arXiv:2509.15673v5 [cs.RO] 29 Mar 2026

Omni-LIVO: Robust RGB-Colored Multi-Camera Visual-Inertial-LiDAR Odometry via Photometric Migration and ESIKF Fusion

Yinong Cao[1], Chenyang Zhang[1], Xin He[1,*], Yuwei Chen[1], Chengyu Pu[1], Bingtao Wang[1], Kaile Wu[1], Shouzheng Zhu[1], Fei Han[2], Shijie Liu[1], Chunlai Li[1] and Jianyu Wang[1,3]

***Abstract*—Wide field-of-view (FoV) LiDAR sensors provide dense geometry across large environments, but existing LiDAR-inertial-visual odometry (LIVO) systems generally rely on a single camera, limiting their ability to fully exploit LiDAR-derived depth for photometric alignment and scene colorization. We present Omni-LIVO, a tightly coupled multi-camera LIVO system that leverages multi-view observations to comprehensively utilize LiDAR geometric information across extended spatial regions. Omni-LIVO introduces a Cross-View direct alignment strategy that maintains photometric consistency across non-overlapping views, and extends the Error-State Iterated Kalman Filter (ESIKF) with multi-view updates and adaptive covariance. The system is evaluated on public benchmarks and our custom dataset, showing improved accuracy and robustness over state-of-the-art LIVO, LIO, and visual-inertial SLAM baselines.**

***Index Terms*—SLAM, multi-camera system, tightly coupled, visual-inertial-LiDAR fusion, direct visual methods, voxel map.**

I. Introduction

SIMULTANEOUS Localization and Mapping (SLAM) has become essential for autonomous systems operating in unknown environments. Traditional methods are categorized by sensor types: vision-based SLAM [1], [2] provides rich visual information but struggles in low-light or texture-poor scenes, while LiDAR-based SLAM [3] offers robust geometry but degrades in repetitive environments. IMUs, though environment-independent, suffer from long-term drift.

To address single-sensor limitations, multi-sensor fusion systems like visual-inertial SLAM (VI-SLAM) and LiDAR-inertial SLAM (LI-SLAM) enhance robustness through complementary sensor properties [4], [5]. Recent LIVO systems combine dense LiDAR geometry, visual measurements, and high-frequency IMU measurements. LVI-SAM [6] employs feature-based visual tracking, while FAST-LIVO [7] and FAST-LIVO2 [8] achieve high accuracy through photometric patch alignment on LiDAR-derived map points.

Multi-camera configurations offer significant advantages for LIVO systems through extended spatial coverage, enabling comprehensive utilization of LiDAR-derived depth for photometric alignment, richer geometric constraints, improved observability, and enhanced robustness under occlusion and illumination changes. While multi-camera systems have been explored in visual SLAM [9], [10], their tight integration with LiDAR and inertial data in LIVO systems remains underexplored.

Realizing these advantages requires addressing key technical challenges. Unlike stereo systems that use overlapping views for feature matching, non-overlapping multi-camera configurations require different approaches. First, photometric continuity must be maintained through temporal patch migration as patches transition between cameras. Second, adaptive covariance modeling is needed to handle heterogeneous measurement reliability across different views [11], [12]. Directly leveraging LiDAR-derived depth for multi-camera sparse direct alignment maintains efficient performance while enabling accurate visual point selection and extended spatial coverage beyond single-camera constraints.

To address these challenges, we present Omni-LIVO[1], a tightly coupled multi-camera LIVO system that extends FAST-LIVO2 with multi-view photometric constraints.

The main contributions are:

1) **Cross-View temporal migration**: A direct alignment strategy preserving photometric consistency across non-overlapping views.
2) **Adaptive multi-view ESIKF**: Enhanced ESIKF with adaptive covariance for improved robustness in adverse conditions.
3) **Extensive experimental evaluation**: Comprehensive evaluation of the tightly coupled multi-camera LIVO framework on public benchmarks and custom datasets.

II. Related Works

A. LiDAR-Inertial Odometry Evolution

LiDAR-inertial odometry (LIO) systems evolved from feature-based LiDAR odometry like LOAM [3] and LeGO-LOAM [13] to direct methods. FAST-LIO [14] eliminated

Manuscript received: September 2, 2025; Revised: November 9, 2025; Accepted: January 13, 2026.

This paper was recommended for publication by Editor Pascal Vasseur upon evaluation of the Associate Editor and Reviewers comments.

