2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

GMM-LIO: Adaptive and Robust LiDAR-Inertial Odometry with Gaussian Mixture Model Voxel Map

Zishun Deng[1], Can Li[1], Wanbiao Lin[2], and Lei Sun[1]

***Abstract*— Tightly coupled LiDAR–inertial odometry (LIO) systems are critical for autonomous navigation, yet their performance often degrades due to insufficient adaptability to diverse environments and limitations in map representation. To address these limitations, this paper presents GMM-LIO, a robust and adaptive LIO framework that integrates a novel information-theoretic scan processing module and a high-fidelity Gaussian Mixture Model (GMM) voxel map structure. At its core, GMM-LIO features a two-level adaptive front-end that dynamically modulates voxel resolution based on state uncertainty and adjusts surface covariance estimation according to local point density on a standard voxel grid. Furthermore, GMM-LIO employs a dynamic Gaussian Mixture Model voxel map to accurately model intersecting surfaces. The entire system is formulated as a robust Maximum a Posteriori (MAP)-based estimator, which employs an Iteratively Reweighted Least Squares (IRLS) solver together with a principled anisotropic information matrix to handle measurement outliers. Extensive evaluations on diverse public and self-collected datasets demonstrate that GMM-LIO achieves state-of-the-art accuracy and robustness, with a 36% relative improvement over leading LIO baselines.**

I. Introduction

High-precision and robust state estimation is a fundamental capability for autonomous systems, such as self-driving vehicles and mobile robots, enabling them to navigate complex and unknown environments reliably. Among various localization modalities, LIO has emerged as a cornerstone approach, providing accurate robot state and map estimation largely independent of ambient illumination. Influential modern frameworks, such as the FAST-LIO family [1], [2] and LIO-SAM [3], have established powerful paradigms based on tightly-coupled Kalman filters and factor graph optimization, respectively, pushing the boundaries of efficiency and global consistency.

Despite these successes, LIO systems still face persistent challenges that limit their performance. A critical issue stems from the data association process itself, particularly within Generalized-ICP (GICP) [4] based methods [5]–[8]. These approaches rely on estimating local surface covariances from neighboring points to model geometry. However, the quality of this estimation is highly sensitive to the spatial distribution and density of points; sparse or irregularly sampled point sets, common in many real-world scenarios, can lead to unreliable covariance estimates and thus degrade registration accuracy. This problem is further exacerbated by the limitations of the underlying map representation. Structures based on raw point clouds [2], [9], surfels [10], or voxels limited to a single geometric primitive (e.g., a single-plane [11], [12] or a single-Gaussian distribution [13]) struggle to accurately model geometric discontinuities like corners and intersecting surfaces. This representational inadequacy leads to information loss and further corrupts the registration process. Furthermore, many systems lack adaptability, relying on fixed parameters that render them brittle when transitioning between environments or encountering sensor degradation. While recent research has addressed specific aspects of robustness, such as handling geometric degeneracy [14], [15] or adapting sensor noise models [16], a unified framework that holistically adapts to both environmental structure and measurement statistics remains an open challenge.

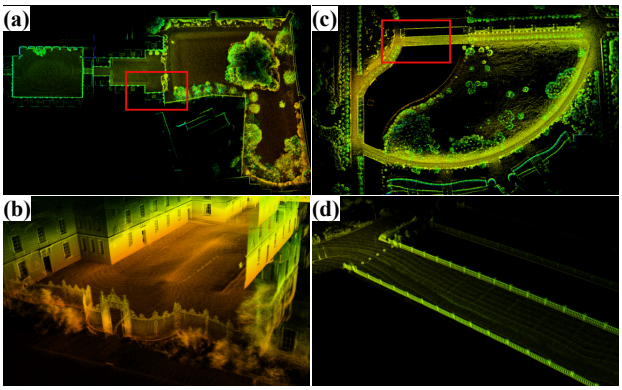


Fig. 1: Mapping results of GMM-LIO in diverse environments. The top row shows the final maps from the Newer College Dataset [17] (a) and the M2DGR Dataset [18] (c). The bottom row shows the corresponding magnified views (b, d) of the red-boxed areas.

To this end, this paper presents GMM-LIO, a robust and adaptive LIO framework that integrates a perceptive, information-theoretic front-end with a novel high-fidelity map representation. The framework incorporates an adaptive scan processing module, which modulates voxel grid resolution according to posterior state uncertainty and refines surface covariance estimation based on local point density. In addition, we propose a Dynamic Gaussian Mixture Model Voxel Map that effectively represents complex intersecting surfaces and sharp features, thereby alleviating the limitations of single-Gaussian assumptions. To ensure robustness against measurement outliers, we adopt a MAP-

This work was supported by the National Natural Science Foundation of China (No. 62173192 and No. 62373365).

[1]Zishun Deng, Can Li, and Lei Sun are with the Institute of Robotics and Automatic Information System; the Engineering Research Center of Trusted Behavior Intelligence, Ministry of Education; and the Tianjin Key Laboratory of Intelligent Robotics, Nankai University, Tianjin 300350, China (e-mail: {2120240552, canli}@mail.nankai.edu.cn, sunl@nankai.edu.cn). *Corresponding author: Lei Sun.*

[2]Wanbiao Lin is with the Shenzhen Research Institute, Nankai University, Shenzhen, China (e-mail: 2120160381@mail.nankai.edu.cn).

based formulation solved via IRLS, where the use of an anisotropic information matrix enables selective attenuation of inconsistent observations. Our observation model is conceptually related to VGICP [5]. Unlike VGICP, however, the point distributions of the current scan are estimated using a standard voxel grid, whereas the local map leverages a richer GMM representation for high-fidelity matching. Fig. 1 illustrates the high-quality mapping results produced by GMM-LIO in diverse environments.

The main contributions of this work are summarized as follows:

- A novel information-theoretic adaptive surface covariance estimation that dynamically adjusts voxel resolution based on posterior state uncertainty and modulates surface covariance estimation according to local point density.
- A novel Dynamic Gaussian Mixture Model Voxel Map that provides high-fidelity modeling of complex geometric structures and supports efficient incremental updates and dynamic component splitting.
- A principled robust MAP estimator that derives an anisotropic information matrix from a second-order approximation of the robust kernel, enabling more precise outlier suppression than standard IRLS methods.
- Extensive validation on diverse public benchmarks and self-collected datasets, demonstrating superior accuracy and robustness over state-of-the-art methods.

