IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Azimuth-LIO: Robust LiDAR-Inertial Odometry Via Azimuth-Aware Voxelization and Probabilistic Fusion

Author Names and Affiliations Omitted for Anonymous Review

***Abstract*—Voxel-based LiDAR–inertial odometry (LIO) is accurate and efficient but can suffer from geometric inconsistencies when single-Gaussian voxel models indiscriminately merge observations from conflicting viewpoints. To address this limitation, we propose Azimuth-LIO, a robust voxel-based LIO framework that leverages azimuth-aware voxelization and probabilistic fusion. Instead of using a single distribution per voxel, we discretize each voxel into azimuth-sectorized substructures, each modeled by an anisotropic 3D Gaussian to preserve viewpoint-specific spatial features and uncertainties. We further introduce a direction-weighted distribution-to-distribution registration metric to adaptively quantify the contributions of different azimuth sectors, followed by a Bayesian fusion framework that exploits these confidence weights to ensure azimuth-consistent map updates. The performance and efficiency of the proposed method are evaluated on public benchmarks including the M2DGR, MCD, and SubT-MRS datasets, demonstrating superior accuracy and robustness compared to existing voxel-based algorithms.**

***Index Terms*—SLAM, LIDAR-inertial odometry (LIO), Probability Distribution, Sensor Fusion.**

I. INTRODUCTION

SIMULTANEOUS Localization and Mapping (SLAM) is a foundational technology for robotic autonomous perception and navigation, with LiDAR-based SLAM offering robust performance in critical scenarios such as disaster response and industrial inspection [1]. LIO serves as the computational core of the SLAM frontend, critically determines the robustness of subsequent mapping and navigation modules. Existing LIO methodologies operate under two primary paradigms: feature-based approaches that employ geometric primitives for registration [2], and probabilistic approaches that utilize spatial distribution modeling for uncertainty-aware pose estimation via distribution matching [3] [4] [5] [6].

The primary challenge stems from sampling-induced geometric truncation, where environmental occlusions and constrained sensor perspectives generate incomplete surface representations that bias statistical parameter estimation. Compounding this issue, cross-view observation conflicts (as demonstrated in Fig.1) are an inherent limitation in probabilistic registration, where existing voxel update strategies indiscriminately merge point cloud statistics across geometrically inconsistent viewpoints. This arbitrary fusion ignores the significant azimuth dependence of LiDAR measurements—a point observed from different poses (due to varying incidence angles and occlusions) leads to fundamentally different surface parameter estimations within the same voxel. This leads to the progressive accumulation of covariance estimation errors, directly compromising the accuracy and robustness of the LIO system, particularly in scenes with complex geometry or dynamic occlusions. Motion-distortion coupling further compounds these issues by introducing spatiotemporally correlated artifacts in single-frame point clouds during ego-motion, leading to propagating distortions in distribution parameters as errors advance from point cloud registration to map fusion stages. These challenges collectively originate from LiDAR's inherent partial observation characteristics and the complexities of multi-scan data fusion under dynamic sensing conditions.

Based on the above analysis, we propose an azimuth-sectorized voxel registration framework. This framework establishes an explicit correlation between sensor pose and voxel geometry through azimuth-aware modeling, achieving robust registration via directional probabilistic fusion. The specific contributions can be summarized as follows:

1) We decompose voxels into azimuth-based substructures while maintaining structural integrity, and introduce a directionally weighted registration metric that explicitly encodes the sensor-to-subvoxel geometric relationship for probabilistic distribution matching.
2) We develop a Bayesian fusion framework by incorporating azimuth-derived geometric weights to modulate measurement confidence in the update process, achieving perspective-consistent observation fusion through viewpoint-dependent probabilistic inference.
3) Comprehensive evaluations on public datasets validate the feasibility of the proposed algorithm. Compared with other state-of-the-art (SOTA) methods, our approach demonstrates enhanced robustness and higher accuracy. For the community's benefit, we will open-source the code at `Azimuth-LIO`.

The rest of this article is organized as follows. Section II presents related work. Section III gives details of the proposed method. The experimental evaluations are shown in Section IV. Finally, Section V concludes this article.

II. RELATED WORK

The integration of Inertial Measurement Units (IMUs) significantly enhances LiDAR-Inertial Odometry (LIO) precision and robustness. Existing approaches fall into loosely- or tightly-coupled frameworks [7]. Loosely-coupled systems apply IMU motion priors to correct LiDAR motion distortion [2], while tightly-coupled methods perform joint state estimation. Optimization-based tight coupling establishes pose constraints through IMU pre-integration [8], optimized with LiDAR residuals (e.g., point-to-plane distance). Filter-based approaches like EKF recursively fuse IMU predictions with LiDAR updates [9]–[11], building upon rigorous manifold error-state formulations [12].

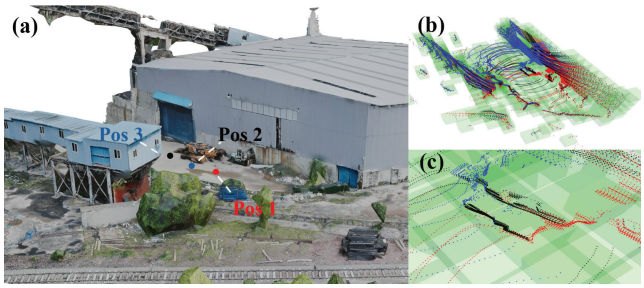

Fig. 1. (a) A representative industrial safety–rescue scene. Three static LiDAR scanning positions (Pos 1–Pos 3) are deployed around the entrance of the building. (b) 3D point clouds acquired from the three positions, color-coded by sensor location (red: Pos 1, black: Pos 2, blue: Pos 3) and overlaid on the voxelized map. (c) Close-up view of the highlighted area, showing that points falling into the same voxel exhibit markedly different spatial distributions and incidence directions when observed from different positions, which demonstrates the strong azimuth dependence of the measurements.

For point cloud representation, FAST-LIO2 [10] implements the ikd-tree to support neighborhood search in $O(m \log n)$ time. Similarly, LOG-LIO [13] extends the ikd-tree for voxel management, enabling the extraction and maintenance of local geometric features. In contrast, voxel-based approaches, such as NDT [3], achive $O(1)$ queries via hashed voxel grids. Recent advances, like Faster-LIO [14], replace the ikd-tree with parallel sparse voxels to accelerate scan registration. Our voxel-based design inherently supports multi-view modeling: Independent voxel partitioning preserves perspective-specific geometric features to mitigate partial observation bias, while concurrent processing maintains computational efficiency.