This work was supported by the Zhejiang Provincial "Jianbing Lingyan" R&D Program under Grants 2024C01126, 2023C03012, and 2025002039, and by Hangzhou Institute for Advanced Study, UCAS, under Grants B02006C021026, B02006C021022, and B02006C021029.

*Corresponding author: Xin He (e-mail: xinhe@ucas.ac.cn)

[1]Hangzhou Institute for Advanced Study, University of Chinese Academy of Sciences, Hangzhou 310024, China.

[2]Advanced Laser Technology Lab of Anhui Province, Hefei 230026, China.

[3]Key Laboratory of Space Active Opto-Electronics Technology, Shanghai Institute of Technical Physics, Chinese Academy of Sciences, Shanghai 200083, China

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

[1]Code and data are available at https://github.com/elon876/Omni-LIVO.

feature extraction and directly registered raw points using an iterated Kalman filter. FAST-LIO2 [15] further improved efficiency through incremental map updates with ikd-Tree. However, these systems lack visual information for operation in geometrically degraded environments.

B. LiDAR-Visual-Inertial Integration

LiDAR-visual-inertial fusion addresses pure LIO limitations. R^2LIVE [16] employs sparse visual features with reprojection error minimization. R^3LIVE [17] advanced to direct photometric alignment with RGB-color consistency. FAST-LIVO [7] pioneered tight coupling by attaching image patches to LiDAR map points. FAST-LIVO2 [8] refined this through sequential ESIKF updates with direct methods. However, existing LIVO systems predominantly employ single-camera configurations, limiting comprehensive exploitation of wide-FoV LiDAR geometry across extended spatial coverage.

C. Multi-Camera Visual-Inertial Systems

Multi-camera visual-inertial SLAM has gained attention for extended spatial coverage. MIMC-VINS [18] supports arbitrary camera numbers using multi-state constraint Kalman filtering. BAMF-SLAM [19] integrates multiple fisheye cameras with IMU through bundle adjustment. For large FoV coverage, LF-VIO [20] and LF-VISLAM [21] employ negative imaging plane models for single wide-FoV cameras. MAVIS [10] extends to partially overlapped multi-camera systems. These systems operate in the visual-inertial domain without leveraging LiDAR geometric structure.

D. Direct Visual Methods

Direct visual methods minimize photometric error without feature extraction. DTAM [22], LSD-SLAM [1], and DSO [23] established foundations for dense and sparse tracking. MultiCol-SLAM [24] extended direct methods to multi-camera systems. This suits multi-camera LIVO by avoiding correspondence complexity in non-overlapping views.

III. System Overview And Methodology

The Omni-LIVO system comprises four modules: Sensor Acquisition, Preprocessing, Multi-Camera ESIKF, and Unified Mapping, as illustrated in Fig. 1. The processing pipeline begins with IMU propagation, followed by sequential ESIKF updates that integrate LiDAR point-to-plane residuals with multi-view photometric constraints. Cross-View temporal migration and adaptive covariance preserve photometric consistency across non-overlapping views.

A. Notational Conventions

Table I summarizes the key mathematical notation used throughout this paper.

TABLE I
Notational Conventions

Symbol	Description
$\mathbf{T}_c^w$	Transformation from world to camera frame
$\mathbf{R}$, $\mathbf{t}$	Rotation matrix and translation vector
$\mathbf{p}$	3D point in world coordinates
τ_c	Inverse exposure time parameter for camera c
N_c	Number of cameras
$\mathcal{P}_k^{\text{seg}}$	Temporally segmented LiDAR point cloud
$\mathcal{C}_k^{(i)}$	Image frame from camera i at time k
$\mathcal{V}_k$	Voxel containing geometric and visual data
$\mathbf{n}_k$	Plane normal vector
$\mathbf{r}_{VIO}$	Visual-inertial odometry residual vector
$\mathbf{r}_{LIO}$	LiDAR-inertial odometry residual
$\hat{\mathbf{x}}^-$, $\mathbf{P}^-$	Prior state estimate and covariance
$\hat{\mathbf{x}}_{k\|k}$, $\mathbf{P}_{k\|k}$	Posterior state estimate and covariance
$\mathbf{H}$	Jacobian matrix
$\mathbf{K}$	Kalman gain
ℓ	Iteration index
$\boxplus$	Manifold addition operator