II. Related Work

LiDAR-inertial odometry has emerged as a fundamental technology for autonomous navigation by tightly coupling the complementary strengths of LiDAR and IMU sensors. High-frequency IMU measurements are instrumental for compensating LiDAR motion distortion and providing high-rate motion predictions [3], while LiDAR's accurate geometric measurements provide robust constraints to mitigate the IMU's long-term drift.

At the core of any LIO system lies the challenge of robust point cloud registration. To this end, methods have evolved beyond the classic Iterative Closest Point (ICP) towards probabilistic approaches like the GICP [4]. GICP models registration as a distribution-to-distribution alignment problem by estimating a surface covariance for each point or local point set to capture the underlying geometry. This probabilistic formulation is a step towards principled uncertainty modeling and provides inherent robustness against measurement noise. However, the fidelity of this covariance estimation, which relies on efficiently estimating local geometric information from neighboring points, is critically dependent on the quality and spatial distribution of the local point set. In sparse, unstructured, or geometrically ambiguous regions, this estimation can become unreliable, thereby degrading the accuracy of the entire registration process [19], [20]. This vulnerability reveals two coupled research gaps: the need for an adaptive scan processing module that can adapt its surface covariance estimation strategy to local data quality, and the need for a map representation that can model complex geometries beyond a single surface. Our work addresses these challenges directly by an information-theoretic adaptive module to ensure reliable covariance estimation across diverse environments and introducing a GMM-based map to represent multi-modal surfaces.

Map representation in LIO is a critical trade-off between geometric fidelity and computational efficiency. To manage the immense volume of points and enable efficient nearest-neighbor searches, these methods typically rely on spatial data structures like k-d trees, with modern variants like the ikd-Tree [21] supporting incremental updates. While efficient, these methods lack explicit geometric priors, requiring local surface properties to be inferred on-the-fly. To embed geometric information directly into the map, a significant body of research has shifted towards structured, voxel-based representations. These methods model the environment using geometric primitives, such as surfels [10], planes [11], or single Gaussian distributions [5], [13], [22]. These approaches mark a significant step forward by embedding geometric priors directly into the map. However, their core limitation lies in the single-distribution-per-voxel assumption. This simplification inherently fails to represent geometric discontinuities, such as corners or intersecting surfaces, where a single voxel may contain points from multiple distinct planes. Our work directly addresses this limitation by introducing a dynamic Gaussian Mixture Model map, enabling a high-fidelity representation of such complex, multi-modal geometries.

III. Methodology

A. System Overview and Problem Formulation

The proposed GMM-LIO is a robust and adaptive tightly-coupled LIO system that achieves high-precision state estimation through three synergistic innovations, as depicted in Fig. 2. The framework is built upon: (1) a perceptive, information-theoretic front-end that adaptively modulates voxel resolution and surface covariance estimation based on state uncertainty and local point density (Sec. III-C); (2) a dynamic Gaussian Mixture Model voxel map for high-fidelity representation of complex geometries such as intersecting surfaces (Sec. III-D); and (3) a robust MAP-based estimator that employs an IRLS solver with a principled anisotropic information matrix to precisely mitigate the influence of outliers (Sec. III-E).

The state of the robot body (IMU frame, $\mathcal{B}$) at the time of the k-th LiDAR scan is defined as:

$$\mathbf{x}_k = \begin{bmatrix} {}^{\mathcal{W}}\mathbf{R}_{\mathcal{B}_k} & {}^{\mathcal{W}}\mathbf{t}_{\mathcal{B}_k} & {}^{\mathcal{W}}\mathbf{v}_{\mathcal{B}_k} & \mathbf{b}_{\boldsymbol{\omega},k} & \mathbf{b}_{\mathbf{a},k} \end{bmatrix}^T, \quad (1)$$

where ${}^{\mathcal{W}}\mathbf{R}_{\mathcal{B}_k} \in SO(3)$, ${}^{\mathcal{W}}\mathbf{t}_{\mathcal{B}_k} \in \mathbb{R}^3$, and ${}^{\mathcal{W}}\mathbf{v}_{\mathcal{B}_k} \in \mathbb{R}^3$ are the rotation, translation, and velocity of the body frame in the world frame $\mathcal{W}$. The terms $\mathbf{b}_{\boldsymbol{\omega},k}, \mathbf{b}_{\mathbf{a},k} \in \mathbb{R}^3$ represent the gyroscope and accelerometer biases, respectively. Our filter operates on an error state $\delta\mathbf{x} \in \mathbb{R}^{15}$, defined as the error between the updated state $\bar{\mathbf{x}}_k$ and its prediction $\widehat{\mathbf{x}}_k$.

When a new LiDAR scan $\mathcal{P}_k$ is available, the measurement update step refines the prior state estimate $\widehat{\mathbf{x}}_k$ to a posterior

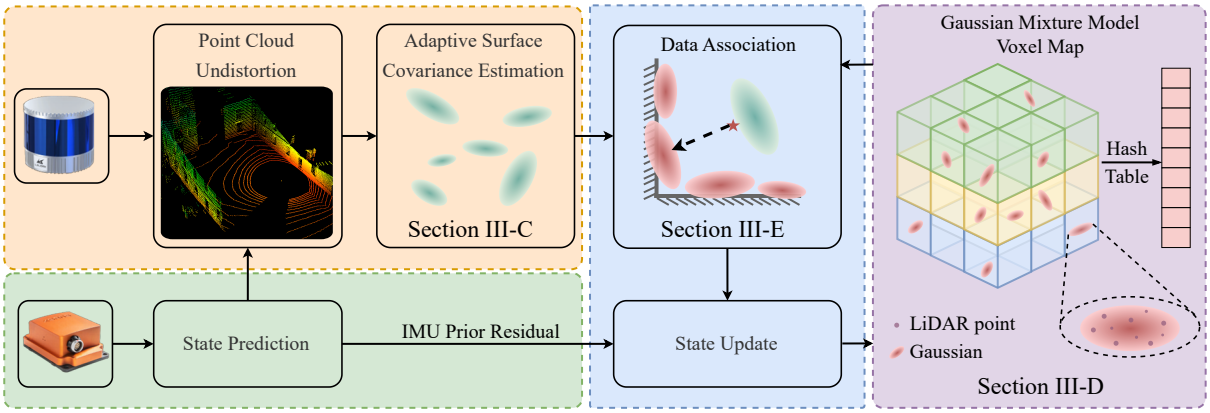


Fig. 2: System overview of GMM-LIO. An adaptive front-end registers scans against the GMM map. This geometric constraint is fused with IMU data within an iterative MAP-based estimator to update the system state.

