

UAF-LIVO: Uncertainty-Aware Feature-Based LiDAR-Inertial-Visual Odometry Robust to Cross-Modal Geometric Inconsistency

Zhipeng Luo[✉], Graduate Student Member, IEEE, Mingnan Hu[✉], Zheming Wang[✉], Member, IEEE, and Bo Chen[✉], Senior Member, IEEE

Abstract—This letter presents UAF-LIVO, a feature-based sequential LiDAR-inertial-visual odometry system robust to cross-modal geometric inconsistency. While sequential fusion mitigates the inherent robustness imbalance between LiDAR and visual measurements, it remains highly susceptible to cross-modal misalignment. These discrepancies typically arise from imperfect sensor calibration and visual feature misalignment in dynamic or degraded environments. To address this, we propose a geometry-consistent visual measurement construction and update strategy. Matched features are categorized as local plane supported, voxel-map supported, or bearing-only. For each, we derive a thorough propagation model to explicitly quantify measurement uncertainty. This theoretical foundation allows UAF-LIVO to adaptively down-weight ill-conditioned observations and reject outliers within an IESKF framework. Experiments on public datasets with diverse sensor configurations and scenarios demonstrate that UAF-LIVO achieves competitive accuracy and consistent robustness against geometric inconsistency compared to state-of-the-art methods.

Index Terms—Localization, SLAM, sensor fusion.

I. INTRODUCTION

INERTIAL-ASSISTED odometry is primarily coupled with two complementary sensing modalities: vision and LiDAR. Visual-inertial odometry (VIO) fuses visual measurements with IMU propagation to achieve metrically scaled 6-DOF estimation [1], [2], but suffers from unreliable feature extraction in low-texture or complex illumination environments and requires sufficient parallax for robust initialization [3]. In contrast, LiDAR-inertial odometry (LIO) derives accurate metric geometry from stable geometric observations [4], [5], and recent advances, including high-bandwidth LiDAR odometry [6], invariant Kalman Filter-based update [7], and plane-centric

feature representations [8], have further improved its accuracy and robustness. However, LiDAR lacks appearance information and remains vulnerable in geometrically sparse or ambiguous scenes.

To exploit the complementary advantages of LiDAR and visual measurements, the LiDAR-inertial-visual odometry (LIVO) framework has been proposed [9], [10], [11], [12]. LIVO combines dense visual information with LiDAR's accurate geometric measurements, achieving robust pose estimation in low-texture, complex illumination, or geometrically challenging environments. On the visual side, LiDAR provides reliable metric information, simplifying the initialization process of the VIO system. On the LiDAR side, visual measurements provide color and texture information, enabling higher-level perception tasks such as object-level scene understanding [13]. These complementary modalities allow LIVO systems to maintain accuracy and robustness across diverse scenarios, and support downstream applications such as autonomous navigation [10] and high-fidelity environmental reconstruction with Gaussian Splatting [14].

Structurally, LIVO systems generally follow either a parallel or a sequential fusion strategy. Parallel fusion maintains the LiDAR and visual branches as two independent odometry threads, coupling them exclusively at the backend factor-graph optimization layer [12], [15], [16]. Because these subsystems inherently operate independently prior to joint optimization, the visual branch can be vulnerable to scale ambiguity. Consequently, visual tracking degrades more easily than the LiDAR branch in challenging scenarios, leading to system-level geometric inconsistency and potential estimation divergence. LIVER [15] improves robustness through a three-stage initialization, but the procedure is relatively complex and highlights the inherent difficulty of maintaining consistency in parallel fusion. In contrast, sequential fusion constructs visual measurements utilizing the LIO state estimation prior, thereby preventing the stringent tracking requirements of the visual subsystem from compromising overall stability [10].

Visual measurements in the LIVO framework can be broadly categorized into three types: feature-based [12], [16], [17], direct [10], [11], and hybrid approaches [9], [18]. Direct methods leverage photometric measurements to enable fast alignment but rely heavily on image gradients and photometric consistency, making them sensitive to optimization divergence under

Received 19 April 2026; accepted 12 June 2026. Date of publication 3 July 2026; date of current version 15 July 2026. This article was recommended for publication by Associate Editor H. Blum and Editor A. Kim upon evaluation of the reviewers' comments. This work was supported in part by the Joint Funds of the National Natural Science Foundation of China under Grant U24A20258, in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LR25F030001, in part by the Fundamental Research Funds for the Provincial Universities of Zhejiang under Grant RF-C2023007, and in part by the Funding of Leading Innovative and Entrepreneur Team Introduction Program of Zhejiang under Grant 2023R01006. (Corresponding author: Bo Chen.)

The authors are with the Department of Automation, Zhejiang University of Technology, Hangzhou 310014, China (e-mail: luozhipengac@gmail.com; huminant@163.com; wangzheming@zjut.edu.cn; bchen@zjut.edu.cn).

This article has supplementary downloadable material available at <https://doi.org/10.1109/LRA.2026.3710002>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2026.3710002

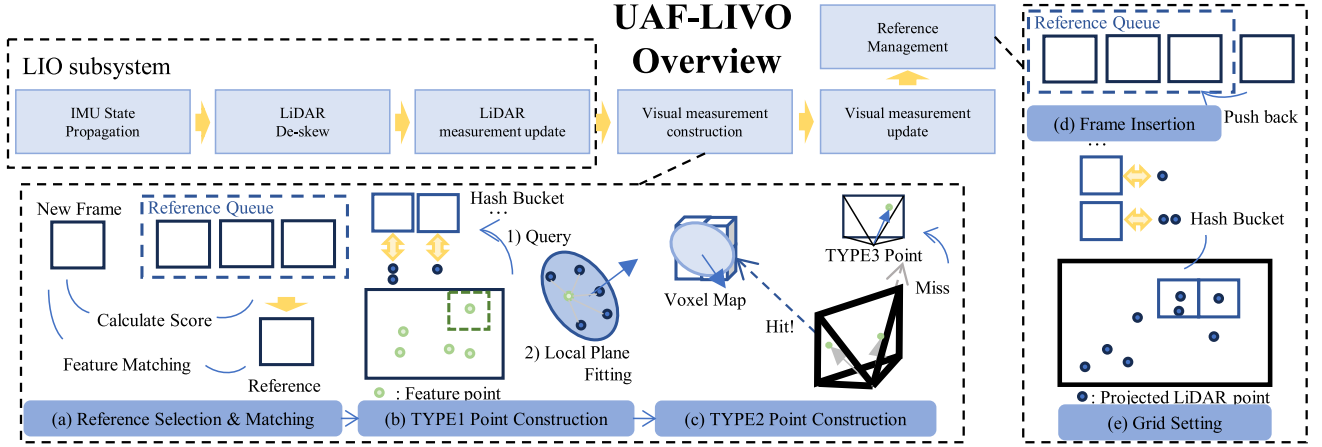


Fig. 1. Overview of the proposed UAF-LIVO system. (a) A reference frame is selected through a score-based strategy and is matched to the current visual frame. (b) For each visual feature, its corresponding hash bucket is queried and a local plane is fitted using neighboring LiDAR points. (c) Along the feature’s viewing ray, an intersection check with the plane parameterized voxel map is performed; features that fail to intersect any historical plane are classified as TYPE 3 point. (d) A new reference frame is selected through a perspective-difference check to maintain geometric consistency. (e) The corresponding LiDAR scan is projected onto the visual image plane, and the hash buckets are constructed accordingly.

complex illumination scenarios. In contrast, feature-based methods formulate 2D–3D geometric correspondences and thus remain largely independent of photometric assumptions. The effectiveness of feature-based methods relies on the capability of front-end feature matchers, allowing the use of specialized matchers for specific conditions, such as XFeat [19] for efficient feature correspondence or Motion-Feat [20] for robust matching under motion blur. Additionally, feature-based measurements allow for explicit outlier rejection, ensuring that invalid constraints commonly found in dynamic or cluttered scenes are discarded before fusion.