B. Multi-Sensor Data Alignment

Building upon FAST-LIVO2 [8], we extend temporal alignment to multi-camera configurations [25]. The system forms temporally coherent measurement packets:

$$\mathcal{M}_{\mathrm{k}} = \{\mathcal{P}_{\mathrm{k}}^{\text{seg}}, \{\mathcal{J}_{\mathrm{j}}\}_{\mathrm{j=k-1}}^{\mathrm{k}}, \{\mathcal{C}_{\mathrm{k}}^{(\mathrm{i})}\}_{\mathrm{i=1}}^{N_c}\} \tag{1}$$

where $\mathcal{M}_k$ denotes the multi-sensor measurement packet at time k, and $\{\mathcal{J}_{\mathrm{j}}\}_{\mathrm{j=k-1}}^{\mathrm{k}}$ represents the set of high-frequency IMU measurements spanning the inter-frame interval.

Although LiDAR and cameras may operate at the same nominal rate (e.g., 10 Hz), their trigger events are not always aligned. To align data across modalities, each LiDAR point p_i with timestamp τ_i is assigned to the temporally nearest camera frame:

$$p_i \rightarrow \text{frame}_k, \quad \text{with } k = \arg\min_j |\tau_i - t_{\text{sync}}^{(j)}| \tag{2}$$

where $t_{\text{sync}}^{(j)}$ is the sync trigger time. Within each frame k, all camera channels are externally triggered to ensure simultaneous exposure:

$$t_{\text{cam}}^{(i,k)} = t_{\text{sync}}^{(k)}, \quad \forall i \in [1, N_c] \tag{3}$$

where $t_{\text{cam}}^{(i,k)}$ is the trigger time for camera i at frame k.

C. Cross-View Direct Method

We formalize Cross-View temporal migration to maintain photometric continuity across non-overlapping views (Fig. 3). The migration indicator $M(\mathbf{p}, i \rightarrow j)$ determines whether a map point transitions from camera i to j:

$$M(\mathbf{p}, \mathrm{i} \rightarrow \mathrm{j}) = 1 \Leftrightarrow \begin{cases} \text{cam}_{\text{ref}}(\mathbf{p}) = \mathrm{i} \\ \text{cam}_{\text{cur}}(\mathbf{p}) = \mathrm{j} \\ \mathrm{i} \neq \mathrm{j} \\ \text{exclusive}_{\text{visible}}(\mathbf{p}, \mathrm{j}) = 1 \\ \text{temporal}_{\text{migrated}}(\mathbf{p}, \mathrm{i} \rightarrow \mathrm{j}) = 1 \end{cases} \tag{4}$$

where $\text{cam}_{\text{ref}}(\mathbf{p})$ and $\text{cam}_{\text{cur}}(\mathbf{p})$ return the camera IDs observing $\mathbf{p}$ in reference and current frames, $\text{exclusive}_{\text{visible}}(\mathbf{p}, \mathrm{j}) = 1$