Probabilistic registration addresses a core gap in geometric methods: conventional pipelines iteratively minimize geometric residuals [10] yet seldom model measurement uncertainty explicitly. VoxelMap [15], [16] introduced a probabilistic voxel map that models sensor noise and planar uncertainty, enabling adaptive-resolution mapping via point-to-plane matching. D-LIOM [17] is a tightly coupled direct LiDAR–inertial odometry framework that registers raw scans to a probabilistic occupancy voxel submap and fuses them with IMU preintegration and a gravity prior in a local factor graph to jointly estimate poses and IMU biases in real time. PLACE-LIO [18] likewise assume planar environments—using, respectively, a fine-to-coarse multi-scale Gaussian map with online noise covariance estimation, and a plane-occupancy voxel grid with hierarchical data association for plane-to-plane matching. Nevertheless, this point-distribution paradigm still processes dense per-frame observations, incurring substantial computation and making accuracy sensitive to plane-fitting fidelity.

By contrast, L2-NDT [6] inaugurates distribution-to-distribution registration via dual voxelization—of both observation and map frames—under a Gaussian approximation, markedly accelerating alignment. The LiTAMIN [19] introduces a morphology-aware paradigm whose objective couples geometric residuals with distribution-shape mismatch by optimizing KL divergence within the GICP framework [4]. Yet extending this paradigm to LIO remains implementation-intensive. Building on LiTAMIN [19] and VoxelMap [15], LIO-GVM [20] reformulates distribution-to-distribution registration, thereby extending L2-NDT [6] to LIO, it preserves full per-voxel covariance descriptors and employs the squared Hellinger distance to temper mismatched associations. Despite these gains, two issues persist: (1) single-Gaussian voxels underfit complex intra-voxel geometry; (2) discrepancies induced by observation-pose variation during mapping are insufficiently addressed. Although GMMs [21] could enrich structural fidelity, their computational burden currently precludes real-time odometry.

To overcome these challenges, we propose an azimuth-weighted probabilistic registration framework that dynamically adapts to sensor pose variations. Our method effectively mitigates partial observation bias by integrating multi-perspective geometric constraints through Bayesian fusion, while maintaining computational efficiency. This approach preserves the advantages of distribution-to-distribution registration while addressing the inherent limitations of existing single-Gaussian representations.

III. METHODOLOGY

The overall system workflow is illustrated in Fig. 2. LiDAR and IMU measurements are first preprocessed for state prediction and scan undistortion. The undistorted points are then inserted into a hash-based voxel map (Fig. 2(a)), which is further decomposed into azimuth-partitioned substructures (Fig. 2(b)). Based on these directional sub-voxels, distribution-to-distribution scan-to-map registration provides azimuth-aware residuals for the IESKF state estimator (Fig. 2(c)). Finally, the probabilistic map is updated via Bayesian fusion of the optimized states and azimuth-derived confidence weights (Fig. 2(d)).

A. IMU Predicts

We define the notations $(\cdot)^W$, $(\cdot)^L$, and $(\cdot)^I$ as the WORLD frame, LiDAR frame, and IMU frame, respectively. The origin of the world frame coincides with the initial IMU frame. In inertial navigation systems, the IMU provides high-frequency angular velocity $\boldsymbol{\omega}$ and linear acceleration $\mathbf{a}$ measurements to predict the robotic state x, expressed as:

$$x = \left[R_I^W, \mathbf{t}_I^W, \mathbf{v}_I^W, \mathbf{b}_a, \mathbf{b}_\omega\right] \in SO(3) \times \mathbb{R}^{12} \tag{1}$$

where $R_I^W \in SO(3)$, $\mathbf{t}_I^W \in \mathbb{R}^3$, and $\mathbf{v}_I^W \in \mathbb{R}^3$ represent the rotation matrix, position, and velocity of the IMU frame relative to the world frame, respectively. $\mathbf{b}_a$ and $\mathbf{b}_\omega$ denote the bias vectors of the accelerometer and gyroscope. The extrinsic transformation between the IMU frame and the world frame is denoted as $T_I^W = (R_I^W, \mathbf{t}_I^W)$.

The IMU integration prediction is governed by:

$$\begin{cases} \mathbf{t}_k^W = \mathbf{t}_{k-1}^W + \Delta t \mathbf{v}_{k-1}^W + \frac{1}{2}\bar{\mathbf{a}}_{k,k-1}\Delta t^2, \\ \mathbf{v}_k^W = \mathbf{v}_{k-1}^W + \bar{\mathbf{a}}_{k,k-1}\Delta t, \\ R_k^W = R_{k-1}^W \cdot \mathrm{Exp}\left(\bar{\boldsymbol{\omega}}_{k,k-1}\Delta t\right), \\ \mathbf{b}_{a,k} = \mathbf{b}_{a,k-1}, \quad \mathbf{b}_{\omega,k} = \mathbf{b}_{\omega,k-1}. \end{cases} \tag{2}$$

where k is the end time of the k-th LiDAR scan, $\Delta t = t_k - t_{k-1}$ is the discretized time interval, $\bar{\boldsymbol{\omega}}$ and $\bar{\mathbf{a}}$ are the midpoint-integrated angular velocity and acceleration, that are based on [22], and $\mathrm{Exp}(\cdot)$ is the exponential map from Lie algebra

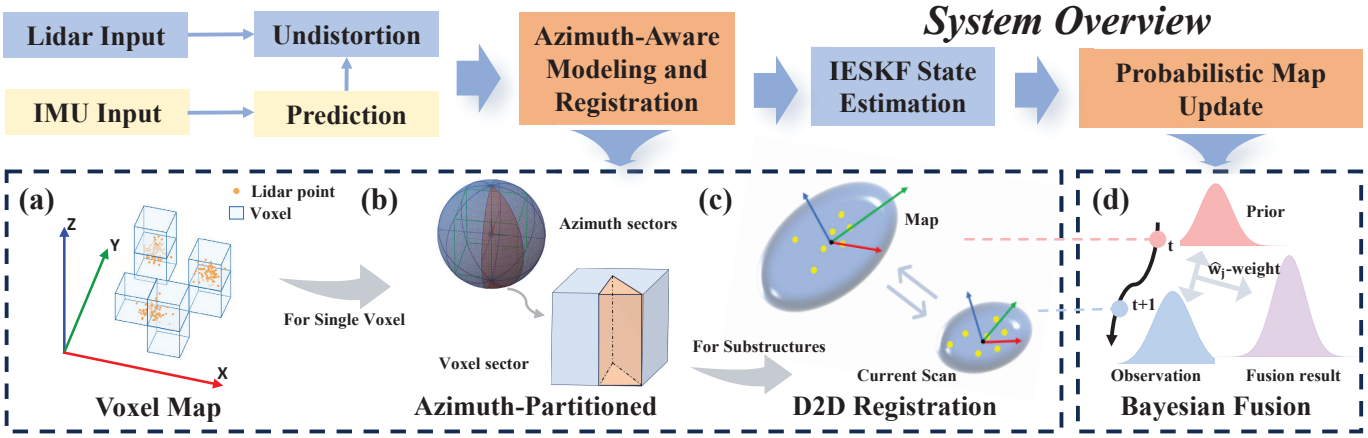

Fig. 2. System framework of the proposed Azimuth-LIO. The top row illustrates the system pipeline, while the bottom row details the key components. Modules highlighted in orange represent our proposed contributions, including the azimuth-aware modeling and probabilistic map update, while blue and yellow modules represent standard LIO components. (a) Voxel map structure. (b) Azimuth-partitioned voxelization. (c) Distribution-to-distribution (D2D) registration. (d) Bayesian fusion for map updates.