Despite the advantages of the sequential feature-based LIVO pipeline, managing the visual state remains challenging due to **cross-modal geometric inconsistency**. We define cross-modal geometric inconsistency as the structural disagreement between LiDAR priors and visual observations, arising from (i) *Physical Misalignment* (e.g., inevitable calibration errors), (ii) *Complex Environment Structures*, and (iii) *Data Association Artifacts* (e.g., dynamic objects). Conventional pipelines often treat modalities as perfectly calibrated and rigidly locked, relying on strict geometric projection assumptions. Such strong assumptions risk estimation divergence in these scenarios. To address this, UAF-LIVO classifies visual features into three spatial categories based on the topological reliability of their corresponding LiDAR priors. By deriving thorough uncertainty propagation models for each category, the system enables adaptive measurement management: ill-conditioned observations are down-weighted via inflated covariance, and invalid depth associations are explicitly rejected, ensuring robustness against geometric inconsistency.

In summary, our contributions are threefold:

- We propose a feature-based sequential LIVO framework that enables robust pose estimation against cross-modal geometric inconsistency arising from both physical misalignments and data association artifacts.
- We propose a geometry-consistent visual measurement strategy that explicitly propagates uncertainty from LiDAR

to visual features, enabling adaptive down-weighting and rejection of unreliable observations.

- Experiments on diverse datasets demonstrate our system’s competitive accuracy and consistent robustness against cross-modal geometric inconsistency.

II. METHODOLOGY

The overall architecture of UAF-LIVO is illustrated in Fig. 1. In this section, we first outline the LIO subsystem which serves as the geometric prior. Subsequently, we detail our core contributions: the geometry-consistent visual measurement construction and update strategy. Then, the visual frames are managed via a reference management strategy to ensure well-conditioned optimization.

A. Notation and Preliminaries

Notation: Scalars, vectors, and matrices are denoted by italic (s), bold lower-case ($\mathbf{v}$), and bold upper-case ($\mathbf{M}$) letters, respectively. We define the LiDAR, IMU, visual, and global coordinate frames as L , I , V , and G . Left superscripts indicate the reference frame in which a quantity is expressed (e.g., ${}^G\mathbf{p}$). For brevity, omitted left superscripts default to G . Visual frames are indexed by cur, ref, and hist for the current, reference, and historical frames, respectively. Finally, $[\cdot]_{\times}$ denotes the skew-symmetric matrix operation.

Ray-Plane Intersection: The depth of a feature point is obtained by intersecting its back-projected viewing ray with a global plane defined by a normal $\mathbf{n}$ and a reference point $\mathbf{p}_0$. The intersection depth t_{hit} is computed as:

$$t_{\text{hit}}(\mathbf{n}, \mathbf{p}_0) = \frac{\mathbf{n}^T(\mathbf{p}_0 - \mathbf{p}_{\text{ref}})}{\mathbf{n}^T \mathbf{r}} \quad (1)$$

where the camera optical center $\mathbf{p}_{\text{ref}}$ and global ray direction $\mathbf{r}$ are derived from the reference pose ($\mathbf{R}_{\text{ref}}, \mathbf{t}_{\text{ref}}$) and intrinsics $\mathbf{K}$:

$$\mathbf{p}_{\text{ref}} = -\mathbf{R}_{\text{ref}}^T \mathbf{t}_{\text{ref}}, \quad \mathbf{r} = \mathbf{R}_{\text{ref}}^T \mathbf{v}, \quad \mathbf{v} \mathbf{r} = \frac{\mathbf{v} \mathbf{r}_u}{\|\mathbf{v} \mathbf{r}_u\|} \quad (2)$$

with the un-normalized ray ${}^V\mathbf{r}_u = \mathbf{K}^{-1}[u, v, 1]^\top$. Finally, the 3D point is recovered in the global frame as $\hat{\mathbf{p}}(t_{\text{hit}}) = \mathbf{p}_{\text{ref}} + t_{\text{hit}} \mathbf{r}$.

B. LIO Subsystem

The LIO subsystem serves as the geometric backbone, providing an accurate state prior for visual measurement update. The process consists of three key steps:

IMU State Propagation: Given raw angular velocity and linear acceleration, the IMU state is propagated using discrete-time zero-order-hold integration. This high-frequency propagation predicts the sensor motion, which is essential for motion-compensating LiDAR scans.

LiDAR De-skewing: Since LiDAR scans are acquired continuously, raw point clouds suffer from motion distortion. To ensure strict temporal alignment with the visual subsystem, the raw LiDAR stream is re-segmented according to the visual frame intervals. Each point is then de-skewed to the timestamp of the visual frame using IMU-based pose interpolation, following the strategy in [10], [21].

Point-to-Plane Residuals: The de-skewed point cloud is down-sampled and registered against a local map to construct point-to-plane residuals [22]. These residuals are minimized within an Iterated Error-State Kalman Filter (IESKF) to update the state vector in the IMU body frame, providing a geometrically consistent prior for the visual measurement construction.

C. Visual Measurement Construction and Update

In this part, we propose an uncertainty-aware visual measurement strategy that explicitly models noise to ensure reliable depth acquisition and detect geometric inconsistencies.

1) TYPE 1. LiDAR-Scan Local Plane Supported Features: Since LiDAR points corresponding to the visual frames only contain raw measurement uncertainty, they are first used to assign depth to the visual features. Once a frame is chosen as a reference, each LiDAR point is projected into the visual frame V :

$$\mathbf{u} = \pi(\mathbf{R}_{\text{ref}} \mathbf{p}_L + \mathbf{t}_{\text{ref}}), \quad (3)$$

where $\pi(\cdot)$ denotes the standard pinhole projection model.

To efficiently determine whether a visual feature has sufficient projected LiDAR support in its neighborhood, we construct a hash-bucket grid over the image plane. The image is uniformly divided into buckets of size $s \times s$, and each projected LiDAR point (u, v) is assigned to a bucket according to:

$$b_x = \lfloor \frac{u}{s} \rfloor, \quad b_y = \lfloor \frac{v}{s} \rfloor. \quad (4)$$

The point is then inserted into the corresponding hash table entry:

$$\mathcal{H}(b_x, b_y) \leftarrow \mathcal{H}(b_x, b_y) \cup \{(u, v)\}. \quad (5)$$

In our implementation, we set s as 8 pixels.