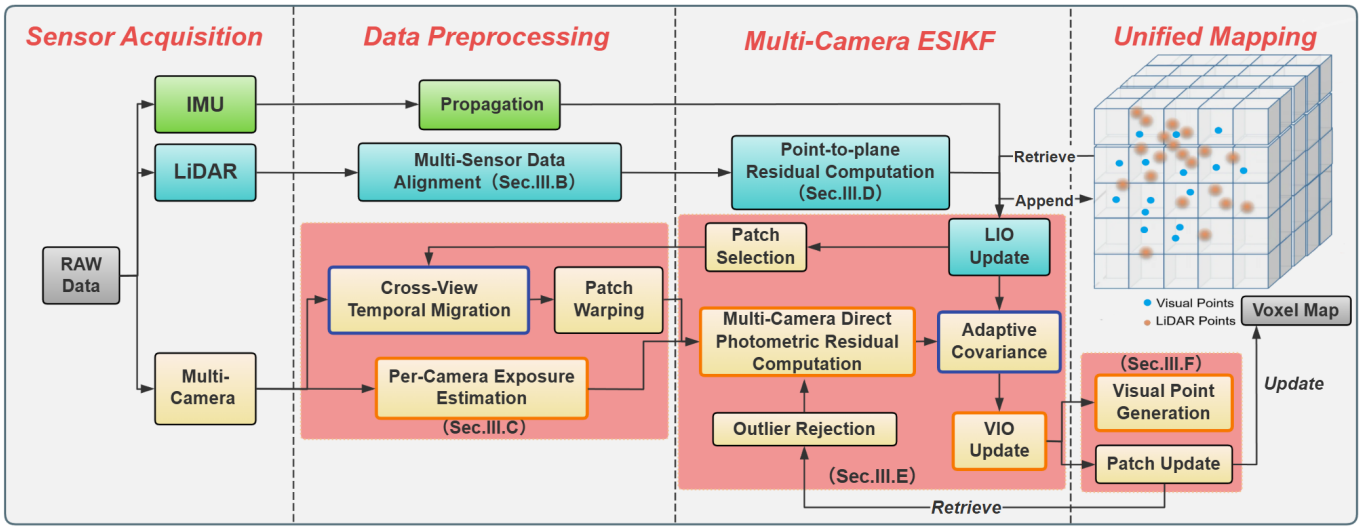


Fig. 1. Overview of the Omni-LIVO architecture. Light yellow, light blue, and light green modules represent visual, LiDAR, and IMU data respectively. Orange borders indicate extensions to baseline methods; blue borders indicate novel contributions.

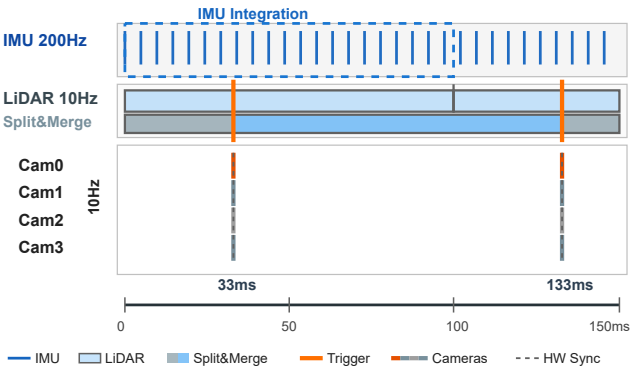


Fig. 2. Multi-sensor temporal alignment. LiDAR points are associated with the nearest camera frame, while IMU measurements provide continuous motion priors for temporally consistent fusion.

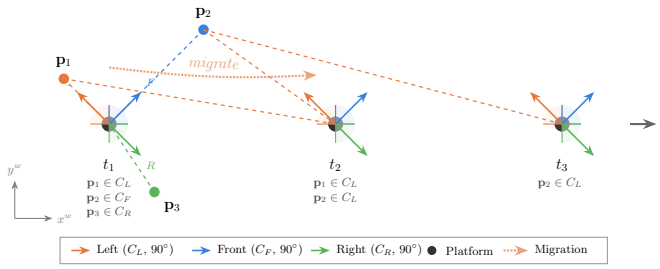


Fig. 3. Cross-View temporal migration. At t_1, map points $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$ are observed by Left (C_L), Front (C_F), and Right (C_R) cameras respectively. As the platform moves, points migrate between camera views ($\mathbf{p}_2$ migrates from C_F to C_L at t_2), maintaining photometric continuity across non-overlapping 90° FoVs through Cross-View residuals.

ensures $\mathbf{p}$ is visible only in camera j (i.e., $\pi_j(\mathbf{T}_j^w \cdot \mathbf{p}) \in \Omega_j$ with Ω_j being the valid image region, and invisible in other cameras), and $\text{temporal}_{\text{migrated}}(\mathbf{p}, \text{i} \rightarrow \text{j}) = 1$ verifies $\mathbf{p}$ transitioned from camera i to j across successive frames.

When camera migration occurs, the reference patch is dynamically switched by prioritizing observations from the current camera, or selecting the patch with minimum photometric error:

$$\mathbf{p}_{\text{ref}}^* = \arg\min_{\mathbf{p}_k \in \mathcal{O}(\mathbf{p})} \frac{1}{|\mathcal{O}(\mathbf{p})| - 1} \sum_{\mathbf{p}_m \in \mathcal{O}(\mathbf{p}) \setminus \{\mathbf{p}_k\}} \|\mathbf{I}_k - \mathbf{I}_m\|^2 \quad (5)$$

where $\mathcal{O}(\mathbf{p})$ denotes the set of all observations of point $\mathbf{p}$, $\mathbf{I}_k \in \mathbb{R}^P$ is the vectorized patch intensity, and P is the patch size.