$\mathfrak{so}(3)$ to Lie group $SO(3)$ [10]. Gravitational acceleration is compensated as a constant.

We adopt the iterated error state Kalman filter (IESKF) framework [10] [12], decoupling the nonlinear state dynamics into a nominal state $\hat{x} \in \mathcal{M}$ on the manifold and an error state $\delta_{\mathbf{x}} \in \mathbb{R}^{15}$ in the local tangent space [10].

The IESKF prediction equations are:

$$\delta \mathbf{x}_{k+1} = \mathbf{F}_{\delta k} \delta \mathbf{x}_k + \mathbf{F}_k \mathbf{w}_k \tag{3}$$

$$\hat{\mathbf{P}}_{k+1} = \mathbf{F}_{\delta k} \hat{\mathbf{P}}_k \mathbf{F}_{\delta k}^T + \mathbf{F}_k \mathbf{W} \mathbf{F}_k^T \tag{4}$$

where $\mathbf{F}_{\delta k}$ and $\mathbf{F}_k$ are the perturbation Jacobian and the noise propagation matrix, respectively. $\mathbf{w}_k = [\mathbf{n}_{b_a}^T, \mathbf{n}_{b_\omega}^T, \mathbf{n}_a^T, \mathbf{n}_\omega^T]^T$ is the system noise vector, $\hat{\mathbf{P}}_k$ is the error-state covariance, and $\mathbf{W}$ is the system noise covariance matrix. The implementation follows the methodology in [12].

For each LiDAR point ${}^L p_i$ at $\tau_i \in [t_{k-1}, t_k]$, its motion-compensated coordinates in the world frame are:

$${}^W p_i = \underbrace{T_I^W}_{\substack{\text{World} \\ \text{frame}}} \cdot \underbrace{T_I(\tau_i)}_{\substack{\text{IMU motion} \\ \tau_i \to t_k}} \cdot \underbrace{T_L^I}_{\substack{\text{LiDAR-IMU} \\ \text{extrinsic}}} \cdot {}^L p_i \tag{5}$$

Here, the relative IMU motion $T_I(\tau_i)$ from τ_i to the reference time t_k is interpolated using adjacent IMU states predicted by the median-integrated angular velocity $\bar{\boldsymbol{\omega}}_{k,k-1}$ and acceleration $\bar{\mathbf{a}}_{k,k-1}$ in (2).

B. Azimuth-Partitioned Voxel Representation

1) **Azimuth-based Substructure Division**. Conventional voxelization models local geometries via single Gaussians in fixed-size voxels. This approach, however, erroneously merges surfaces from opposing perspectives in complex geometries (e.g., Fig. 1), producing degenerate covariance matrices and biased normals—issues exacerbated by low-resolution voxels.

We propose an azimuth-partitioned voxelization approach anchored in the global coordinate system, with the standing assumption that motion is largely confined to the global xy-plane—a common condition for ground-robot applications. The distortion-corrected points are first aligned to the global frame using prior state estimates. The process then proceeds with hash-based voxelization and angular partitioning. Hash functions generate per-voxel indices, and the XY-plane is subdivided into angular subregions as shown in Fig. 3. Specifically, starting from the positive X-axis ($\phi_{\text{ref}} = 0$) with clockwise increasing angles, the 2π range is equally divided into N subregions with angular intervals $\Delta\phi$:

$$S_j = [\phi_{\text{ref}} + j\Delta\phi,\ \phi_{\text{ref}} + (j+1)\Delta\phi)\,, \quad j = 0, \ldots, N-1 \tag{6}$$

where $\Delta\phi = \frac{2\pi}{N}$. For a point cloud ${}^W p_i = [x_i, y_i, z_i]$, we perform voxelization with a voxel resolution r. The voxel index $I_v = [i_x, i_y, i_z]$ corresponding to the point is computed as a standard discretization step [22]. The physical center coordinates of the voxel, denoted $C_v = [c_x, c_y, c_z]$, are calculated as: $c_x = (i_x + 0.5) \cdot r, \quad c_y = (i_y + 0.5) \cdot r, \quad c_z = (i_z + 0.5) \cdot r$. Voxel data is managed using Cuckoo Hashing [23], which provides $\mathcal{O}(1)$ query time and supports efficient multi-threaded operations. The azimuth angle ϕ_i of point $\mathbf{p}_i$ relative to voxel center C_v is computed as:

$$\phi_i = \left(90^\circ - \frac{180}{\pi} \arctan 2\big(x_i - c_x,\ y_i - c_y\big)\right) \mod 360^\circ \tag{7}$$

measured clockwise from $+X$.

The point ${}^W p_i$ is assigned to a sub-region S_j within the voxel, where: $j = \left\lfloor \frac{\phi_i}{\Delta\phi} \right\rfloor$, and $\lfloor \cdot \rfloor$ denotes the floor function, ensuring $j \in \{0, 1, \ldots, N-1\}$.

For the current observed point cloud, we independently compute the mean μ_j and covariance matrix Σ_j for each voxel substructure S_j. Specifically, given all points $\{{}^W p_i\}_{i=1}^{N_j}$ within S_j, the statistics are calculated as:

$$\mu_j = \frac{1}{N_j} \sum_{i=1}^{N_j} {}^W p_i, \quad \Sigma_j = \frac{1}{N_j} \sum_{i=1}^{N_j} \left({}^W p_i - \mu_j\right) \left({}^W p_i - \mu_j\right)^T \tag{8}$$

where N_j is the number in S_j. Simultaneously, the mean of the entire voxel is dynamically updated for correspondence search.

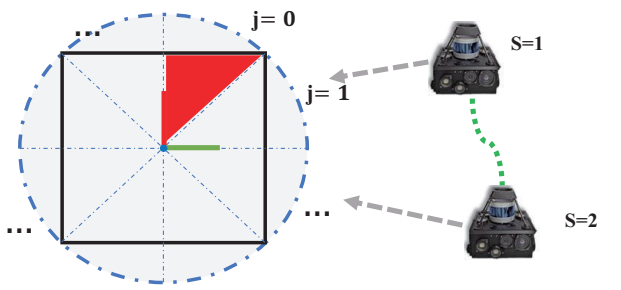

Fig. 3. Top-down planar view of a single voxel partitioned in plane. The red region indicates one voxel substructure currently observed from the vehicle. Observing the same voxel from different vehicle positions corresponds to different voxel substructures.