To ensure geometric consistency and suppress abrupt directional depth changes on the image plane (e.g., around object boundaries), we require that each of the 3×3 neighboring buckets around a query pixel contains at least one projected LiDAR point.

All LiDAR points projected into this neighborhood are collected, and their global 3D coordinates and covariances are denoted as $\{{}^G\mathbf{p}_L^j, {}^G\boldsymbol{\Sigma}_L^j\}_{j=1}^N$. A local plane is then fitted by computing the centroid and covariance:

$$\bar{\mathbf{p}} = \frac{1}{N} \sum_{j=1}^N {}^G\mathbf{p}_L^j, \quad \mathbf{C} = \frac{1}{N} \sum_{j=1}^N ({}^G\mathbf{p}_L^j - \bar{\mathbf{p}})({}^G\mathbf{p}_L^j - \bar{\mathbf{p}})^\top. \quad (6)$$

The plane normal $\mathbf{n}$ is determined as the eigenvector associated with the smallest eigenvalue of $\mathbf{C}$. A neighborhood is regarded as *locally smooth* if:

$$r_j = |\mathbf{n}^\top ({}^G\mathbf{p}_L^j - \bar{\mathbf{p}})| < \tau_{\text{plane}}, \quad \forall j. \quad (7)$$

In our implementation, τ_{plane} is set to 0.05.

For a smooth-neighborhood feature, the depth is obtained by intersecting its back-projected ray with the fitted plane using the *ray-plane intersection formula* $t_{\text{hit}}(\mathbf{n}, \mathbf{p}_0)$ and corresponding 3D point is recovered as $\hat{\mathbf{p}}(t_{\text{hit}})$.

Since the points used here lie on the same fitted plane, the estimated point $\hat{\mathbf{p}}(t_{\text{hit}})$ can be approximated as a linear combination of the neighboring LiDAR points. Therefore, the uncertainty is propagated as:

$$\boldsymbol{\Sigma}_{\text{geo}} = \sum_{j=1}^N w_j^2 {}^G\boldsymbol{\Sigma}_L^j, \quad (8)$$

where the weights w_j are obtained by solving the overdetermined system:

$$\mathbf{w} = \arg \min_{\mathbf{w}} \left\| {}^G\hat{\mathbf{p}} - \sum_{j=1}^N w_j {}^G\mathbf{p}_L^j \right\|^2, \quad (9)$$

in a least-squares sense, and $\mathbf{w} = [w_1, w_2, \dots, w_N]^\top$ denotes the coefficient of linear combination.

2) TYPE 2. Voxel-Map Supported Features: For features lacking local support due to the sparsity of LiDAR, we utilize a planar parameterized voxel map [23] with accumulated time geometric uncertainty to further construct visual constraints.

We back-project the feature ray and employ the 3D DDA algorithm [24] to efficiently traverse the map. Upon intersecting the nearest valid planar voxel $(\mathbf{n}, \mathbf{p}_0)$, the depth $t_{\text{hit}}(\mathbf{n}, \mathbf{p}_0)$ is computed, yielding the global point $\hat{\mathbf{p}}(t_{\text{hit}})$.

The uncertainty of the intersection point arises from four sources: (i) camera rotation, (ii) camera translation, (iii) plane normal, and (iv) a point on the plane, all of which follow Gaussian distributions:

$$\begin{aligned} \mathbf{R}_{\text{ref}} &\sim \mathcal{N}(\hat{\mathbf{R}}_{\text{ref}}, \boldsymbol{\Sigma}_{\mathbf{R}}), & \mathbf{t}_{\text{ref}} &\sim \mathcal{N}(\hat{\mathbf{t}}_{\text{ref}}, \boldsymbol{\Sigma}_{\mathbf{t}}), \\ \mathbf{n} &\sim \mathcal{N}(\hat{\mathbf{n}}, \boldsymbol{\Sigma}_{\mathbf{n}}), & \mathbf{p}_0 &\sim \mathcal{N}(\hat{\mathbf{p}}_0, \boldsymbol{\Sigma}_{\mathbf{p}_0}). \end{aligned}$$

where $\hat{\cdot}$ denotes the estimated state values, and $\boldsymbol{\Sigma}_{\cdot}$ are the corresponding covariance matrices.

The uncertainty of t_{hit} can be calculated by first-order uncertainty propagation:

$$\begin{aligned} \sigma_t^2 &= \mathbf{J}_{\mathbf{p}_0} \boldsymbol{\Sigma}_{\mathbf{p}_0} \mathbf{J}_{\mathbf{p}_0}^\top + \mathbf{J}_{\mathbf{n}} \boldsymbol{\Sigma}_{\mathbf{n}} \mathbf{J}_{\mathbf{n}}^\top + \mathbf{J}_{\mathbf{t}_{\text{ref}}} \boldsymbol{\Sigma}_{\mathbf{t}} \mathbf{J}_{\mathbf{t}_{\text{ref}}}^\top \\ &\quad + \mathbf{J}_{\mathbf{R}_{\text{ref}}} \boldsymbol{\Sigma}_{\mathbf{R}} \mathbf{J}_{\mathbf{R}_{\text{ref}}}^\top, \end{aligned} \quad (10)$$

where the Jacobian of t_{hit} with respect to $\mathbf{p}_0$ can be calculated with:

$$\mathbf{J}_{\mathbf{p}_0} = \frac{\partial t_{\text{hit}}}{\partial \mathbf{p}_0} = \frac{\hat{\mathbf{n}}^\top}{\hat{\mathbf{n}}^\top \mathbf{r}}, \quad (11)$$

and Jacobian of t_{hit} with respect to $\mathbf{n}$ can be calculated with:

$$\mathbf{J}_{\mathbf{n}} = \frac{\partial t_{\text{hit}}}{\partial \mathbf{n}} = \frac{(\hat{\mathbf{p}}_0 - \hat{\mathbf{p}}_{\text{ref}})^\top}{\hat{\mathbf{n}}^\top \mathbf{r}} - \frac{(\hat{\mathbf{n}}^\top (\hat{\mathbf{p}}_0 - \hat{\mathbf{p}}_{\text{ref}})) \mathbf{r}^\top}{(\hat{\mathbf{n}}^\top \mathbf{r})^2}. \quad (12)$$

where $\hat{\mathbf{p}}_{\text{ref}} = -\hat{\mathbf{R}}_{\text{ref}}^\top \hat{\mathbf{t}}_{\text{ref}}$.