To ensure consistent intensity comparison across cameras, image measurements are normalized by vignetting:

$$I'_k(u, v) = V_{\text{k}}(r^2) \cdot I_{\text{k}}(u, v) \quad (6)$$

where $I_k(u, v)$ denotes the raw image intensity at pixel coordinates (u, v), $I'_k(u, v)$ is the vignetting-normalized intensity, r is the radial distance from the image center, and $V_k(r^2)$ is a vignetting function modeled as:

$$V_k(r^2) = 1 + \sum_{n=1}^{3} \beta_{k,n} \cdot r^{2n} \quad (7)$$

where $\beta_{k,n}$ are the vignetting polynomial coefficients calibrated offline for each camera.

Exposure variations across cameras are handled through per-camera inverse exposure time parameters τ_c jointly optimized in the ESIKF framework, ensuring photometric consistency without requiring relative exposure factor estimation.

The relative transformation and induced homography under local planar assumption are formulated to warp patches between cameras. The relative transformation from camera i to camera j is:

$$\mathbf{T}_{\text{i}\rightarrow\text{j}}(t) = \mathbf{T}_{\text{j}}^{\text{w}}(t_{\text{j}}) \cdot (\mathbf{T}_{\text{i}}^{\text{w}}(t_{\text{i}}))^{-1} \quad (8)$$

Under the local planar assumption, the homography matrix $\mathbf{H}_{i\rightarrow j}(\mathbf{p})$ that maps points from camera i to camera j is [26]:

$$\mathbf{H}_{\text{i}\rightarrow\text{j}}(\mathbf{p}) = \mathbf{R}_{\text{i}\rightarrow\text{j}} + \frac{\mathbf{t}_{\text{i}\rightarrow\text{j}} \mathbf{n}_{\text{i}}^{\text{T}}}{d_{\text{i}}} \quad (9)$$

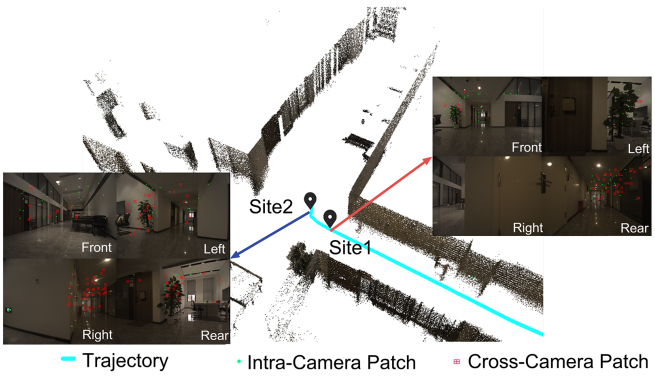


Fig. 4. Photometric patch associations during turning: intra-camera patches (green) remain within the same view, while migrated patches (red) form Cross-View associations across different cameras.

where $\mathbf{R}_{\mathrm{i}\to\mathrm{j}}$ and $\mathbf{t}_{\mathrm{i}\to\mathrm{j}}$ are the rotation and translation from $\mathbf{T}_{\mathrm{i}\to\mathrm{j}}$, $\mathbf{n}_{\mathrm{i}}$ is the unit surface normal at the patch in camera i frame, and d_{i} is the point depth in camera i frame. This homography enables direct patch warping across non-overlapping camera views for photometric alignment.

Based on these definitions, we formulate both intra-camera and Cross-View residuals, with the patch association process visualized in Fig. 4. For intra-camera alignment, the direct photometric residual r_{direct} is:

$$r^{\mathrm{c}}_{\text{direct}}(\mathbf{p}) = \tau_c \cdot I'_c(\pi_c(\mathbf{T}^w_c \cdot \mathbf{p})) - \tau_{\text{ref}} \cdot I'_{\text{ref}}(\pi_{\text{ref}}(\mathbf{p}_{\text{ref}})) \quad (10)$$

where $\pi_c(\cdot)$ denotes the camera projection function from 3D to 2D image coordinates, and subscript "ref" denotes the reference frame.

Cross-View temporal migration residuals $r_{\text{migration}}$ establish constraints between different cameras across time:

$$r^{i\to j}_{\text{migration}}(p) = \tau_i \cdot W_H\left[I'_i, \pi_i(p, t_i)\right] - \tau_j \cdot I'_j(\pi_j(p, t_j)) \quad (11)$$

where camera i is the source (reference) and camera j is the target (current), W_H represents warping using homography H, and t_i, t_j are the respective camera timestamps.

