estimation with trainable local geometric features," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12459–12466, 2022.

[14] M. Yokozuka, S. Oishi, and A. Banno, "LiTAMIN2: Ultra light LiDAR-based SLAM using geometric approximation applied with KL-divergence," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[15] Y. Pan, Y. He, Z. Shao, and Z. Li, "MULLS: Versatile LiDAR SLAM via multi-metric linear least square," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2021, pp. 11633–11640.

[16] X. Zheng and J. Zhu, "Traj-LO: In defense of LiDAR-only odometry using an effective continuous-time trajectory," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1961–1968, 2024.

[17] J. Zhang and S. Singh, "Low-drift and real-time LiDAR odometry and mapping," *Autonomous Robots*, vol. 41, no. 2, pp. 401–416, 2017.

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

[19] G. T. Vizzo, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "KISS-ICP: In defense of point-to-point ICP – simple, accurate, and robust registration if done the right way," *IEEE Robotics and Automation Letters*, vol. 8, pp. 1029–1036, 2023.

[20] Y. Zhu, C. Zheng, C. Yuan, X. Huang, and X. Hong, "CamVox: A low-cost and accurate LiDAR-assisted visual SLAM system," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2021, pp. 5049–5055.

[21] 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, 2022.

[22] C. Zheng et al., "FAST-LIVO: Fast and tightly-coupled sparse-direct LiDAR-inertial-visual odometry," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2022, pp. 4003–4009.

[23] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. on Computer Vision (ICCV)*, 2011, pp. 2564–2571.

[24] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, Apr. 2010.

[25] J. Behley and C. Stachniss, "Efficient surfel-based SLAM using 3D laser range data in urban environments," in *Robotics: Science and Systems (RSS)*, 2018.

[26] Y. Liu et al., "BotanicGarden: A high-quality dataset for robot navigation in unstructured natural environments," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2798–2805, 2024.

[27] Z. Chen, Y. Qi, D. Feng, et al., "Heterogeneous LiDAR dataset for benchmarking robust localization in diverse degenerate scenarios," *Int. J. Robot. Res.*, 2024.

[28] H. Li et al., "MARS-LVIG dataset: A multi-sensor aerial robots SLAM dataset for LiDAR–visual–inertial–GNSS fusion," *Int. J. Robot. Res.*, vol. 43, no. 8, pp. 1114–1127, 2024.

[29] D. Zhang, J. Zhang, Y. Sun, et al., "Towards robust sensor-fusion ground SLAM: A comprehensive benchmark and a resilient framework," *arXiv preprint*, arXiv:2507.08364, 2025.

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