For the motion part, the Jacobian of t_{hit} with respect to the state translation $\mathbf{t}_{\text{ref}}$ is:

$$\mathbf{J}_{\mathbf{t}_{\text{ref}}} = \frac{\partial t_{\text{hit}}}{\partial \mathbf{t}_{\text{ref}}} = \frac{(\hat{\mathbf{R}}_{\text{ref}} \hat{\mathbf{n}})^\top}{\hat{\mathbf{n}}^\top \mathbf{r}}, \quad (13)$$

and the Jacobian of t_{hit} with respect to $\mathbf{R}_{\text{ref}}$ is:

$$\mathbf{J}_{\mathbf{R}_{\text{ref}}} = \frac{(\hat{\mathbf{p}}_{\text{ref}} \times \hat{\mathbf{n}})^\top}{\hat{\mathbf{n}}^\top \mathbf{r}} + \frac{\hat{\mathbf{n}}^\top (\hat{\mathbf{p}}_0 - \hat{\mathbf{p}}_{\text{ref}})}{(\hat{\mathbf{n}}^\top \mathbf{r})^2} (\mathbf{r} \times \hat{\mathbf{n}})^\top. \quad (14)$$

Finally, the uncertainty of the TYPE 2 3D point $\mathbf{p}$ is obtained by:

$$\Sigma_{\text{geo}} = \sigma_t^2 \mathbf{r} \mathbf{r}^\top. \quad (15)$$

3) TYPE 3: Bearing-Only Features: For feature points where the ray-plane intersection is unavailable (e.g., LiDAR-unobserved regions), we fall back to a bearing-only constraint [25] to constrain rotation and prevent the estimator from becoming ill-conditioned due to sparse depth associations.

We model the pixel-plane noise as:

$$\Sigma_{\text{pix}} = \sigma_{\text{pix}}^2 \mathbf{I}_2, \quad \sigma_{\text{pix}} = \frac{k}{\text{score}_i}, \quad (16)$$

where score_i denotes the matching confidence from the feature matcher, $\mathbf{I}_2$ is the 2×2 identity matrix, and we set $k = 3$ pixels. The Jacobian from pixel coordinates to the un-normalized camera ray is:

$$\mathbf{J}_{\mathbf{v}_{\mathbf{r}_u, \mathbf{u}}} = \begin{bmatrix} 1/f_x & 0 \\ 0 & 1/f_y \\ 0 & 0 \end{bmatrix}. \quad (17)$$

Following normalization and transformation into the global frame, the ray-direction uncertainty is computed by propagating Σ_{pix} through the mapping:

$$\Sigma_{\text{geo}} = \mathbf{J}_{\mathbf{r}, \mathbf{u}} \Sigma_{\text{pix}} \mathbf{J}_{\mathbf{r}, \mathbf{u}}^\top + \mathbf{J}_{\mathbf{r}, \mathbf{R}_{\text{ref}}} \Sigma_{\mathbf{R}} \mathbf{J}_{\mathbf{r}, \mathbf{R}_{\text{ref}}}^\top, \quad (18)$$

where

$$\mathbf{J}_{\mathbf{r}, \mathbf{u}} = \mathbf{R}_{\text{ref}}^\top \|\mathbf{v}_{\mathbf{r}_u}\|^{-1} (\mathbf{I}_3 - \mathbf{v}_{\mathbf{r}_u} \mathbf{v}_{\mathbf{r}_u}^\top) \mathbf{J}_{\mathbf{v}_{\mathbf{r}_u, \mathbf{u}}}, \quad \mathbf{J}_{\mathbf{r}, \mathbf{R}_{\text{ref}}} = [\mathbf{r}]_{\times}. \quad (19)$$

4) Visual Measurement Update: We employ an IESKF to estimate the error state $^I \delta \mathbf{x}$. Crucially, the propagated geometric uncertainty Σ_{geo} is explicitly incorporated into the measurement covariance $\mathbf{R}_k$:

$$\mathbf{R}_k = \mathbf{J}_{\text{geo}} \Sigma_{\text{geo}} \mathbf{J}_{\text{geo}}^\top + \sigma_{\text{pix}}^2 \mathbf{I}, \quad (20)$$

where $\sigma_{\text{pix}}^2 \mathbf{I}$ is the pixel measurement noise on the current frame.

For TYPE 1/2 features, the geometric primitive is the global 3D point $\mathbf{p}$. The Jacobian $\mathbf{J}_{\text{geo}}$ is derived via the chain rule w.r.t. the visual-frame coordinates $^V \mathbf{p} = [X, Y, Z]^\top$:

$$\mathbf{J}_{\text{geo}}^{\mathbf{p}} = \frac{\partial \mathbf{u}}{\partial ^V \mathbf{p}} \frac{\partial ^V \mathbf{p}}{\partial \mathbf{p}} = \begin{bmatrix} \frac{f_x}{Z} & 0 & -\frac{f_x X}{Z^2} \\ 0 & \frac{f_y}{Z} & -\frac{f_y Y}{Z^2} \end{bmatrix} \mathbf{R}_{\text{cur}}. \quad (21)$$

For TYPE 3 features, the geometric primitive is the global bearing vector $\mathbf{r}$. The Jacobian is derived w.r.t. the visual-frame ray $^V \mathbf{r} = [x, y, z]^\top$:

$$\mathbf{J}_{\text{geo}}^{\mathbf{r}} = \frac{\partial \mathbf{u}}{\partial ^V \mathbf{r}} \frac{\partial ^V \mathbf{r}}{\partial \mathbf{r}} = \begin{bmatrix} \frac{f_x}{z} & 0 & -\frac{f_x x}{z^2} \\ 0 & \frac{f_y}{z} & -\frac{f_y y}{z^2} \end{bmatrix} \mathbf{R}_{\text{cur}}. \quad (22)$$

Uncertain constraints thus yield a large $\mathbf{R}_k$, automatically down-weighting their influence in the IESKF update:

$$^I \delta \mathbf{x} = (\mathbf{H}^\top \mathbf{R}_k^{-1} \mathbf{H} + \Sigma_{\text{prior}}^{-1})^{-1} \mathbf{H}^\top \mathbf{R}_k^{-1} \delta \mathbf{z}, \quad (23)$$

where Σ_{prior} is the prior state covariance, and $\delta \mathbf{z}$ is the corresponding visual residual. The measurement Jacobian $\mathbf{H} = \frac{\partial \mathbf{u}}{\partial ^V \mathbf{p}} \frac{\partial ^V \mathbf{p}}{\partial ^I \delta \mathbf{x}}$ explicitly chains the camera projection model with the IMU error state via the camera-IMU extrinsics.

5) Unreliable Visual Measurement Rejection: Visual measurements of all types are rejected using the Chi-squared test based on the Mahalanobis distance:

$$r_M^2 = \delta \mathbf{z}^\top \mathbf{R}_k^{-1} \delta \mathbf{z} > \chi_{\text{th}}^2, \quad (24)$$

where $\mathbf{R}_k$ is the uncertainty-aware covariance derived in (20). With (24), visual measurements are judged relative to their geometric confidence. In our implementation, the threshold is set to 9.

Remark (Physical Insights of Decoupled Covariance): Since the projection Jacobian $\frac{\partial \mathbf{u}}{\partial ^V \mathbf{p}}$ scales inversely with depth Z , a fixed-covariance strategy may make near-field observations disproportionately influential in $\Lambda = \mathbf{H}^\top \mathbf{R}_k^{-1} \mathbf{H}$, especially under cross-modal misalignment or unreliable depth association. UAF-LIVO decouples 2D tracking noise from 3D structural uncertainty Σ_{geo} , where TYPE 1 features inherit bounded local LiDAR uncertainty and TYPE 2 features carry depth-, pose-, and plane-dependent ray-plane intersection uncertainty. By projecting Σ_{geo} onto the image plane, $\mathbf{J}_{\text{geo}} \Sigma_{\text{geo}} \mathbf{J}_{\text{geo}}^\top$ increases the covariance of geometrically uncertain or projection-sensitive constraints. This mitigates depth-induced imbalance in the visual information matrix Λ and yields a more balanced visual information contribution.