For cross-camera constraints, we formulate bidirectional coupled Jacobians that account for geometric contributions from both cameras:

$$\mathbf{J}^{\text{coupled}}_{\mathbf{R}} = \mathbf{J}^i_{\mathbf{R}} - \mathbf{J}^j_{\mathbf{R}}, \quad \mathbf{J}^{\text{coupled}}_{\mathbf{t}} = \mathbf{J}^i_{\mathbf{t}} - \mathbf{J}^j_{\mathbf{t}} \quad (12)$$

where $\mathbf{J}^k_{\mathbf{R}} = \tau_k \frac{\partial I_k}{\partial \mathbf{R}}$ and $\mathbf{J}^k_{\mathbf{t}} = \tau_k \frac{\partial I_k}{\partial \mathbf{t}}$ are exposure-weighted Jacobians with respect to rotation and translation for camera k. Similarly, exposure Jacobians are computed as $\frac{\partial r^{i\to j}}{\partial \tau_i} = I_i$ and $\frac{\partial r^{i\to j}}{\partial \tau_j} = -I_j$. This bidirectional formulation enforces both geometric and photometric consistency between camera pairs.

The complete residual vector $\mathbf{r}_{VIO}$ stacks all intra-camera and Cross-View residuals for joint optimization:

$$\mathbf{r}_{VIO} = \left[\mathbf{r}^1_{\text{direct}}, \ldots, \mathbf{r}^{N_c}_{\text{direct}}, \mathbf{r}^{1\to 2}_{\text{migration}}, \ldots, \mathbf{r}^{i\to j}_{\text{migration}}\right]^T \quad (13)$$

Temporal migration constraints preserve photometric continuity across cameras, enhancing geometric diversity and pose estimation robustness. Multi-camera configurations enable stricter photometric thresholds while maintaining sufficient observations, improving both efficiency and accuracy.

D. LiDAR Geometric Constraint Processing

LiDAR measurements provide structural constraints that complement photometric observations in our multi-camera framework. We employ voxel-based plane extraction to efficiently process point clouds and generate geometric residuals.

The raw LiDAR point cloud is discretized into voxels of resolution v_{size}. For each voxel $\mathcal{V}_k$ containing N_k points $\mathcal{P}_k = \{p_i\}_{i=1}^{N_k}$, we compute the voxel centroid $\bar{p}_k = \frac{1}{N_k}\sum_{i=1}^{N_k} p_i$ and the sample covariance matrix $\mathbf{C}_k$:

$$\mathbf{C}_k = \frac{1}{N_k}\sum_{i=1}^{N_k}(p_i - \bar{p}_k)(p_i - \bar{p}_k)^{\mathrm{T}} \quad (14)$$

Planar voxels are identified via eigen-decomposition of $\mathbf{C}_k$. If the smallest eigenvalue $\lambda_{\min} < \epsilon_{\text{plane}}$ (where ϵ_{plane} is the planarity threshold), the corresponding eigenvector defines the plane normal $\mathbf{n}_k$, which satisfies the plane equation for any point p on the plane:

$$\mathbf{n}^{\mathrm{T}}_k(p - \bar{p}_k) = 0 \quad (15)$$

LiDAR geometric residuals are constructed by measuring point-to-plane distances. Let $\mathbf{p}^{\mathrm{w}}$ denote a LiDAR point in world frame and $\mathbf{T}^{\mathrm{w}}_{\mathrm{L}}$ the world-to-LiDAR transformation. The point in LiDAR frame is $\mathbf{p}^{\mathrm{L}} = \mathbf{R}^{\mathrm{w}}_{\mathrm{L}}\mathbf{p}^{\mathrm{w}} + \mathbf{t}^{\mathrm{w}}_{\mathrm{L}}$, and the LiDAR-inertial residual $\mathbf{r}_{LIO}$ measuring point-to-plane distance is:

$$\mathbf{r}_{LIO}(\mathbf{p}^{\mathrm{w}}) = \mathbf{n}^{\mathrm{T}}_k(\mathbf{R}^{\mathrm{w}}_{\mathrm{L}}\mathbf{p}^{\mathrm{w}} + \mathbf{t}^{\mathrm{w}}_{\mathrm{L}} - \bar{\mathbf{p}}_k) \quad (16)$$

where $\bar{\mathbf{p}}_k$ is the plane centroid in LiDAR frame.