state $\bar{\mathbf{x}}_k$. This refinement is formulated as a nonlinear Maximum a Posteriori problem. The goal is to find the optimal error state $\delta\mathbf{x}^*$ that minimizes the sum of the Mahalanobis distances of the prior and measurement residuals. At the κ-th iteration of the Gauss-Newton solver, this objective function is:

$$\delta\mathbf{x}^* = \arg\min_{\delta\mathbf{x}} \left\{ \|\delta\mathbf{x}\|^2_{\widehat{\mathbf{P}}_k^{-1}} + \sum_{i\in\mathcal{P}_k} \|r_{\mathcal{L}}(\widehat{\mathbf{x}}_k^{\kappa} \boxplus \delta\mathbf{x}, \mathbf{p}_i)\|^2_{\mathbf{\Sigma}_{\mathcal{L},i}} \right\}, \tag{2}$$

where the notation $\|\mathbf{a}\|^2_{\mathbf{M}} \triangleq \mathbf{a}^T\mathbf{M}^{-1}\mathbf{a}$. The cost function fuses the **IMU prior residual**, $\|\delta\mathbf{x}\|^2_{\widehat{\mathbf{P}}_k^{-1}}$, which penalizes deviations from the predicted state, with the **LiDAR measurement residual**, $r_{\mathcal{L}}(\cdot)$. The latter is derived from a GICP formulation [4], quantifying the distribution-to-distribution alignment error between the current scan and the local map. Upon convergence, the posterior state is updated as $\bar{\mathbf{x}}_k = \widehat{\mathbf{x}}_k \boxplus \delta\mathbf{x}^*$.

B. State Prediction

This module propagates the system state $\bar{\mathbf{x}}_{k-1}$ and its covariance $\bar{\mathbf{P}}_{k-1}$ forward in time from t_{k-1} to t_k using high-frequency IMU measurements. The nominal state is propagated by integrating the bias-corrected IMU data via a midpoint integration scheme to compute the prior state $\widehat{\mathbf{x}}_k$. Concurrently, the error-state covariance is propagated using a first-order discretization of the continuous-time dynamics. The discrete-time update between two IMU measurements is:

$$\mathbf{P}_{i+1} = \mathbf{F}_i\mathbf{P}_i\mathbf{F}_i^T + \mathbf{G}_i\mathbf{Q}\mathbf{G}_i^T, \tag{3}$$

where $\mathbf{F}_i$ and $\mathbf{G}_i$ are the discretized state transition and noise input Jacobians, and $\mathbf{Q}$ is the IMU noise covariance. This propagation, which starts with $\mathbf{P}_0 = \bar{\mathbf{P}}_{k-1}$, is chained over all IMU measurements to yield the final prior covariance $\widehat{\mathbf{P}}_k$. The detailed formulation follows established on-manifold integration methods [1].

C. Adaptive Information-Theoretic Covariance Estimation

This module processes each incoming LiDAR scan, $\mathcal{P}_k$, by partitioning its points using a standard voxel grid. The purpose of this grouping is to facilitate the estimation of local surface distributions, which are essential for the subsequent registration step.

1) Information-Driven Voxelization: The first adaptation level predictively modulates the voxel resolution, v_k, for the current scan based on the geometric constraint quality derived from the previous posterior state estimate. The objective is to define an appropriate spatial scale for grouping points to estimate a single, representative covariance for that region, rather than merely downsampling the data.

Specifically, we define an information sufficiency metric, $S_{k-1} \in (0, 1)$, derived from the LiDAR factor's Hessian matrix, $\mathbf{H}_{\mathcal{L}}^{(k-1)}$, from the prior update step. The trace of the Hessian's rotational, $\mathrm{Tr}(\mathbf{H}_{rr})$, and translational, $\mathrm{Tr}(\mathbf{H}_{tt})$, sub-blocks provide scalar measures of the available geometric information. We normalize these into sufficiency scores using a sigmoid function:

$$S_{\text{subspace}} = \frac{1}{1 + \exp(-\alpha(\mathrm{Tr}(\mathbf{H}_{\text{subspace}}) - \beta_{\text{subspace}}))}, \tag{4}$$

where 'subspace' is either 'rot' or 'trans'. Here, $\alpha > 0$ is a slope parameter that controls the steepness of the mapping, and β_{subspace} is an information threshold. The final metric, $S_{k-1} = \min(S_{\text{rot}}, S_{\text{trans}})$, reflects the weakest link in the geometric constraints. This metric then interpolates the voxel size, v_k, between a predefined minimum, $v_{\min}$, and maximum, $v_{\max}$:

$$v_k = v_{\min} + (v_{\max} - v_{\min}) \cdot S_{k-1}^{\gamma}. \tag{5}$$

The exponent $\gamma > 0$ is a tuning factor that controls the nonlinearity of the interpolation. Consequently, when geometric constraints are weak ($S_{k-1} \to 0$), the voxel resolution is dynamically decreased ($v_k \to v_{\min}$) to preserve fine details critical for registration in feature-sparse scenes. Conversely, when constraints are strong ($S_{k-1} \to 1$), the resolution is increased ($v_k \to v_{\max}$), allowing for more spatial aggregation to improve computational efficiency without sacrificing robustness.

2) Density-Adaptive Surface Covariance Estimation: Once the scan is discretized into voxels of adaptive size v_k, the second level of adaptation occurs during the estimation of the surface covariance. This process bifurcates based on the local point cloud density, addressing the instability that

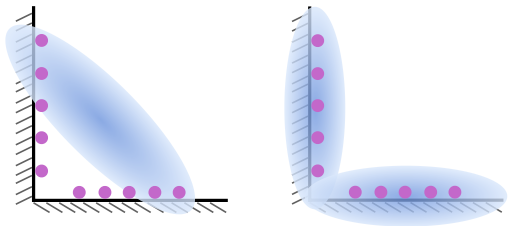
arises when a voxel contains too few points to form a reliable distribution [5]. Let N_p be the number of raw points within a given voxel and N_{min} be the minimum number of points required for a stable intra-voxel covariance estimation.

For locally dense voxels where $N_p \geq N_{min}$, the surface covariance is computed *exclusively* from this intra-voxel point set. This approach yields a highly accurate, localized geometric description, as the points themselves are sufficient to define the local surface without ambiguity.

For sparsely populated voxels where $N_p < N_{min}$, a stable covariance cannot be reliably estimated from the points alone. In this case, the system falls back to a hierarchical neighborhood search strategy. It aggregates points from voxels in a small, 1-hop ($3 \times 3 \times 3$) neighborhood. If the total number of points remains insufficient, the search expands to a larger, 2-hop ($5 \times 5 \times 5$) neighborhood. The covariance is then computed from this aggregated set of points. This adaptive search ensures that a stable and reasonable covariance can be estimated even in sparse regions of the scan, thereby enhancing the robustness of the GICP registration. This dual-strategy approach ensures a fine balance between geometric fidelity in dense areas and estimation robustness in sparse ones.