D. Visual Keyframe Management

To ensure well-conditioned optimization, we implement a frame management strategy consisting of historical maintenance and optimal reference selection.

1) *Historical Frame Maintenance*: We maintain a fixed-length sliding window of historical keyframes to support visual tracking. To ensure geometric diversity and avoid redundancy from hovering motion, a new frame is inserted into the queue only if it exhibits sufficient motion relative to the latest keyframe.

The relative rotation $\mathbf{R}_{\text{rel}}$ and global translation baseline $\mathbf{t}_{\text{rel}}$ between historical frames are computed as:

$$\mathbf{R}_{\text{rel}} = \mathbf{R}_{\text{cur}} \mathbf{R}_{\text{hist}}^T, \quad \mathbf{t}_{\text{rel}} = \mathbf{t}_{\text{cur}} - \mathbf{t}_{\text{hist}}. \quad (25)$$

The frame is accepted as a new keyframe if **any** relative motion exceeds preset thresholds:

$$\angle(\mathbf{R}_{\text{rel}}) > \theta_{\text{th}} \quad \text{or} \quad \|\mathbf{t}_{\text{rel}}\| > t_{\text{th}}. \quad (26)$$

In practice, we set $\theta_{\text{th}} = 10^\circ$ and $t_{\text{th}} = 1$ m. If the queue reaches its maximum capacity N_{max} , the oldest frame is discarded. Upon acceptance, the new keyframe immediately undergoes the LiDAR projection process to establish local geometric support for future tracking. In the implementation, we set $N_{\text{max}} = 3$.

2) *Reference Frame Selection*: At each measurement update step, we select the optimal reference frame from the history queue to form constraints with the current frame. The selection maximizes a perspective-utility score S that balances parallax, overlap, and pose uncertainty.

Given the current optical center and a candidate historical optical center, we define the baseline vector $\mathbf{b} = \mathbf{t}_{\text{cur}} - \mathbf{t}_{\text{ref}}$. The score is formulated as:

$$S = \frac{\|\mathbf{b}_\perp\| \exp\left(-\frac{\theta^2}{2\sigma^2}\right)}{\text{tr}(\Sigma_{\mathbf{R}}) + \text{tr}(\Sigma_{\mathbf{t}})}, \quad (27)$$

where $\mathbf{b}_\perp = \mathbf{b} - (\mathbf{v}_{\text{ref}}^T \mathbf{b}) \mathbf{v}_{\text{ref}}$ represents the baseline component orthogonal to the reference viewing direction $\mathbf{v}_{\text{ref}}$, and θ is the relative rotation angle. This scoring function explicitly favors frames with large effective parallax for well-conditioned optimization, while penalizing excessive viewpoint divergence via the exponential term and high pose uncertainty via the denominator.

III. EXPERIMENTS

This section presents a comprehensive evaluation of the proposed UAF-LIVO framework. We first validate the fidelity of our uncertainty-aware visual construction and explicitly demonstrate its resilience against cross-modal geometric inconsistency through controlled stress tests. Subsequently, the overall system is compared against state-of-the-art frameworks across diverse and challenging environments, followed by a detailed real-time efficiency analysis.

A. Implementation Details

We evaluate our method on M2DGR [26], M3DGR [27], and GEODE [28]. M2DGR provides RGB and thermal data with a Velodyne spinning LiDAR, whereas M3DGR provides RGB data with LiDAR measurements captured by Livox Avia. GEODE metro-tunnel gamma sequences are used to evaluate robustness against cross-modal geometric inconsistency.

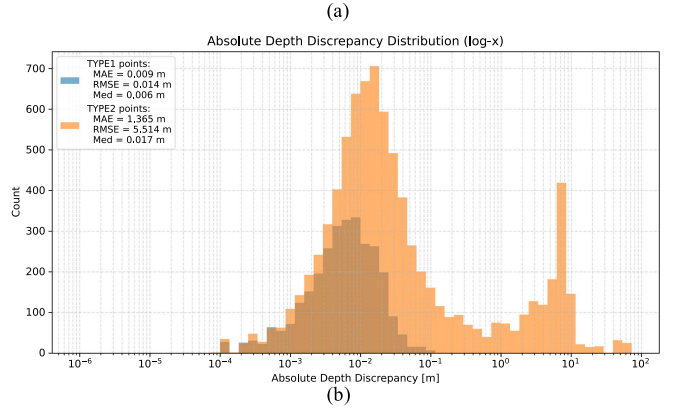
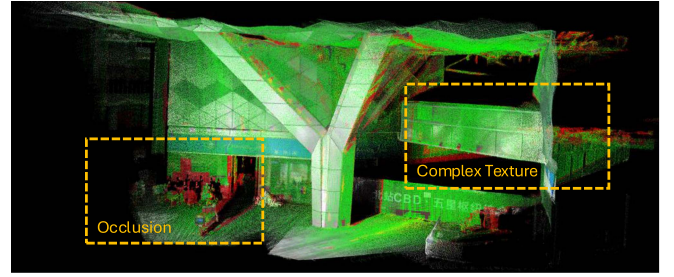


Fig. 2. Validation of visual measurement construction. (a) Qualitative global spatial distribution of depth discrepancy. Green: depth discrepancy less than 1%, Yellow: depth discrepancy less than 3% but over 1%, Red: depth discrepancy over 3%. (b) Quantitative depth discrepancy distribution in the occlusion case.

Experiments were conducted on a PC with RTX 5070 Ti GPU, i7-13700KF CPU, and 32 GB RAM. Our implementation builds on the LIO subsystem of Fast-LIVO2 [10], using a 0.5 m voxel map, 10 map-update iterations per frame, and a 100 m DDA ray-casting range. LightGlue is accelerated with TensorRT FP32 using at most 128 keypoints. For **Dark01**, whose trajectory forms a nearly rectangular closed loop, unconstrained full-trajectory SE(3) alignment occasionally produced inconsistent registrations and anomalously large ATE values. We therefore adopt a constrained deterministic alignment protocol consisting of initial-frame anchoring and scale-free rotation-only alignment. The same protocol is applied to all compared methods.

B. Validation of Visual Measurement Construction

To validate the fidelity of our visual measurement construction, we evaluate cross-modal depth discrepancies using a structural building sequence from the Fast-LIVO2 dataset [10], which features prominent planar structures coupled with inherent geometric inconsistencies. As illustrated in Fig. 2(a), the spatial distribution of these discrepancies is consistent with our underlying rationale: while planar regions on frontal walls maintain high depth fidelity, complex textures and occlusion boundaries suffer from severe structural misalignment.

To quantify this misalignment, we analyze the error distribution specifically within the occlusion region (Fig. 2(b)). Benefiting from thorough planar validity checks, TYPE 1 features achieve sub-centimeter accuracy with an MAE of 0.009 m and an RMSE of 0.014 m. In contrast, TYPE 2 features exhibit a pronounced long-tailed error distribution. Although most raw

TABLE I
ATE RMSE (M) UNDER EXTRINSIC PERTURBATION. SIGNIFICANT ATE INCREASES ARE MARKED WITH AN ASTERISK (*).