E. Iterative Multi-Camera State Estimation with ESIKF

Measurement quality varies across cameras due to heterogeneous exposure and environments. We extend ESIKF to integrate multi-view photometric constraints with LiDAR measurements through sequential update and adaptive covariance, as shown in Fig. 5. The state is first updated with LiDAR point-to-plane residuals, then sequentially updated with multi-camera photometric residuals. Multi-camera residuals and Jacobians are stacked across all cameras and processed through iterative ESIKF updates until convergence.

For heterogeneous exposure settings, independent inverse exposure time parameters τ_c are maintained for each camera $c \in [1, N_c]$ and jointly optimized within the ESIKF framework. The Jacobian matrix is extended to $\mathbb{R}^{m\times(6+N_c)}$, where m is the total number of measurements, 6 represents pose parameters (rotation and translation), and N_c represents per-camera exposure parameters.

The photometric covariance matrix R_{img} is adaptively scaled based on real-time photometric error to account for varying measurement quality. The covariance scaling factor $\sigma^{(t)}$ at time t is computed via linear mapping:

$$\sigma^{(t)} = \sigma_{\min} + \frac{\bar{e}^{(t-1)} - 1}{e_{\max} - 1} \cdot (\sigma_{\max} - \sigma_{\min}) \quad (17)$$

where $\bar{e}^{(t-1)}$ is the mean squared photometric error from the previous frame, $\sigma_{\min}$ and $\sigma_{\max}$ are the minimum and maximum covariance scaling bounds (set to 100.0 and 1000.0

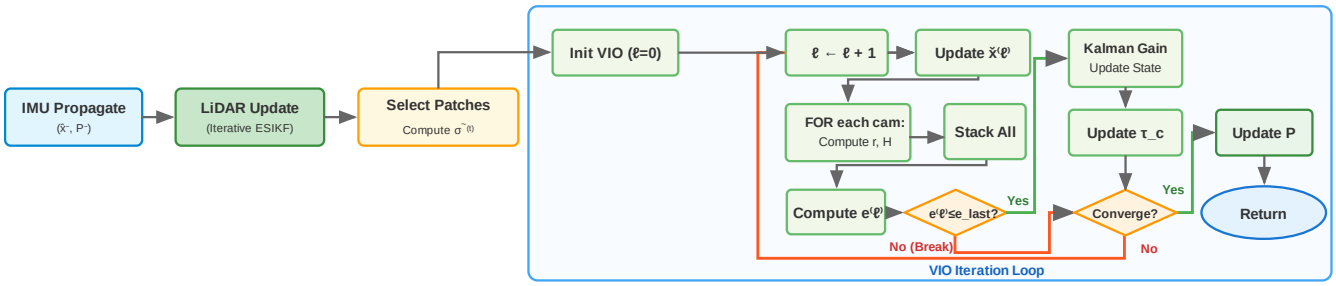


Fig. 5. Iterative Multi-Camera ESIKF Framework. The pipeline performs Synchronize , IMU Propagate , LiDAR Update, Select Patches, and Adaptive Cov, followed by Init VIO and iterative updates of the photometric VIO loop until convergence.

respectively), and $e_{\max}$ is the maximum error threshold for scaling (typically set to 100.0). Temporal smoothing ensures stability:

$$\tilde{\sigma}^{(t)} = \lambda \cdot \sigma^{(t)} + (1-\lambda) \cdot \tilde{\sigma}^{(t-1)} \tag{18}$$

where $\tilde{\sigma}^{(t)}$ is the smoothed covariance scaling factor and λ is the smoothing parameter (set to 0.3). This adaptive weighting down-weights cameras with high photometric errors (e.g., motion blur, occlusion) and up-weights reliable observations, improving robustness in adverse conditions.

Algorithm 1 summarizes the complete implementation. Algorithm-specific notation: $\breve{\mathbf{x}}^{(\ell)}$ is the linearization point; $\mathbf{r}_c^{(\ell)}$, $\mathbf{H}_c^{(\ell)}$ are per-camera residuals and Jacobians stacked into $\mathbf{r}_{VIO}^{(\ell)}$, $\mathbf{H}_{VIO}^{(\ell)}$; $\delta\hat{\mathbf{x}}^{(\ell)}$ is the state increment; $e^{(\ell)}$ is the mean squared error; $\ell_{\max} = 10$ is the maximum iteration count.