D. Gaussian Mixture Model Voxel Map

To accurately model complex geometries like corners and edges, critical for robust localization, we propose a map structure based on a Gaussian Mixture Model where each voxel can represent multiple intersecting surfaces. This approach fundamentally overcomes the limitations of single-Gaussian voxel representations that inherently fail to capture such features. The conceptual advantage of our approach is visually illustrated in Fig. 3, where a single-Gaussian model smears out an intersecting corner, whereas our GMM precisely models the distinct surfaces.



(a) Single-Gaussian (b) Our GMM Voxel Map

Fig. 3: A conceptual comparison of map representations for modeling a geometric corner. (a) A single-Gaussian model attempts to fit points from two intersecting surfaces into a single, smeared distribution, leading to a loss of geometric fidelity. (b) Our GMM Voxel Map accurately represents the two distinct surfaces using two separate Gaussian components, preserving the sharp geometric feature.

1) Representation of the GMM Map: The global map $\mathcal{M}$ is a hash table mapping a 3D spatial index to a voxel $\mathcal{V}_i$. Each voxel models local geometry as a GMM with $K_i \in \{1, 2\}$ components, sufficient for common intersecting planar structures (e.g., wall-floor, wall-wall intersections) while maintaining tractability. A voxel $\mathcal{V}_i$ is defined as:

$$\mathcal{V}_i = \langle K_i, \{\mathcal{G}_{i,k}\}_{k=1}^{K_i}, \mathcal{P}_i \rangle, \tag{6}$$

where K_i is the number of active Gaussian components, $\{\mathcal{G}_{i,k}\}$ is the set of these components, and $\mathcal{P}_i$ is the set of raw points within the voxel, retained for dynamic updates.

Each Gaussian component $\mathcal{G}_{i,k}$ is defined by its parameters, including those required for efficient incremental updates:

$$\mathcal{G}_{i,k} = \langle N_{i,k}, \mathbf{S}_{i,k}, \mathbf{C}_{i,k}, \boldsymbol{\mu}_{i,k}, \boldsymbol{\Sigma}_{i,k}^{\mathcal{M}} \rangle. \tag{7}$$

Here, $N_{i,k}$ is the number of points associated with the component. $\mathbf{S}_{i,k} = \sum \mathbf{p}$ is the sum of the points' positions, and $\mathbf{C}_{i,k} = \sum \mathbf{p}\mathbf{p}^T$ is the sum of their outer products. From these sums, the mean $\boldsymbol{\mu}_{i,k}$ and covariance $\boldsymbol{\Sigma}_{i,k}$ are computed:

$$\boldsymbol{\mu}_{i,k} = \frac{1}{N_{i,k}} \mathbf{S}_{i,k}, \tag{8}$$

$$\boldsymbol{\Sigma}_{i,k} = \frac{1}{N_{i,k} - 1} \left(\mathbf{C}_{i,k} - \frac{\mathbf{S}_{i,k}\mathbf{S}_{i,k}^T}{N_{i,k}} \right). \tag{9}$$

To enforce a planar model consistent with the GICP framework, the computed covariance $\boldsymbol{\Sigma}_{i,k}$ is regularized. We perform an SVD, $\boldsymbol{\Sigma}_{i,k} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^T$, and then reconstruct the final map covariance $\boldsymbol{\Sigma}_{i,k}^{\mathcal{M}}$ with modified singular values:

$$\boldsymbol{\Sigma}_{i,k}^{\mathcal{M}} = \mathbf{U} \cdot \text{diag}(1, 1, \epsilon) \cdot \mathbf{V}^T, \tag{10}$$

where ϵ is a small value (e.g., 10^{-3}) to model a thin planar distribution. The probability density of a point $\mathbf{p}$ in this voxel is the weighted sum of its components:

$$p(\mathbf{p}|\mathcal{V}_i) = \sum_{k=1}^{K_i} w_{i,k} \mathcal{N}(\mathbf{p}|\boldsymbol{\mu}_{i,k}, \boldsymbol{\Sigma}_{i,k}^{\mathcal{M}}), \quad w_{i,k} = \frac{N_{i,k}}{\sum_j N_{i,j}}. \tag{11}$$

2) Incremental Update and Splitting: The GMM map is updated incrementally with each new scan. For a single-component voxel ($K_i = 1$), the arrival of a new set of points $\mathcal{P}_{new}$ triggers a direct update of its sufficient statistics:

$$\mathbf{S}_{i,1} \leftarrow \mathbf{S}_{i,1} + \sum_{\mathbf{p} \in \mathcal{P}_{new}} \mathbf{p}, \tag{12}$$

$$\mathbf{C}_{i,1} \leftarrow \mathbf{C}_{i,1} + \sum_{\mathbf{p} \in \mathcal{P}_{new}} \mathbf{p}\mathbf{p}^T, \tag{13}$$

$$N_{i,1} \leftarrow N_{i,1} + |\mathcal{P}_{new}|. \tag{14}$$

After this update, a splitting procedure is considered to determine if the single Gaussian component should be partitioned into two. This procedure is initiated only if three conditions are met: first, the total number of points in the voxel, $|\mathcal{P}_i|$, must exceed a predefined `split_threshold`; second, the voxel must currently be represented by a single component ($K_i = 1$); and third, the geometry must not be distinctly planar. Planarity is evaluated using the eigenvalues ($\lambda_1 \geq \lambda_2 \geq \lambda_3$) of the point set's covariance; a split is triggered if the metric $\lambda_2/\lambda_1 < \tau_p$, indicating low planarity.

If triggered, the points $\mathcal{P}_i$ are partitioned into two planar clusters, $\mathcal{C}_0$ and $\mathcal{C}_1$, using the method in Alg. 1. The split is

validated by ensuring centroids are separated ($\|\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1\| > d_{min}$) and each cluster has enough points ($|\mathcal{C}_k| > N_{min}$). If validated, the original component is replaced by two new ones, and K_i is set to 2.