Method	σ_{pix} (px)	Dynamic03			Dynamic04		
		0°	1°	3°	0°	1°	3°
FixCov	5	0.22	0.23	0.22	0.38	0.40	0.35
	10	0.23	0.23	0.51*	0.38	0.40	0.38
	20	0.23	0.23	0.23	0.39	0.75*	0.53*
Ours	-	0.22	0.22	0.23	0.39	0.40	0.39

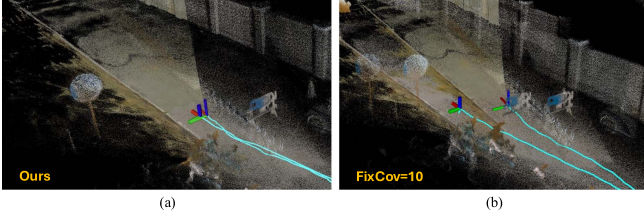


Fig. 3. Qualitative comparison of map consistency of **Dynamic03** sequence under 3° perturbation. (a) Map of our uncertainty-aware mechanism. (b) Map of fixed covariance ($\sigma_{\text{pix}} = 10$ px) strategy.

errors remain bounded, a critical subset of outliers severely violates the LiDAR prior, increasing the overall TYPE 2 errors to an MAE of 1.365 m and an RMSE of 5.514 m.

This performance gap highlights the importance of our uncertainty-aware formulation to prevent such geometric outliers from corrupting the estimator. Accordingly, our method adaptively down-weights or rejects ill-conditioned TYPE 2 constraints based on the estimated uncertainty.

C. Robustness Analysis to Geometric Inconsistency

To explicitly validate robustness against cross-modal geometric inconsistency, we conduct a stress test on the **Dynamic03** and **Dynamic04** sequences by injecting systematic extrinsic perturbations. We define two discrete perturbation levels: a 1° rotational error coupled with a 0.05 m per-axis translational offset, and a 3° error coupled with a 0.10 m per-axis offset. For brevity, these coupled perturbations are denoted solely by their rotational magnitudes in Table I. To establish a conventional fixed-covariance baseline (denoted as FixCov), we fix the visual measurement covariance as $\mathbf{R}_k = \sigma_{\text{pix}}^2 \mathbf{I}$, where σ_{pix} represents a uniform pixel noise tolerance. We evaluate this baseline under three distinct tolerance levels: $\sigma_{\text{pix}} \in \{5, 10, 20\}$ pixels.

As shown in Table I, utilizing a loose pixel noise tolerance ($\sigma_{\text{pix}} \geq 10$ px) increases the global ATE across different scenarios and perturbation levels. Specifically, in the **Dynamic03** sequence under a 3° perturbation, the fixed tolerance of $\sigma_{\text{pix}} = 10$ px further exacerbates local trajectory jitter, leading to observable map distortion, as visualized in Fig. 3. A tighter tolerance ($\sigma_{\text{pix}} = 5$ px) achieves comparable ATE by enforcing globally strict rejection, yet introduces local jitter due to disproportionately high visual weights, as highlighted in Fig. 4. By deriving visual uncertainty directly from the LIO structural covariance, our formulation alleviates this sensitivity. Although system extrinsic perturbations inevitably distort the measurement Jacobian and skew the resulting Hessian in (23),

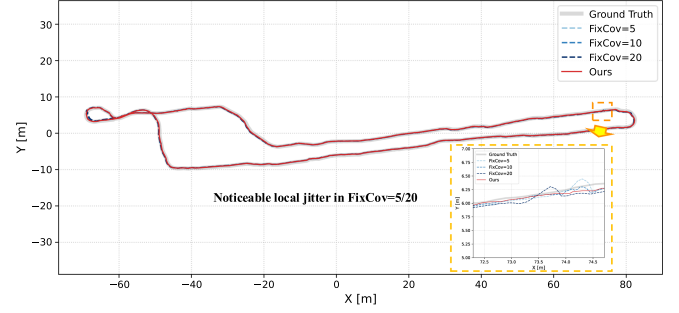


Fig. 4. Visualization of estimated trajectory of **Dynamic04** sequence under 3° perturbation.

our method effectively suppresses these structural deviations via uncertainty-aware rejection and adaptive near-field down-weighting, thereby mitigating local trajectory jitter.

D. M2DGR & M3DGR Datasets

In this subsection, we quantitatively compare our method against four representative frameworks: Fast-LIO2 [5], SR-LIVO [9], Fast-LIVO2 [10], and LVI-SAM [16]. Fast-LIO2 serves as a pure LIO baseline to compare the impact of vision on system accuracy. LVI-SAM is a feature-based parallel LIV-SLAM framework tightly coupled through factor graph optimization. For evaluations on the M3DGR dataset, we utilize the modified LVI-SAM implementation provided by the M3DGR benchmark. SR-LIVO relies on LIO for pose estimation and its visual component for online calibration. Fast-LIVO2 utilizes sequential fusion strategy with direct photometric visual alignment.

Table II reports the overall performance across various scenarios. For the outdoor **gate_01** sequence, LVI-SAM exhibits a noticeably higher ATE. This primarily stems from the imbalance between LiDAR and visual modalities: the VIO subsystem exhibits sensitivity to short baseline initialization, which negatively impacts the joint factor graph optimization. Fast-LIVO2's dense photometric alignment exacerbates geometric inconsistencies in outdoor unstructured scenarios, resulting in inferior performance compared with the thermal modality. In contrast, our uncertainty-aware mechanism naturally down-weights and rejects these ill-conditioned structural mismatches, ensuring stability across both RGB and thermal modalities. For the indoor **hall** sequences featuring structured corridors, our method achieves competitive accuracy, securing the lowest ATE of 1.16 m in **hall_05**. Regarding the small-scale **room** sequences, LVI-SAM yields the best absolute accuracy. This is theoretically expected, as LVI-SAM's explicit edge-feature extraction is inherently advantageous in highly cluttered environments where continuous planar structures are severely fragmented. However, among all LiDAR feature-free scan-to-map frameworks, our method consistently matches or outperforms Fast-LIVO2 and SR-LIVO. Furthermore, in the textureless **room_dk_01** sequence, the competing LIVO frameworks are compromised by the adverse optimization effects of degraded visual tracking, whereas our system maintains robust performance.

TABLE II

QUANTITATIVE COMPARISON AGAINST LIO AND LIVO FRAMEWORKS. ATE: ABSOLUTE TRAJECTORY ERROR [M]; RTE: RELATIVE TRANSLATION ERROR [M]. BEST RESULTS ARE **BOLDED**, SECOND BEST ARE UNDERLINED.