In practice, as the point cloud from a single scan sparsifies with range from the sensor, any substructure S_j with too few points ($N_j \leq 3$) is discarded for that frame. Fig. 4 shows that different sensor positions yield differences in the points observed within the voxel substructures, the points from the substructure that aligns with the sensor's azimuth are highlighted in red.

2) **Sensor-Azimuth-Based Weight Assignment**. To mitigate bias from partial observations, we propose an azimuth-adaptive weighting mechanism that prioritizes substructures geometrically aligned with the sensor position. As shown in Fig. 3, the azimuth angle of the Lidar relative to the voxel center identifies the reference substructure S_s. Substructures S_j are assigned confidence weights based on their angular proximity to S_s, defined as:

$$\hat{w}_j = \alpha^{d(s,j)}, \quad d(s,j) = \min\left(|j-s|, N-|j-s|\right) \tag{9}$$

where $d(s,j)$ is the minimal circular distance between substructures S_j and S_s, and $\alpha \in (0,1)$ controls the spatial confidence decay rate.

The weighting mechanism prioritizes substructures facing the sensor (small $d(s,i)$), which receive higher confidence due to reduced occlusion risks, while the exponential decay ensures smooth transitions between adjacent sectors to suppress partitioning-induced noise. By integrating substructure covariances with azimuthal confidence weights, the framework adaptively suppresses angularly inconsistent observations, effectively mitigating the degeneracy issues inherent in conventional single-Gaussian models.

C. Distribution-to-Distribution Registration

For the i-th substructure of the j-th voxel, the LiDAR residual $r_{j,i}^{\text{LIDAR}}$ is defined as:

$$r_{j,i}^{\text{LIDAR}} = \mu_{j,i}^{\text{obs}}(T_I^W) - \mu_{j,i}^{\text{map}} \tag{10}$$

The correspondence between observed and map voxels is established via a voxel centroid matching scheme coupled with a spatial 7-neighborhood search strategy [20].

The information matrix Ω_i^{LIDAR}, which quantifies the uncertainty of the residual, is computed by:

$$\Omega_i^{\text{LIDAR}} = \left(\Sigma_{j,i}^{\text{obs}} + \Sigma_{j,i}^{\text{map}} + \alpha I\right)^{-1} \tag{11}$$

For numerical stability, we usually set α around 10^{-4} [20].

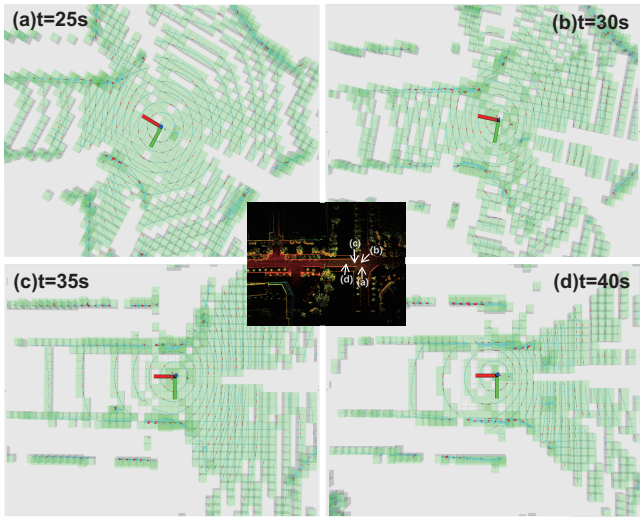

Fig. 4. Single LiDAR scans from *street_06* dataset at 25 s, 30 s, 35 s, and 40 s. Each scan is voxelized (green); all points are plotted in blue. Points belonging to the substructure aligned with the vehicle's azimuth are marked in red.

The registration cost aggregates weighted residuals from all active substructures across matched voxels:

$$E(T_I^W) = \sum_{i \in \mathcal{A}_j} \hat{w}_{j,i} \cdot \left(r_{j,i}^{\text{LIDAR}}\right)^\top \Omega_{j,i}^{\text{LIDAR}} \left(r_{j,i}^{\text{LIDAR}}\right) \tag{12}$$

where $\mathcal{A}_j$ denotes the set of indices corresponding to active substructures in the j-th voxel.

D. State Estimation Based on IESKF

By fusing the state prior with valid distribution-to-distribution matching constraints, the maximum a posteriori (MAP) estimation of the error state $\delta\boldsymbol{x}$ is formulated as:

$$\arg\min_{\delta\boldsymbol{x}} \left(\sum_{j=1}^{N_L} \left\| \boldsymbol{r}_j^{\text{LIDAR}} \right\|^2_{\boldsymbol{\Omega}_j^{\text{LIDAR}}} + \left\| \boldsymbol{r}^{\text{IMU}} \right\|^2_{\boldsymbol{\Omega}^{\text{IMU}}} \right) \tag{13}$$

where N_L is the number of LiDAR constraints (the sum of the activated substructures in all voxels), and $\|\boldsymbol{r}\|^2_{\boldsymbol{\Omega}} = \boldsymbol{r}^\top \boldsymbol{\Omega}^{-1} \boldsymbol{r}$ denotes the Mahalanobis norm.

This nonlinear least-squares problem is solved iteratively via the Gauss-Newton algorithm. At the n-th iteration, the state update $\delta\boldsymbol{x}^n$ is computed by solving:

$$\boldsymbol{H}^n \delta\boldsymbol{x}^n = \boldsymbol{b}^n \tag{14}$$

where the approximate Hessian $\boldsymbol{H}^n$ and gradient vector $\boldsymbol{b}^n$ are derived from residual Jacobians:

$$\boldsymbol{H}^n = \sum_i \boldsymbol{J}_i^\top \boldsymbol{\Omega}_i \boldsymbol{J}_i, \quad \boldsymbol{b}^n = \sum_i \boldsymbol{J}_i^\top \boldsymbol{\Omega}_i \boldsymbol{r}_i. \tag{15}$$

The LiDAR residual Jacobian w.r.t. $\delta\boldsymbol{x} = [\delta\boldsymbol{\theta}^\top, \delta\boldsymbol{t}^\top, \delta\boldsymbol{v}^\top, \delta\boldsymbol{b}_a^\top, \delta\boldsymbol{b}_\omega^\top]^\top$ is:

$$\begin{aligned} \boldsymbol{J}^{\text{LIDAR}} &= \frac{\partial \boldsymbol{r}^{\text{LIDAR}}}{\partial \delta\boldsymbol{x}} \\ &= \begin{bmatrix} \boldsymbol{R}_I^W \lfloor \boldsymbol{\mu}^{\text{obs}} \rfloor_\times & -\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \end{bmatrix} \end{aligned} \tag{16}$$