Algorithm 1 Plane K-Means Clustering

Input: Point set $\mathcal{P}$
Initialize: Use K-Means++ to select two initial centroids $\boldsymbol{\mu}_0, \boldsymbol{\mu}_1$. Set initial normals $\mathbf{n}_0, \mathbf{n}_1$ (e.g., as random unit vectors).
repeat
  Set `has_changed` $\leftarrow$ false
  // – Assignment Step –
  for each point $\mathbf{p}_j \in \mathcal{P}$ **do**
    $d_0 \leftarrow |\mathbf{n}_0^T(\mathbf{p}_j - \boldsymbol{\mu}_0)|$
    $d_1 \leftarrow |\mathbf{n}_1^T(\mathbf{p}_j - \boldsymbol{\mu}_1)|$
    $k_j \leftarrow \arg\min_{k\in\{0,1\}} d_k$
    if assignment of $\mathbf{p}_j$ changed **then**
      set `has_changed` $\leftarrow$ true
    end if
    Assign $\mathbf{p}_j$ to cluster $\mathcal{C}_{k_j}$
  end for
  // – Update Step –
  for each cluster $k \in \{0, 1\}$ **do**
    $\boldsymbol{\mu}_k \leftarrow \frac{1}{|\mathcal{C}_k|}\sum_{\mathbf{p}\in\mathcal{C}_k} \mathbf{p}$
    $\mathbf{Cov}_k \leftarrow \mathrm{Cov}(\mathcal{C}_k)$
    $\mathbf{n}_k \leftarrow$ eigenvector corresponding to the minimum eigenvalue of $\mathbf{Cov}_k$
  end for
until `has_changed` is false or max iterations reached
Output: Clusters $\mathcal{C}_0, \mathcal{C}_1$ and plane parameters $(\boldsymbol{\mu}_0, \mathbf{n}_0), (\boldsymbol{\mu}_1, \mathbf{n}_1)$

E. MAP-based State Update

The state is estimated by iteratively solving the MAP problem (Eq. 2) to fuse the IMU prior with LiDAR measurements.

1) Data Association: The LiDAR measurement model is founded on the GICP distribution-to-distribution metric. For each downsampled scan point $\mathbf{p}_i$ with its surface covariance $\mathbf{C}_i^{\mathcal{L}}$, a correspondence is established with a component in a map voxel $\mathcal{V}_j$. As detailed in Sec. IV-C, if $\mathcal{V}_j$ is a GMM, a robust data association is performed by selecting the Gaussian component that has the smallest Mahalanobis distance to the scan point $\mathbf{p}_i$. This process yields the corresponding map distribution $\mathcal{N}(\boldsymbol{\mu}_j^{\mathcal{M}}, \mathbf{C}_j^{\mathcal{M}})$.

The residual vector $\mathbf{r}_i$ for this correspondence is defined as the difference between the map component's mean and the transformed scan point's position:

$$\mathbf{r}_i(\mathbf{x}_k) = \boldsymbol{\mu}_j^{\mathcal{M}} - ({}^{\mathcal{W}}\mathbf{R}_{\mathcal{B}_k}\mathbf{p}_i + {}^{\mathcal{W}}\mathbf{t}_{\mathcal{B}_k}). \tag{15}$$

The corresponding GICP covariance $\boldsymbol{\Sigma}_i$ is the sum of the map covariance and the rotated scan point covariance:

$$\boldsymbol{\Sigma}_i(\mathbf{R}_k) = \mathbf{C}_j^{\mathcal{M}} + {}^{\mathcal{W}}\mathbf{R}_{\mathcal{B}_k}\mathbf{C}_i^{\mathcal{L}}({}^{\mathcal{W}}\mathbf{R}_{\mathcal{B}_k})^T. \tag{16}$$

To handle outliers arising from incorrect correspondences or dynamic objects, we employ a robust kernel $\rho(\cdot)$ on the squared Mahalanobis distance $s_i = \mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{r}_i$. This transforms the optimization into an IRLS problem.

2) State Update: The total MAP objective is to find the error state $\delta\mathbf{x}^*$ that minimizes the sum of the Mahalanobis distances of the prior and measurement residuals. Within each Gauss-Newton iteration, this is achieved by solving a linear system that approximates the original nonlinear problem. To achieve a more principled robust estimation, our IRLS implementation goes beyond a simple scalar reweighting of residuals. We construct an adaptive information matrix $\boldsymbol{\Omega}_{i,\text{robust}}$ derived from the second-order approximation of the robust cost, involving both the first and second derivatives of the kernel $\rho(\cdot)$ (see Appendix A for the full derivation):

$$\boldsymbol{\Omega}_{i,\text{robust}} = \rho'(s_i)\boldsymbol{\Sigma}_i^{-1} + 2\rho''(s_i)(\boldsymbol{\Sigma}_i^{-1}\mathbf{r}_i)(\mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}). \tag{17}$$

Critically, unlike standard IRLS implementations that typically use only the first derivative, $\rho'(s_i)$, for isotropic (uniform) down-weighting of outliers, our formulation incorporates the second derivative, $\rho''(s_i)$. This inclusion introduces a rank-1 update term that provides a more sophisticated, anisotropic adjustment. Specifically, it reduces the information precisely along the direction of the residual vector $\mathbf{r}_i$ itself. This rigorous approach ensures that the influence of outliers is suppressed with greater accuracy, significantly enhancing the final robustness of the state estimate.

The total robust Hessian of the LiDAR factor, $\mathbf{H}_{\mathcal{L},\text{robust}}$, is constructed by summing the contributions of all correspondences, each weighted by this adaptive information matrix:

$$\mathbf{H}_{\mathcal{L},\text{robust}} = \sum_i \mathbf{J}_i^T\boldsymbol{\Omega}_{i,\text{robust}}\mathbf{J}_i, \tag{18}$$

where $\mathbf{J}_i = \frac{\partial\mathbf{r}_i}{\partial\delta\mathbf{x}}$ is the Jacobian of the residual with respect to the error state. Similarly, the robust LiDAR gradient vector, $\mathbf{b}_{\mathcal{L},\text{robust}}$, is the sum of the weighted gradients:

$$\mathbf{b}_{\mathcal{L},\text{robust}} = -\sum_i \rho'(s_i)\mathbf{J}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{r}_i. \tag{19}$$

Finally, the optimal error state increment $\delta\mathbf{x}^*$ is found by solving the full linear system, which incorporates the IMU prior information ($\mathbf{H}_{\text{prior}} = \widehat{\mathbf{P}}_k^{-1}$ and its corresponding gradient $\mathbf{b}_{\text{prior}}$):

$$(\mathbf{H}_{\mathcal{L},\text{robust}} + \mathbf{H}_{\text{prior}})\delta\mathbf{x}^* = \mathbf{b}_{\mathcal{L},\text{robust}} + \mathbf{b}_{\text{prior}}. \tag{20}$$

Upon convergence of the Gauss-Newton iterations, the optimal error state is used to correct the prior state to yield the final posterior estimate: $\bar{\mathbf{x}}_k = \widehat{\mathbf{x}}_k \boxplus \delta\mathbf{x}^*$.