Sequence	Fast-LIO2		LVI-SAM				Fast-LIVO2				SR-LIVO				Ours			
			RGB		Ther.		RGB		Ther.		RGB		Ther.		RGB		Ther.	
	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE	ATE	RTE
gate_01	1.53	0.005	2.20	0.009	1.59	<u>0.008</u>	1.90	0.010	1.56	<u>0.008</u>	1.68	0.008	1.97	<u>0.008</u>	1.63	0.009	1.57	<u>0.008</u>
hall_01	2.22	<u>0.101</u>	<u>2.18</u>	0.100	2.20	0.100	2.16	0.104	<u>2.24</u>	<u>0.102</u>	2.38	<u>0.102</u>	2.41	<u>0.102</u>	2.30	0.102	2.23	<u>0.105</u>
hall_04	1.69	0.086	1.74	0.086	1.73	0.086	<u>1.62</u>	0.084	1.73	0.088	1.57	<u>0.085</u>	1.68	0.087	<u>1.62</u>	0.084	1.73	0.089
hall_05	1.28	0.066	1.30	<u>0.067</u>	1.42	0.073	<u>1.22</u>	0.068	1.30	0.068	1.61	<u>0.072</u>	1.58	0.070	1.16	<u>0.067</u>	1.30	<u>0.067</u>
room_01	0.44	0.007	0.27	0.007	<u>0.28</u>	0.007	0.46	<u>0.008</u>	0.44	0.007	0.64	0.010	0.66	0.010	0.46	0.007	0.43	0.007
room_dk_01	0.59	0.008	0.44	0.007	0.44	0.007	0.69	0.009	0.59	<u>0.008</u>	0.87	0.009	0.53	0.009	0.58	0.009	<u>0.50</u>	0.009
Dark01	2.46	0.913	6.55	2.005	-	-	2.43	0.923	-	-	2.56	1.052	-	-	2.42	0.963	-	-
Dynamic03	0.30	0.080	3.05	0.813	-	-	<u>0.25</u>	<u>0.081</u>	-	-	1.28	<u>0.081</u>	-	-	0.22	0.080	-	-
Occlusion03	0.24	0.082	8.86	2.866	-	-	<u>0.29</u>	<u>0.083</u>	-	-	×	×	-	-	0.33	<u>0.083</u>	-	-

TABLE III

ABSOLUTE TRAJECTORY ERROR (M) ON GEODE METRO TUNNEL GAMMA SEQUENCES. **DRIFT** MEANS ATE > 100 M.

Method	Tunnel 1	Tunnel 2	Tunnel 3	Tunnel 4	Tunnel 5
LVI-SAM [16]	35.4	<u>2.10</u>	<u>0.24</u>	<u>0.35</u>	0.17
RELEAD [29]	0.37	2.32	0.22	0.15	<u>0.35</u>
SR-LIVO [9]	×	×	×	×	×
Fast-LIVO2 [10]	1.51	drift	6.69	3.75	21.3
Ours (w/o Rej.)	2.40	drift	drift	drift	drift
Ours (Full)	<u>0.61</u>	1.95	1.71	0.85	0.52

On the M3DGR dataset, Fast-LIVO2 and our method achieve comparable sub-meter accuracy across all sequences. In contrast, other baselines encounter specific challenges. Notably, LVI-SAM is primarily designed for spinning LiDARs, resulting in insufficient edge-feature detection under non-repetitive scanning. This geometric sparsity, compounded by unreliable visual constraints in the textureless **Dark01** sequence, makes it susceptible to scale drift. Meanwhile, in **Occlusion03**, SR-LIVO fails to complete the trajectory due to premature visual termination under extended occlusion. Beyond these cases, LVI-SAM and SR-LIVO also experience trajectory drift and degraded accuracy in dynamic or occluded scenarios, demonstrating the robustness advantages of the sequential fusion strategy.

E. GEODE Tunnel Scenarios

We evaluate our method against state-of-the-art methods across five metro tunnel sequences captured by Livox Avia LiDAR in the GEODE dataset [28]. The overall quantitative results are shown in Table III. As the official versions of LVI-SAM and RELEAD supporting the Avia LiDAR are not yet open-source, we adopt the baseline results reported in the GEODE benchmark directly.

In tunnel scenarios, SR-LIVO does not produce complete trajectories on these sequences due to RANSAC PnP failures in its visual component. Notably, RELEAD achieves the best overall performance across the five sequences, benefiting from its LiDAR degradation detection via Hessian ill-conditioning analysis and graduated non-convexity optimization. LVI-SAM also achieves competitive performance through its ability to identify structural or textureless degeneracy; it selectively suppresses

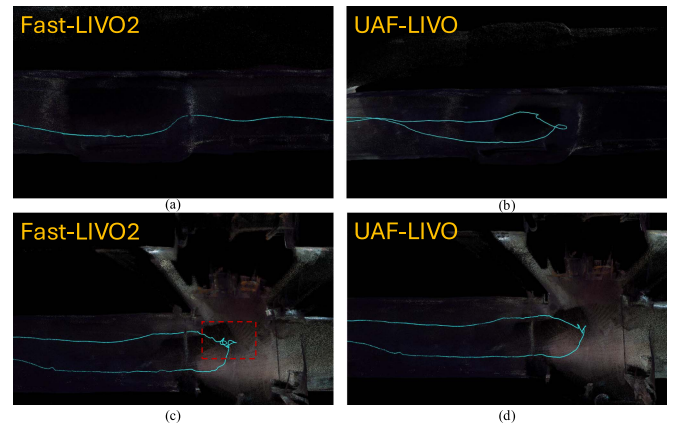


Fig. 5. Visualization of estimated trajectory in GEODE Metro Tunneling Sequence. First Row: GEODE Metro Tunnel 2 sequence. Second Row: GEODE Metro Tunnel 4 sequence.

unreliable updates in the optimization window by monitoring Hessian eigenvalues and tracking quality, thereby maintaining well-conditioned state estimation. Since our LIO subsystem inherently lacks explicit LiDAR degradation detection and suppression mechanisms, its overall performance in these LiDAR-degraded tunnels is slightly inferior to LVI-SAM and RELEAD. However, by strictly discarding unreliable visual constraints, our method still achieves competitive accuracy on Tunnels 1 and 2.

Although sharing a similar sequential update architecture, Fast-LIVO2 exhibits higher tracking errors across the five evaluated sequences, primarily due to its sensitivity to cross-modal inconsistency. In the GEODE metro tunnel scenarios, the LiDAR modality often encounters degeneracy during rotations within narrow passages, which is further compounded by extremely low-light conditions that compromise visual reliability. Fast-LIVO2 tends to integrate these degraded visual measurements into the state estimation without an explicit rejection scheme. Specifically, in the Tunnel 2 sequence, the low rotation velocity prolongs LiDAR degeneracy, allowing unreliable visual constraints to corrupt the optimization and trigger global drift, as illustrated in Fig. 5(a). A similar failure is observed in our ablation model without rejection (w/o Rej.), which drifts in more sequences because erroneous feature-based 2D–3D constraints

TABLE IV
RUNTIME COMPARISON (MS): LIDAR AND VISUAL BRANCHES PERFORMANCE.
VLP-32: MECHANICAL; AVIA: NON-REPETITIVE. BEST: **BOLDED**.