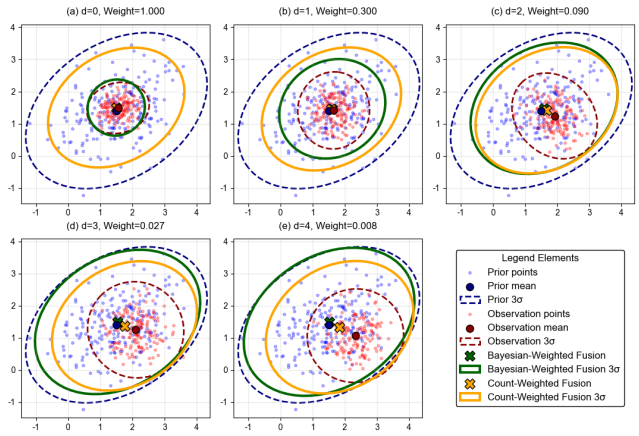

Fig. 5. Comparison of Geometry-Weighted Bayesian Fusion and Count-Weighted Fusion under Multi-View Observations ($\alpha = 0.3$, both the prior and observation point clouds are maintained at constant counts).

where $\lfloor\cdot\rfloor_\times$ is the skew-symmetric operator. For IMU residuals:

$$\boldsymbol{J}^{\text{IMU}} = \frac{\partial \boldsymbol{r}^{\text{IMU}}}{\partial \delta \boldsymbol{x}} = \begin{bmatrix} \boldsymbol{J}_r^{-1}\Big(\text{Log}\Big((\boldsymbol{R}_I^W)^\top \hat{\boldsymbol{R}}_I^W\Big)\Big) & \boldsymbol{0}_{3\times 12} \\ \boldsymbol{0}_{12\times 3} & \boldsymbol{I}_{12\times 12} \end{bmatrix} \tag{17}$$

where $\boldsymbol{J}_r^{-1} : \mathfrak{so}(3) \to \mathbb{R}^{3\times 3}$ denotes the inverse right Jacobian of the rotation group.

Post-convergence, the nominal state is updated via:

$$\hat{\boldsymbol{x}}_k^{n+1} = \hat{\boldsymbol{x}}_k^n \boxplus \delta \boldsymbol{x}^n \tag{18}$$

and the covariance is projected to the new tangent space:

$$\hat{\boldsymbol{P}}_k = \boldsymbol{L}_k (\boldsymbol{H}^n)^{-1} \boldsymbol{L}_k^\top, \quad \boldsymbol{L}_k = \begin{bmatrix} \boldsymbol{I}_3 - \frac{1}{2}(\delta\boldsymbol{\theta})^\wedge & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_{12} \end{bmatrix} \tag{19}$$

where $\boldsymbol{L}_k$ ensures consistency between the covariance and manifold retraction. Neglecting this projection risks covariance misalignment with true error distributions [12]. The error state is then reset for subsequent observations.

E. Voxel–Substructure Map Update

To enable efficient map updates under diverse observation perspectives, we propose a Bayesian fusion framework with perspective-adaptive reliability weights. Each voxel substructure is modeled as an anisotropic Gaussian prior $\mathcal{N}(\boldsymbol{\mu}_{j,i}^{\text{map}}, \boldsymbol{\Sigma}_{j,i}^{\text{map}})$ representing the current map state, whereas sensor observations are incorporated as likelihood distributions $\mathcal{N}(\boldsymbol{\mu}_{j,i}^{\text{obs}}, \boldsymbol{\Sigma}_{j,i}^{\text{obs}})$ characterizing the sensor measurements. At initialization, the first scan defines the global frame, and each voxel substructure is initialized as an anisotropic Gaussian prior. The weight $\hat{w}_j \in [0, 1]$ (Section III-B), which quantifies the geometric alignment between the sensor observation direction and the substructure dominant orientation. This weight adaptively prioritizes azimuth-aligned observations ($\hat{w}_j \approx 1$) and attenuates oblique perspectives ($\hat{w}_j \approx 0$).

The posterior parameters are derived via weighted Bayesian fusion:

$$\boldsymbol{\Sigma}_{j,i}^{\text{post}} = \left((\boldsymbol{\Sigma}_{j,i}^{\text{map}})^{-1} + \hat{w}_j (\boldsymbol{\Sigma}_{j,i}^{\text{obs}})^{-1}\right)^{-1} \tag{20}$$

$$\boldsymbol{\mu}_{j,i}^{\text{post}} = \boldsymbol{\Sigma}_{j,i}^{\text{post}} \left((\boldsymbol{\Sigma}_{j,i}^{\text{map}})^{-1} \boldsymbol{\mu}_{j,i}^{\text{map}} + \hat{w}_j (\boldsymbol{\Sigma}_{j,i}^{\text{obs}})^{-1} \boldsymbol{\mu}_{j,i}^{\text{obs}}\right) \tag{21}$$

The closed-form updates operate in $\mathcal{O}(1)$ time per voxel, requiring no raw data storage and maintaining only $\{\boldsymbol{\mu}_{j,i}^{\text{map}}, \boldsymbol{\Sigma}_{j,i}^{\text{map}}\}$.

Unlike the fusion method which is based solely on point cloud density in LIO-GVM [20], our Bayesian-weight fusion dynamically adjusts observation weights based on angular alignment. When the sensor faces the target region (Fig. 5(a)), a weight of 1 is applied, yielding precise mean correction and significant posterior covariance contraction. As the observation angle deviates (Figs. 5(d)-(e)), the weight decays exponentially, gradually reducing the influence of off-axis observations and causing the posterior to remain closer to the prior. This mechanism reflects the inherent reliability of frontal measurements—due to an unobstructed field of view and reduced occlusion—and effectively suppresses perspective-induced bias, thereby enhancing long-term consistency in dynamic environments.

IV. EXPERIMENTS

Comparative experiments are conducted to systematically evaluate performance of the proposed method across multiple scenarios. All algorithms are deployed on a computational platform featuring an Intel i7-12700H processor and 32 GB RAM, running the Robot Operating System (ROS) on Ubuntu 20.04 LTS. Core components are implemented in C++ for optimized execution. Evaluation metrics include the root mean square error (RMSE) in Absolute Trajectory Error (ATE) for localization accuracy assessment [24], computational efficiency analysis through runtime measurements, and systematic ablation studies to verify component effectiveness.

A. Experimental Setup

Our framework was comprehensively validated on three datasets, comprising three public benchmarks (M2DGR [25], MCD [26], and SubT-MRS).