IV. Experiments

A. Experimental Setup

We evaluated our system on four challenging public benchmark datasets and one self-collected dataset. These public datasets include the **M2DGR Dataset** [18], a large-scale dataset for ground robots with scenarios explicitly designed to challenge modern SLAM systems. We also used the

Newer College Dataset [17], a diverse dataset collected with an Ouster LiDAR, featuring aggressive motions and varied environments, ideal for general performance evaluation. We further included the **GRACO Dataset** [23], a multimodal dataset for ground and aerial cooperative localization and mapping, and the **BotanicGarden Dataset** [24], a high-quality dataset for robot navigation in unstructured natural environments. Additionally, we utilized a self-collected dataset equipped with a RoboSense LiDAR and Xsens IMU, which features a cluttered environment with dynamic pedestrians.

We compare GMM-LIO against several state-of-the-art open-source LIO systems: FAST-LIO2 [2], LIO-SAM [3], Faster-LIO [9], DLIO [7] and AKF-LIO [16]. All experiments were performed on a laptop with an Intel Core i7-11800H CPU and 16GB of RAM. We use the `evo` package [25] for quantitative evaluation.

B. Comparative Evaluation on Diverse Datasets

To comprehensively evaluate the performance of our proposed GMM-LIO framework, we present a quantitative and qualitative comparison against several state-of-the-art methods. The objective is to demonstrate the superior accuracy and robustness of our approach across a wide range of challenging and diverse environments.

Table I presents a quantitative comparison of the Absolute Trajectory Error (ATE) RMSE in meters. The evaluation spans 18 sequences from five datasets, each chosen to probe different aspects of algorithmic performance. The results clearly indicate that GMM-LIO achieves a superior level of accuracy, demonstrating an average relative improvement of 36% over the FAST-LIO2 baseline across all sequences. This strength stems from different core components tailored to specific environmental challenges.

In structured urban environments such as **M2DGR** and **GRACO**, GMM-LIO exhibits excellent accuracy. This is primarily attributed to our GMM-based map, which provides a high-fidelity representation of sharp geometric features like corners and intersecting planes, offering superior constraints for registration compared to single-Gaussian voxel models. This advantage is particularly evident in sequences like `M2DGR-Street05` and `GRACO-G03`, where our method achieves the lowest error.

Conversely, in the unstructured and geometrically ambiguous sequences of the **BotanicGarden** dataset, the system's leading performance stems from a different set of strengths. Here, where distinct planar structures are scarce, the adaptive front-end excels by dynamically adjusting its processing strategy to the sparse and irregular point distributions. Concurrently, the principled robust estimator effectively mitigates noise from environmental clutter such as foliage. This synergy allows GMM-LIO to significantly outperform other methods in these challenging natural scenes.

Furthermore, in the **Newer College Dataset** sequences involving aggressive motions (e.g., `ncd05`, `ncd06`), the combination of the adaptive front-end, which handles rapid changes in point density, and the robust estimator, which down-weights motion-induced outliers, ensures stable and accurate tracking where other methods falter. Our method consistently delivers the lowest ATE in these high-dynamics tests.

Fig. 4 provides a qualitative visualization of the estimated trajectories, which corroborates the quantitative results. The trajectories from GMM-LIO are visibly more consistent with the ground truth. Additionally, Fig. 5 showcases the high-fidelity mapping result on our self-collected dataset, demonstrating the system's dual strengths in both accurate localization and high-quality map generation in real-world conditions.

C. Ablation Study

To validate the individual contribution of each key component in our framework, we conducted a thorough ablation study. We compare the full GMM-LIO system against three degraded versions: (1) **w/o Robust GICP**, which uses a standard GICP cost function with a simple robust kernel; (2) **w/o Adaptive Front-end**, which employs fixed parameters for voxel resolution and covariance search; and (3) **w/o GMM Map**, which replaces our GMM representation with a simpler single-Gaussian-per-voxel map.

The quantitative results in Table II highlight how each component contributes to robustness in various scenarios. Specifically, the impact of the GMM Map is most pronounced in structured environments like M2DGR and GRACO, confirming its critical role in accurately modeling complex geometric features. Conversely, the Adaptive Front-end proves most essential in the unstructured BotanicGarden sequence, where its ability to handle sparse and irregular point distributions is paramount. Finally, removing the Robust GICP formulation consistently degrades performance across all datasets, particularly in the dynamic M2DGR scenario, validating its effectiveness in mitigating outliers. This analysis confirms that the components work synergistically, each addressing distinct environmental challenges to achieve the full system's superior performance. The error distribution is shown in Fig. 6.

D. Real-Time Performance Analysis

To demonstrate the system's efficiency, we provide a detailed breakdown of the average processing time per scan, with all timings measured on an Intel Core i7-11800H CPU. Table III presents this analysis, where the total time is decomposed into three main modules: the *Front-end*, which includes motion correction and our adaptive covariance estimation; the *State Update*, encompassing the core GICP scan-to-map alignment and the MAP-based IRLS solver; and the *Map Update* for the incremental update of the GMM Voxel Map.

The average total time is approximately 14.04 ms, confirming the system's real-time capability for typical 10Hz LiDAR sensors. The performance varies logically with sensor configuration and environmental complexity. The highest computational load (25.87 ms) is observed in the **Newer College** dataset. This can be attributed to two main factors:

TABLE I: Quantitative comparison against state-of-the-art methods in terms of Absolute Trajectory Error (ATE) RMSE [m].

Method	M2DGR				Newer College Dataset				BotanicGarden		Self-Collected		GRACO					
	Street03	Street04	Street05	Street06	ncd01[a]	ncd05	ncd06	ncd07	1006-01	1008-03	S01	S02	G01	G02	G03	G04	G05	G06
FAST-LIO2	0.198	0.443[b]	0.378	0.413	0.353	0.125	0.079	**0.127**	1.911	1.673	0.676	0.623	0.428	1.004	1.610	0.633	1.899	2.351
LIO-SAM	0.192	1.022	0.399	0.417	0.468	0.127	0.121	0.173	–	–	0.505	0.694	0.316	0.832	1.125	0.729	1.980	0.922
Faster-LIO	0.498	1.182	0.588	0.457	0.340	0.119	0.079	0.141	1.697	2.086	0.587	0.814	0.663	0.762	0.734	0.715	**0.655**	0.645
DLIO	**0.141**[b]	0.629	1.272	**0.412**	0.361	0.105	0.121	0.157	1.798	2.245	0.563	0.681	0.346	0.788	0.832	**0.492**	2.021	0.720
AKF-LIO	–[c]	–	–	–	0.325	0.118	0.085	0.140	2.017	1.748	0.713	0.877	0.740	1.888	6.485	1.202	2.465	4.238
GMM-LIO	0.173	**0.426**	**0.269**	0.437	**0.323**	**0.100**	**0.077**	0.134	**1.406**	**1.489**	**0.463**	**0.594**	**0.284**	**0.761**	**0.718**	0.516	0.761	**0.614**

[a] The sequence names are abbreviated as follows: ncd01 denotes `01_short_experiment`, ncd05 denotes `05_quad_with_dynamics`, ncd06 denotes `06_dynamic_spinning`, and ncd07 denotes `07_parkland_mound`. [b] The best results are highlighted in **bold**, while the second-best results are underlined. [c] A double dash "–" indicates that the corresponding method failed to provide a valid trajectory for the sequence.