Method	VLP-32		Avia	
	LIO	VO	LIO	VO
LVI-SAM	82.72	11.86	21.86	12.57
Fast-LIVO2	47.77	7.90	7.77	2.92
SR-LIVO	14.78	11.43	4.74	6.64
Ours	53.96	12.10	7.90	8.40

impose more direct geometric restrictions than photometric residuals. This confirms that LiDAR degeneracy compromises baseline accuracy, while unhandled cross-modal inconsistency further exacerbates degradation and may cause divergence. In contrast, our framework explicitly discards ill-conditioned measurements to maintain estimation consistency. Furthermore, Tunnel 4 shows that unhandled spatial inconsistency between LiDAR and visual modalities can induce observable trajectory offsets (Fig. 5(c)), which our approach avoids through consistency validation.

F. Time Efficiency Analysis

Table IV details the runtime performance on the **gate_01** and **Dynamic03** sequences. Although our uncertainty-aware fusion introduces marginal overhead compared to SR-LIVO or Fast-LIVO2, it is significantly faster than LVI-SAM's factor graph optimization. Our system maintains high real-time efficiency, requiring only 53.96/7.90 ms for LIO subsystem and 12.10/8.40 ms for visual component. Notably, the framework operates predominantly on the CPU, utilizing GPU acceleration exclusively for visual feature matching.

IV. CONCLUSION

This letter presents UAF-LIVO, a sequential framework addressing cross-modal geometric inconsistency via thorough uncertainty propagation and explicit outlier rejection. By classifying visual features into three geometry-aware types based on LiDAR spatial relationships, we derive measurement covariances to adaptively down-weight ill-conditioned observations. Experiments demonstrate its competitive accuracy and consistent robustness in severely degraded environments. While our system exhibits significant resilience against cross-modal mismatches, it may still experience trajectory drift under severe LiDAR modality degeneracy. Future work will focus on mitigating this reliance on the LiDAR geometric prior to sustain robust estimation when LiDAR measurements are unreliable.

REFERENCES

- [1] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.
- [2] L. V. Stumberg and D. Cremers, "DM-VIO: Delayed marginalization visual-inertial odometry," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 1408–1415, Apr. 2022.
- [3] J. Kaiser, A. Martinelli, F. Fontana, and D. Scaramuzza, "Simultaneous state initialization and gyroscope bias calibration in visual inertial aided navigation," *IEEE Robot. Automat. Lett.*, vol. 2, no. 1, pp. 18–25, Jan. 2017.
- [4] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled LiDAR inertial odometry via smoothing and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5135–5142.
- [5] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, 2022.
- [6] D. He et al., "Point-LIO: Robust high-bandwidth light detection and ranging inertial odometry," *Adv. Intell. Syst.*, vol. 5, no. 7, 2023, Art. no. 2200459.
- [7] C. Zhang, J. Zhang, Q. Liu, Y. Liu, and J. Qin, "SI-LIO: High-precision tightly-coupled LiDAR-inertial odometry via single-iteration invariant extended Kalman filter," *IEEE Robot. Automat. Lett.*, vol. 10, no. 1, pp. 564–571, Jan. 2025.
- [8] L. He, B. Li, and G. Chen, "Place-LIO: Plane-centric LiDAR-inertial odometry," *IEEE Robot. Automat. Lett.*, vol. 10, no. 6, pp. 6231–6238, Jun. 2025.
- [9] Z. Yuan, J. Deng, R. Ming, F. Lang, and X. Yang, "SR-LIVO: LiDAR-inertial-visual odometry and mapping with sweep reconstruction," *IEEE Robot. Automat. Lett.*, vol. 9, no. 6, pp. 5110–5117, Jun. 2024.
- [10] C. Zheng et al., "Fast-LIVO2: Fast, direct LiDAR-inertial-visual odometry," *IEEE Trans. Robot.*, vol. 41, pp. 326–346, 2024.
- [11] T. Shi, K. Qian, Y. Fang, Y. Zhang, and H. Yu, "Point-line LIVO using patch-based gradient optimization for degenerate scenes," *IEEE Robot. Automat. Lett.*, vol. 9, no. 11, pp. 9717–9724, Nov. 2024.
- [12] J. Lin, C. Zheng, W. Xu, and F. Zhang, "R² live: A robust, real-time, lidar-inertial-visual tightly-coupled state estimator and mapping," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 7469–7476, Oct. 2021.
- [13] H. Liu et al., "A real-time LiDAR-visual-inertial object level semantic slam for forest environments," *ISPRS J. Photogrammetry Remote Sens.*, vol. 219, pp. 71–90, 2025.
- [14] S. Hong et al., "GS-LIVO: Real-time LiDAR, inertial, and visual multi-sensor fused odometry with Gaussian mapping," *IEEE Trans. Robot.*, vol. 41, pp. 4253–4268, 2025.
- [15] T. Wen, Y. Fang, B. Lu, X. Zhang, and C. Tang, "LIVER: A tightly coupled LiDAR-inertial-visual state estimator with high robustness for underground environments," *IEEE Robot. Automat. Lett.*, vol. 9, no. 3, pp. 2399–2406, Mar. 2024.
- [16] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LVI-SAM: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 5692–5698.
- [17] S. Zhou et al., "LIR-LIVO: A lightweight, robust LiDAR/vision/inertial odometry with illumination-resilient deep features," 2025, *arXiv:2502.08676*.
- [18] J. Lin and F. Zhang, "R3LIVE: A robust, real-time, RGB-colored, LiDAR-inertial-visual tightly-coupled state estimation and mapping package," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 10672–10678.
- [19] G. Potje, F. Cadar, A. Araujo, R. Martins, and E. R. Nascimento, "XFeat: Accelerated features for lightweight image matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 1–10.
- [20] Y. Gao, D. Zhang, Y. Yu, J. Zhang, H. Li, and J. Chen, "Motion-feat: Motion blur-aware local feature description for image matching," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2025, pp. 2448–2454.
- [21] Z. Yuan, Q. Wang, K. Cheng, T. Hao, and X. Yang, "SDV-LOAM: Semi-direct visual-LiDAR odometry and mapping," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 11203–11220, Sep. 2023.
- [22] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Conf. Robot. Sci. Syst.*, Berkeley, CA, USA, 2014, pp. 1–9.
- [23] C. Yuan, W. Xu, X. Liu, L. Chen, and F. Zhang, "Efficient and probabilistic adaptive voxel mapping for accurate online LiDAR odometry," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 8518–8525, Jul. 2022.
- [24] J. Amanatides and A. Woo, "A fast voxel traversal algorithm for ray tracing," *Eurographics*, vol. 87, no. 3, pp. 3–10, 1987.
- [25] T. Bailey, "Constrained initialisation for bearing-only SLAM," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2003, pp. 1966–1971.
- [26] J. Yin, A. Li, T. Li, W. Yu, and D. Zo, "M2DGR: A multi-sensor and multi-scenario SLAM dataset for ground robots," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2266–2273, Apr. 2022.
- [27] D. Zhang et al., "Towards robust sensor-fusion ground SLAM: A comprehensive benchmark and a resilient framework," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Hangzhou, China, 2025, pp. 8894–8901.
- [28] Z. Chen et al., "Heterogeneous LiDAR dataset for benchmarking robust localization in diverse degenerate scenarios," *Int. J. Robot. Res.*, vol. 45, pp. 6–22, 2026.
- [29] Z. Chen et al., "RELEAD: Resilient localization with enhanced LiDAR odometry in adverse environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 3999–4005.