M2DGR was acquired by a ground robot equipped with a Velodyne VLP-32C LiDAR and a 9-axis IMU sampling at 150 Hz; ground-truth trajectories were provided by a high-precision GPS/BeiDou integrated navigation system. The sequences include canonical motion patterns—circle, loop-back, and zigzag—offering substantial trajectory diversity. MCD contains both vehicle-mounted and handheld recordings, captured with a 10 Hz OS1-64/128 LiDAR and the OUSTER-integrated 6-axis IMU operating at 100 Hz. SubT-MRS was collected in subterranean tunnel environments typified by complex spatial structures and perceptual degradation, it employs a 10 Hz Velodyne VLP-16 and a 200 Hz Epson M-G365 IMU with precise hardware time synchronisation, making it suitable for robustness evaluation under extreme conditions.

For comparative experiments, we benchmark three representative voxel-based algorithms—PLACE-LIO [18], LOG-LIO [13], and VOXELMAP++ [16]—and adopt DLIO [27] and FAST-LIO2 [10] as non-voxel baselines. To ensure fairness and reproducibility, each method uses fixed hyperparameters and calibration across all sequences without per-sequence tuning. Our method sets the voxel resolution to $1\,\text{m}$ and discretizes each voxel into K=8 azimuth sectors with $\Delta\phi$=45°;

TABLE I
COMPREHENSIVE ATE EVALUATION ACROSS DIVERSE DATASETS AND MOTION SCENARIOS

Sequence	Duration (s)	PLACE-LIO	LOG-LIO	VOXELMAP++	FAST-LIO2	DLIO	Azimuth-LIO
walk01	291	0.092	0.115	0.096	0.115	0.118	**0.090**
gate01	172	**0.137**	0.179	0.155	0.187	0.212	0.149
gate02	327	**0.314**	0.328	1.309	0.329	0.332	0.316
gate03	283	0.164	0.203	0.211	0.198	0.171	**0.129**
street01	1028	0.208	0.311	x[b]	0.345	0.524	**0.124**
street02	1227	3.077	2.641	x	**2.311**	3.240	2.474
street06	494	0.323	0.273	1.105	0.305	0.442	**0.211**
rot01	234	0.130	1.535	0.493	1.290	0.405	**0.118**
rot02	244	**0.141**	1.211	1.965	1.201	0.279	0.150
ntu02	229	—[a]	0.486	x	**0.242**	0.533	0.356
ntu04	297	—	x	x	2.471	2.442	**1.446**
ntu13	234	—	1.600	x	**0.813**	1.065	1.063
kth05	665	—	**1.201**	x	4.223	4.452	1.808
kth06	891	—	x	x	2.371	3.117	**1.996**
tuhh07	444	—	1.908	x	0.661	**0.375**	0.522
tuhh08	709	—	22.345	x	2.365	3.476	**1.154**
tuhh09	185	—	**0.191**	x	0.351	0.456	0.318
subt_ugv1	1600	0.307	0.513	x	0.715	0.335	**0.298**
subt_ugv2	3390	x	0.044	x	x	0.051	**0.036**
subt_ugv3	1714	x	1.273	x	11.351	0.732	**0.241**
urban_ugv1	513	0.033	0.052	0.052	0.201	0.051	**0.031**
urban_ugv2	3120	**0.166**	0.205	x	3.642	0.214	0.174

Bold: Best performance; Underlined: Second best; [a] denotes that the algorithm is not applicable; [b] indicates that the algorithm diverges. *Note*: The last 30s of *walk01* were omitted from the evaluation owing to unstable RTK signals.

a sector is initialized only when it contains more than three points. We maintain the formulation of the direction-weighted metric, the fusion weights, and crucially, the parameter $\alpha = 0.3$ in the weighting function $\hat{w}_j = \alpha^{d(s,j)}$ fixed throughout all experiments (empirically set and held constant). This *a priori* configuration, including the fixed weighting parameters and the set K value, balances angular specificity against per-voxel sample sufficiency and yields predictable computational cost; the impact of K is specifically analyzed in the ablation study. Baselines are evaluated at a resolution of $0.5\,\mathrm{m}$ (for non-voxel methods, this resolution is used for voxel-grid downsampling of the input). To preserve the geometric details required for multi-view fusion, our method does not apply additional downsampling to the original point clouds.

B. Accuracy Analysis

As shown in Table I, the proposed method attains accuracy comparable to PLACE-LIO on the M2DGR dataset, while PLACE-LIO remains superior in highly structured scenes (e.g., *gate02*) thanks to its hierarchical association scheme. By incorporating multi-view observations, our approach compensates for the limited structural representation of a single local voxel map and achieves clearly higher accuracy than LOG-LIO and VOXELMAP++. In sequences with aggressive rotational motion (e.g., *rot01* and *rot02*), our method stays stable due to the tightly coupled framework and robust LiDAR motion-distortion compensation, as illustrated in Fig. 6(a) and Fig. 6(b), whereas FAST-LIO2 suffers a notable performance drop, partly because of the lower-grade $150\,\mathrm{Hz}$ IMU it employs. LOG-LIO is sensitive to sensor configuration, it performs worse on 128-beam *ntu** than on 64-beam *kth*/tuhh** sequences, indicating that higher beam count and point density can even aggravate voxel-wise distribution conflicts instead of improving odometry. Our method achieves higher accuracy on 128-beam *ntu** while remaining competitive on the 64-beam data.

On the more challenging SubT-MRS dataset, with long-duration underground trajectories and narrow tunnel corridors, PLACE-LIO, LOG-LIO, and FAST-LIO2 become unstable. VOXELMAP+ suffers from amplified errors due to the lack of mergable coplanar features. Our method maintains optimal accuracy. Fig. 6(c) and Fig. 6(d) illustrate discrepancies relative to baseline reconstructions, whereas Fig. 7 exhibits detailed reconstruction quality on the highly challenging *subt_ugv1* sequence. This stems, on one hand, from the abundance of viewpoints provided by the numerous corners in the mine tunnels, which benefits the multi-view voxel model (e.g., achieving an RMSE of only 0.036 m on the *subt_ugv2* sequence). On the other hand, the minimal vertical displacement in the dataset renders the contribution of Z-axis drift to the overall RMSE negligible. Trajectories from six benchmark datasets are selected and visualized in Fig. 8.