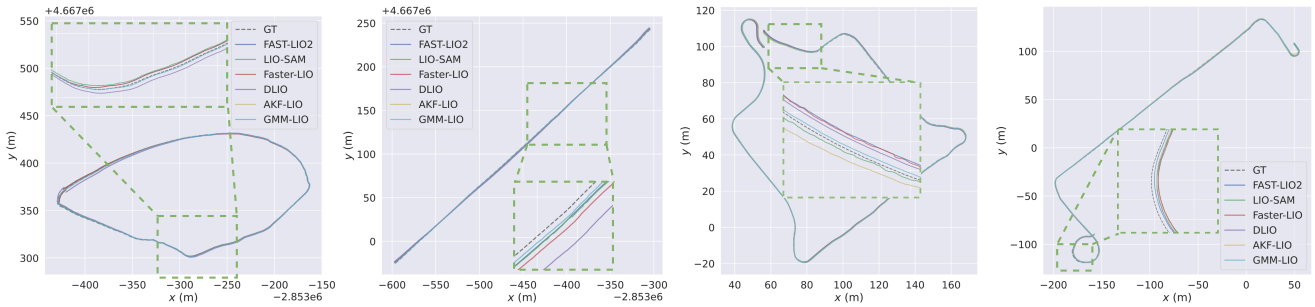


Fig. 4: A qualitative comparison of estimated trajectories from various methods (colored lines) against the ground-truth trajectories (black dashed lines). From left to right, the subfigures show results on the M2DGR-Street04, M2DGR-Street05, GRACO-G01, and GRACO-G02 sequences. Our proposed GMM-LIO demonstrates superior consistency and accuracy.

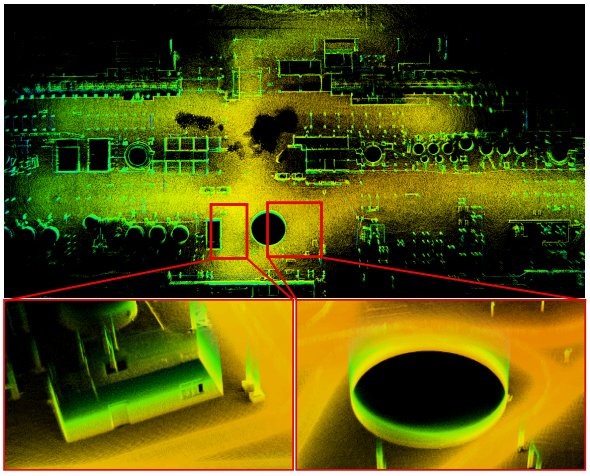


Fig. 5: Mapping results from GMM-LIO on a challenging, self-collected dataset. The figure shows the generated map and its magnified views.

TABLE II: Ablation study of GMM-LIO components on ATE [m].

Configuration	M2DGR (Street04)	Newer College (ncd06)	BotanicGarden (1008-03)	GRACO (G01)
w/o Robust GICP	0.447	0.078	1.518	0.289
w/o Adaptive Front-end	0.558	0.078	1.584	0.295
w/o GMM Map	0.580	0.085	1.490	0.310
GMM-LIO (Full)	**0.426**	**0.077**	**1.489**	**0.284**

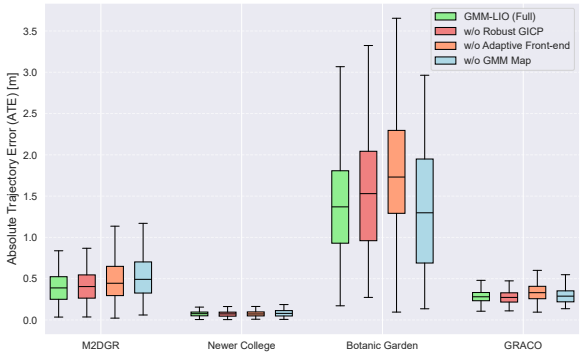


Fig. 6: Error distribution from the ablation study.

TABLE III: Average Processing Time per Scan [ms].

Sequence	Front-end	State Update	Map Update	Total ↓
M2DGR	7.36	1.11	3.49	11.96
Newer College	10.81	4.42	10.64	25.87
BotanicGarden	3.57	2.04	1.68	7.29
GRACO	5.49	1.81	3.72	11.02
Average	6.81	2.35	4.13	14.04

firstly, the higher-resolution LiDAR used in this dataset generates a denser point cloud per scan; secondly, the prevalence of structured features, such as intersecting planes at building corners, frequently activates the computationally intensive splitting mechanism within our GMM map. Conversely, the system is most efficient (7.29 ms) in the unstructured **BotanicGarden** dataset, which features sparser geometry. This analysis confirms that the enhanced accuracy and robustness of GMM-LIO are achieved without sacrificing real-time performance.

V. Conclusion

In this paper, we presented GMM-LIO, a tightly-coupled LiDAR-inertial odometry framework that enhances robustness and precision through the synergy of three key innovations: an information-theoretic adaptive front-end for surface covariance estimation, a high-fidelity Gaussian Mixture Model voxel map for complex geometries, and a principled robust MAP estimator to mitigate outliers. Extensive evaluations on challenging public datasets demonstrated that GMM-LIO achieves state-of-the-art accuracy and robustness, particularly in geometrically degenerate and dynamic scenarios. Future work will focus on exploring the fusion of visual

data for richer metric-semantic mapping.

APPENDIX

A. Derivation of the Robust Information Matrix

The robust cost for a single GICP correspondence is given by $\frac{1}{2}\rho(s_i)$, where $s_i = \mathbf{r}_i^T \boldsymbol{\Sigma}_i^{-1} \mathbf{r}_i$ is the squared Mahalanobis distance. To solve this non-linear least squares problem using the Gauss-Newton method, we need to find a quadratic approximation of the cost function with respect to a small state increment $\delta\mathbf{x}$.