C. Ablation Study

To more comprehensively assess the impact of our method on LiDAR–Inertial Odometry, we conducted three ablation studies on the M2DGR dataset: (1) we replace our inertial module with the IMU propagation algorithm from FAST-LIO2 while keeping the multi-view substructure modeling and the view-adaptive weighting unchanged, denoted as A1-LIO; (2) we rewrite and replace the core registration module following the NDT-LIO scheme, and implement the map update module with reference to LIO-GVM [20] using a voxel size of $0.5\,\mathrm{m}$, denoted as A2-LIO; (3) to isolate the effect of the number of substructures per voxel, we discretize each voxel into $K{=}4$

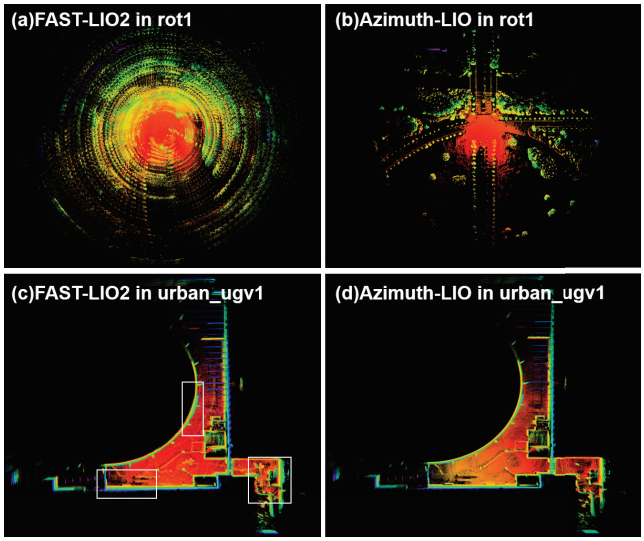

Fig. 6. (a) and (b) show the mapping results of FAST-LIO2 and Azimuth-LIO in *rot01*. Azimuth-LIO attains better motion-distortion mitigation for point clouds during high-angular-velocity rotation. (c) and (d) show the mapping results of FAST-LIO2 and Azimuth-LIO in *urban_ugv1*. White boxes mark map inconsistencies with FAST-LIO2, whereas Azimuth-LIO yields a clearer result.

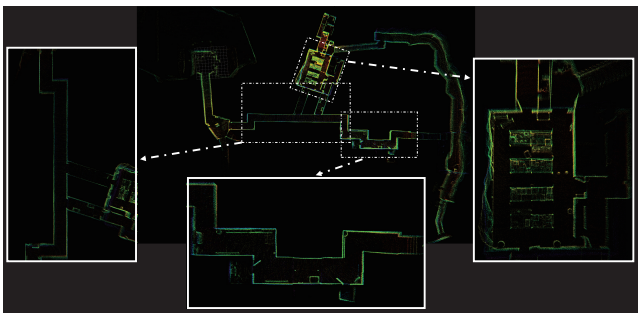

Fig. 7. Large-scale experiment in *subt_ugv1*, with zoomed-in views highlighting local details.

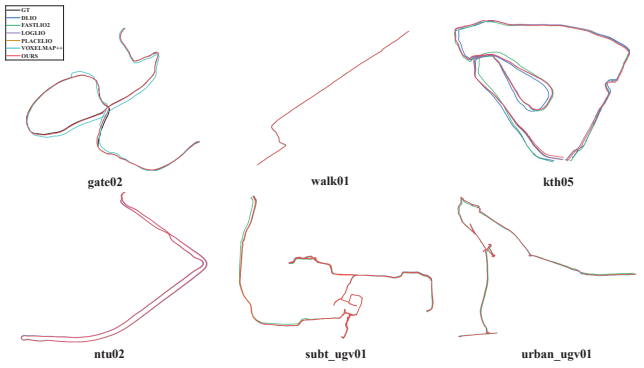

Fig. 8. Visualization of the estimated trajectories on six benchmark sequences.

azimuth sectors, denoted as A3-LIO. All other parameters are identical to those used in the experiments of IV-B.

Across the five M2DGR sequences, the mean translational accuracy of our full method is 0.177 m, with quantitative results summarized in Table II. In `rot01` and `rot02`, where the platform undergoes strong rotational motion, A1-LIO cleanly isolates the contribution of the IMU module and further confirms the limitations of the FAST-LIO2 tightly coupled scheme when used with low-cost IMUs. For A2-LIO and A3-LIO, a joint analysis shows that using four substructures per voxel yields planar-scene accuracy comparable to A2-LIO; however, our full map-update formulation better accommodates multi-view observations and, together with the view-adaptive weighting, effectively suppresses the noise growth associated with larger LiDAR incidence angles (weaker returns, lower SNR), leading to improved robustness. Results for `rot01` and `rot02` are essentially equivalent because near in-place rotational motion causes little change in viewpoint, therefore finer substructure partitioning brings only modest improvements. Notably, as motion becomes more complex (e.g., continuous yaw spins and aggressive maneuvers), the advantages of the proposed method become more pronounced across sequences.

TABLE II
ATE COMPARISON OF ABLATION VARIANTS ON M2DGR SEQUENCES

	walk01	gate02	street06	rot01	rot02	Average
A1-LIO	0.117	0.335	0.270	1.345	2.409	0.895
A2-LIO	0.124	0.336	0.286	0.123	0.152	0.204
A3-LIO	0.117	0.336	0.273	0.121	0.151	0.200

D. Efficiency Analysis

Time efficiency is evaluated under the same parameter settings as in the odometry experiments, covering LiDAR sensors with 16/32/64/128 beams. Following the time-recording methodology in DLIO, the DLO algorithm [28] is incorporated solely as a reference benchmark for runtime comparison. As illustrated in Fig. 9, our method delivers competitive time efficiency among existing state-of-the-art systems. FAST-LIO2 remains the fastest due to its lightweight map representation, whereas DLIO fails to sustain stable operation at 10 Hz. Our system employs a Cuckoo Hashing-based key–value query mechanism for feature matching, achieving $\mathcal{O}(1)$ lookup complexity. The combination of dual hash functions and dynamic table expansion effectively controls collision rates and reduces the computational overhead of hash-table operations. An LRU caching policy further promotes high cache-hit rates by relocating recently accessed voxels with minimal reordering cost. Each voxel maintains up to eight azimuthal sub-regions, and the parallelized processing strategy provides balanced workloads on multi-core CPUs. On the *urban_ugv1* dataset, the multi-threaded implementation reaches a total CPU utilization of 105%, compared with 47% for single-threaded FAST-LIO2. Using a 1.0 m voxel resolution (instead of the 0.5 m commonly used in baselines) reduces the number of maintained voxels by 70% in the same dataset (from 20,635 to 6,065), yielding a favorable accuracy–efficiency trade-off suitable for time-critical tasks such as rescue and reconnaissance in unknown environments.

V. CONCLUSION

This paper presents Azimuth-LIO, a voxel-based LiDAR–inertial odometry framework that strengthens geometric representation through azimuth-aware distribution-to-distribution registration. By sectorizing voxels in the horizontal

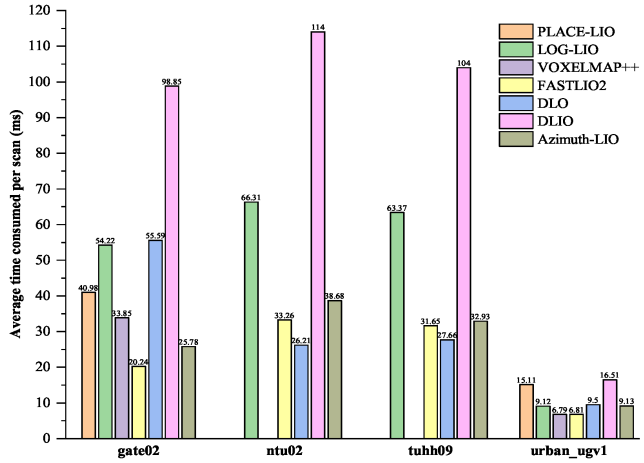

Fig. 9. Average per-scan processing latency(ms) across four datasets with different sensor configurations. PLACE-LIO and VOXELMAP++ are not compared on the MCD dataset.

plane and embedding observation-angle dependence into the probabilistic model, the method maintains a consistent link between sensor pose and voxel geometry while preserving a lightweight hash-based implementation. Experiments on public datasets show that Azimuth-LIO achieves state-of-the-art localization accuracy and competitive runtime.

In this work, Azimuth-LIO employs an azimuth sectorization anchored to the global XY-plane as a pragmatic, lightweight design that has been empirically validated on ground and handheld platforms. Building on this foundation, our future research will explore sparsity-adaptive spatial partitioning strategies and lightweight probabilistic representations that extend the azimuth-aware idea to more general 3D sectorization schemes. These developments are expected to further enhance robust LiDAR–inertial odometry and long-term map consistency under a tight real-time computational budget, thereby strengthening the system's performance under challenging 6-DoF motions.

REFERENCES

[1] J. Delmerico, S. Mintchev, A. Giusti, B. Gromov, K. Melo, T. Horvat, C. Cadena, M. Hutter, A. Ijspeert, D. Floreano *et al.*, "The current state and future outlook of rescue robotics," *Journal of Field Robotics*, vol. 36, no. 7, pp. 1171–1191, 2019.

[2] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2018, pp. 4758–4765.

[3] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3d-ndt," *Journal of Field Robotics*, vol. 24, no. 10, pp. 803–827, 2007.

[4] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp." in *Robotics: science and systems*, vol. 2, no. 4. Seattle, WA, 2009, p. 435.

[5] K. Koide, M. Yokozuka, S. Oishi, and A. Banno, "Voxelized gicp for fast and accurate 3d point cloud registration," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 11 054–11 059.

[6] T. Stoyanov, M. Magnusson, and A. J. Lilienthal, "Point set registration through minimization of the l 2 distance between 3d-ndt models," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 5196–5201.

[7] H. Ye, Y. Chen, and M. Liu, "Tightly coupled 3d lidar inertial odometry and mapping," in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 3144–3150.

[8] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[9] 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 *2020 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2020, pp. 8899–8906.

[10] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.

[11] H. Liu, S. Pan, X. Wang, W. Gao, B. Yu, and H. Zhao, "Long-term localization method integrated with voxel mapping lidar odometry and adaptive updating map in diverse environment," *IEEE Transactions on Instrumentation and Measurement*, 2025.

[12] J. Sola, "Quaternion kinematics for the error-state kalman filter," *arXiv preprint arXiv:1711.02508*, 2017.

[13] K. Huang, J. Zhao, Z. Zhu, C. Ye, and T. Feng, "Log-lio: A lidar-inertial odometry with efficient local geometric information estimation," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 459–466, 2023.

[14] C. Bai, T. Xiao, Y. Chen, H. Wang, F. Zhang, and X. Gao, "Faster-lio: Lightweight tightly coupled lidar-inertial odometry using parallel sparse incremental voxels," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4861–4868, 2022.

[15] C. Yuan, W. Xu, X. Liu, X. Hong, and F. Zhang, "Efficient and probabilistic adaptive voxel mapping for accurate online lidar odometry," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8518–8525, 2022.

[16] C. Wu, Y. You, Y. Yuan, X. Kong, Y. Zhang, Q. Li, and K. Zhao, "Voxelmap++: Mergeable voxel mapping method for online lidar (-inertial) odometry," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 427–434, 2023.

[17] Z. Wang, L. Zhang, Y. Shen, and Y. Zhou, "D-liom: Tightly-coupled direct lidar-inertial odometry and mapping," *IEEE Transactions on Multimedia*, vol. 25, pp. 3905–3920, 2022.

[18] L. He, B. Li, and G. Chen, "Place-lio: Plane-centric lidar-inertial odometry," *IEEE Robotics and Automation Letters*, 2025.

[19] M. Yokozuka, K. Koide, S. Oishi, and A. Banno, "Litamin: Lidar-based tracking and mapping by stabilized icp for geometry approximation with normal distributions," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2020, pp. 5143–5150.

[20] X. Ji, S. Yuan, P. Yin, and L. Xie, "Lio-gvm: an accurate, tightly-coupled lidar-inertial odometry with gaussian voxel map," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2200–2207, 2024.

[21] Q. Li, R. Xiong, and T. Vidal-Calleja, "A gmm based uncertainty model for point clouds registration," *Robotics and Autonomous Systems*, vol. 91, pp. 349–362, 2017.

[22] Z. Chen, Y. Xu, S. Yuan, and L. Xie, "ig-lio: An incremental gicp-based tightly-coupled lidar-inertial odometry," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1883–1890, 2024.

[23] X. Li, D. G. Andersen, M. Kaminsky, and M. J. Freedman, "Algorithmic improvements for fast concurrent cuckoo hashing," in *Proceedings of the Ninth European Conference on Computer Systems*, 2014, pp. 1–14. [Online]. Available: https://github.com/efficient/libcuckoo?tab=readme-ov-file

[24] M. Grupp, "evo: Python package for the evaluation of odometry and slam." 2017. [Online]. Available: https://github.com/MichaelGrupp/evo/

[25] J. Yin, A. Li, T. Li, W. Yu, and D. Zou, "M2dgr: A multi-sensor and multi-scenario slam dataset for ground robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2266–2273, 2021.

[26] T.-M. Nguyen, S. Yuan *et al.*, "Mcd: Diverse large-scale multi-campus dataset for robot perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 22 304–22 313. [Online]. Available: https://mcdviral.github.io/

[27] K. Chen, R. Nemiroff, and B. T. Lopez, "Direct lidar-inertial odometry: Lightweight lio with continuous-time motion correction," *arXiv preprint arXiv:2203.03749*, 2022.

[28] K. Chen, B. T. Lopez, A.-a. Agha-mohammadi, and A. Mehta, "Direct lidar odometry: Fast localization with dense point clouds," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2000–2007, 2022.