We begin by analyzing the incremental cost, ΔC_i, using a second-order Taylor expansion of the robust kernel $\rho(\cdot)$:

$$\Delta C_i = \frac{1}{2}\rho(s_i+\Delta s_i) - \frac{1}{2}\rho(s_i) \approx \frac{1}{2}\rho'(s_i)\Delta s_i + \frac{1}{4}\rho''(s_i)(\Delta s_i)^2. \tag{21}$$

Next, we find an expression for the change in the squared error, Δs_i. By linearizing the residual vector around the current state estimate, $\mathbf{r}_i(\mathbf{x} \boxplus \delta\mathbf{x}) \approx \mathbf{r}_i + \mathbf{J}_i\delta\mathbf{x}$, where $\mathbf{J}_i$ is the Jacobian of the residual with respect to the error state, we get:

$$\begin{aligned}\Delta s_i &= (\mathbf{r}_i + \mathbf{J}_i\delta\mathbf{x})^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{r}_i + \mathbf{J}_i\delta\mathbf{x}) - s_i \\ &\approx \underbrace{2\mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{J}_i\delta\mathbf{x}}_{O(\delta\mathbf{x})} + \underbrace{(\delta\mathbf{x})^\top\mathbf{J}_i^\top\boldsymbol{\Sigma}_i^{-1}\mathbf{J}_i\delta\mathbf{x}}_{O(\delta\mathbf{x}^2)}.\end{aligned} \tag{22}$$

We now substitute Eq. (22) into Eq. (21). To form a valid quadratic approximation for the Gauss-Newton step, we keep all terms up to the second order in $\delta\mathbf{x}$. The term $(\Delta s_i)^2$ contains terms of $O(\delta\mathbf{x}^2)$ and higher. We only retain its second-order component: $(2\mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{J}_i\delta\mathbf{x})^2 = 4(\delta\mathbf{x})^T\mathbf{J}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{r}_i\mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{J}_i\delta\mathbf{x}$.

Substituting and collecting all terms, the incremental cost function becomes:

$$\begin{aligned}\Delta C_i(\delta\mathbf{x}) \approx &\underbrace{\rho'(s_i)\mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{J}_i}_{\text{gradient part}}\delta\mathbf{x} \\ &+ \frac{1}{2}(\delta\mathbf{x})^T\underbrace{\left(\rho'(s_i)\mathbf{J}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{J}_i + 2\rho''(s_i)\mathbf{J}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{r}_i\mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{J}_i\right)}_{\text{Hessian part}}\delta\mathbf{x}.\end{aligned} \tag{23}$$

This expression is a quadratic form, $\Delta C_i \approx \mathbf{b}_i^T\delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x}^T\mathbf{H}_i\delta\mathbf{x}$. The Hessian part is the term in the parentheses. By factoring out the Jacobians $\mathbf{J}_i^T$ and $\mathbf{J}_i$, we can identify the adaptive information matrix $\boldsymbol{\Omega}_{i,\text{robust}}$ that defines the weighting for this correspondence:

$$\boldsymbol{\Omega}_{i,\text{robust}} = \rho'(s_i)\boldsymbol{\Sigma}_i^{-1} + 2\rho''(s_i)(\boldsymbol{\Sigma}_i^{-1}\mathbf{r}_i)(\mathbf{r}_i^T\boldsymbol{\Sigma}_i^{-1}). \tag{24}$$

This allows us to write the Hessian contribution for the i-th correspondence in the familiar form $\mathbf{H}_{i,\text{robust}} = \mathbf{J}_i^T\boldsymbol{\Omega}_{i,\text{robust}}\mathbf{J}_i$.

REFERENCES

[1] W. Xu and F. Zhang, "Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3317–3324, 2021.

[2] 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.

[3] 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.

[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] 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.

[7] K. Chen, R. Nemiroff, and B. T. Lopez, "Direct lidar-inertial odometry: Lightweight lio with continuous-time motion correction," in *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2023, pp. 3983–3989.

[8] A. Reinke, M. Palieri, B. Morrell, Y. Chang, K. Ebadi, L. Carlone, and A.-A. Agha-Mohammadi, "Locus 2.0: Robust and computationally efficient lidar odometry for real-time 3d mapping," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9043–9050, 2022.

[9] 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.

[10] J. Behley and C. Stachniss, "Efficient surfel-based slam using 3d laser range data in urban environments." in *Robotics: Science and Systems*, vol. 2018, 2018, p. 59.

[11] 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.

[12] L. Wang, X. Zhang, C. Li, D. He, Y. Pan, and J. Yi, "Super-lio: A robust and efficient lidar-inertial odometry system with a compact mapping strategy," *IEEE Robotics and Automation Letters*, vol. 11, no. 3, pp. 2666–2673, 2026.

[13] 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*, 2024.

[14] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 809–816.

[15] T. Tuna, J. Nubert, Y. Nava, S. Khattak, and M. Hutter, "X-icp: Localizability-aware lidar registration for robust localization in extreme environments," *IEEE Transactions on Robotics*, 2023.

[16] X. Xie, R. Geng, J. Ma, and B. Zhou, "Akf-lio: Lidar-inertial odometry with gaussian map by adaptive kalman filter," *arXiv preprint arXiv:2503.06891*, 2025.

[17] M. Ramezani, Y. Wang, M. Camurri, D. Wisth, M. Mattamala, and M. Fallon, "The newer college dataset: Handheld lidar, inertial and vision with ground truth," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4353–4360.

[18] 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, 2022.

[19] 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*, 2024.

[20] J. Jiao, Y. Zhu, H. Ye, H. Huang, P. Yun, L. Jiang, L. Wang, and M. Liu, "Greedy-based feature selection for efficient lidar slam," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5222–5228.

[21] Y. Cai, W. Xu, and F. Zhang, "ikd-tree: An incremental kd tree for robotic applications," *arXiv preprint arXiv:2102.10808*, 2021.

[22] M. Yokozuka, K. Koide, S. Oishi, and A. Banno, "Litamin2: Ultra light lidar-based slam using geometric approximation applied with kl-divergence," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 11 619–11 625.

[23] Y. Zhu, Y. Kong, Y. Jie, S. Xu, and H. Cheng, "Graco: A multimodal dataset for ground and aerial cooperative localization and mapping," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 966–973, 2023.

[24] Y. Liu, Y. Fu, M. Qin, Y. Xu, B. Xu, F. Chen, B. Goossens, P. Z. Sun, H. Yu, C. Liu, L. Chen, W. Tao, and H. Zhao, "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.

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