Algorithm 1: Multi-Camera ESIKF Update

IMU propagation: $\hat{\mathbf{x}}^-$, $\mathbf{P}^-$; LiDAR plane extraction
LiDAR update: Iterative ESIKF with $\mathbf{r}_{LIO}$, $\mathbf{H}_{LIO}$
Select patches from cameras $c \in [1, N_c]$; Compute $\tilde{\sigma}^{(t)}$ via Eq. (17)-(18)
VIO update: $\ell \leftarrow 0$, $\delta\hat{\mathbf{x}}^{(0)} \leftarrow \mathbf{0}$, $e_{\text{last}} \leftarrow \infty$
repeat
  $\ell \leftarrow \ell + 1$; Update $\breve{\mathbf{x}}^{(\ell)} = \hat{\mathbf{x}}^- \boxplus \delta\hat{\mathbf{x}}^{(\ell-1)}$
  for *camera* $c = 1, \ldots, N_c$ **do**
    Compute $\mathbf{r}_c^{(\ell)}$, $\mathbf{H}_c^{(\ell)}$ with cross-camera constraints
  Stack $\mathbf{r}_{VIO}^{(\ell)} = [\mathbf{r}_1^{(\ell)}; \ldots; \mathbf{r}_{N_c}^{(\ell)}]$, $\mathbf{H}_{VIO}^{(\ell)} = [\mathbf{H}_1^{(\ell)}; \ldots; \mathbf{H}_{N_c}^{(\ell)}]$
  Compute $e^{(\ell)} = \|\mathbf{r}_{VIO}^{(\ell)}\|^2 / n_{\text{meas}}$
  if $e^{(\ell)} \leq e_{last}$ **then**
    $e_{\text{last}} \leftarrow e^{(\ell)}$
    $\mathbf{K} = ((\mathbf{H}_{VIO}^{(\ell)})^{\mathrm{T}} \mathbf{H}_{VIO}^{(\ell)} + \mathbf{P}^{-1}/\tilde{\sigma}^{(t)})^{-1}$
    $\delta\hat{\mathbf{x}}^{(\ell)} = \mathbf{K}((\mathbf{H}_{VIO}^{(\ell)})^{\mathrm{T}} \mathbf{r}_{VIO}^{(\ell)} - \mathbf{P}^{-1}\delta\hat{\mathbf{x}}^{(\ell-1)}/\tilde{\sigma}^{(t)})$
    $\hat{\mathbf{x}}^- \leftarrow \hat{\mathbf{x}}^- \boxplus \delta\hat{\mathbf{x}}^{(\ell)}$
  else
    break
until $\|\delta\mathbf{R}^{(\ell)}\| < 0.001°$ *and* $\|\delta\mathbf{t}^{(\ell)}\| < 0.01mm$ *or* $\ell \geq \ell_{\max}$
$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}\mathbf{H}_{VIO}^{(\ell)})\mathbf{P}^-$
return $\hat{\mathbf{x}}_{k|k}$, $\mathbf{P}_{k|k}$

F. Unified Voxel Mapping with Multi-View Integration

We integrate LiDAR geometry and multi-view photometric patches within a single adaptive volumetric structure employing density-driven octree refinement [27]. Each voxel $\mathcal{V}_k$ encapsulates three components:

$$\mathcal{V}_k = \{\mathcal{G}_k, \mathcal{W}_k, \mathcal{S}_k\} \tag{19}$$

where $\mathcal{G}_k$ represents geometric plane parameters from LiDAR, $\mathcal{W}_k$ contains visual map points with multi-view observations, and $\mathcal{S}_k$ stores octree subdivision structure. Visual map points are generated by partitioning images into regular grids and selecting LiDAR points with highest Shi-Tomasi corner response. Each visual point j maintains a feature structure $\mathcal{F}_j$ containing N_{obs} observations across multiple frames:

$$\mathcal{F}_j = \{\mathcal{P}_j^{(k)}, \mathrm{T}_{wc}^{(k)}, \tau^{(k)}, id_{\text{cam}}^{(k)}\}_{k=1}^{N_{\text{obs}}} \tag{20}$$

where N_{obs} is the number of observations, $\mathcal{P}_j^{(k)}$ is the patch organized in multi-level pyramid structure, $\mathrm{T}_{wc}^{(k)}$ is the camera pose, $\tau^{(k)}$ is the exposure parameter, and $id_{\text{cam}}^{(k)}$ is the camera identifier.

IV. Experimental Evaluation

We evaluate the proposed system on public benchmarks and custom datasets, and compare with state-of-the-art methods.

A. Datasets and Baselines

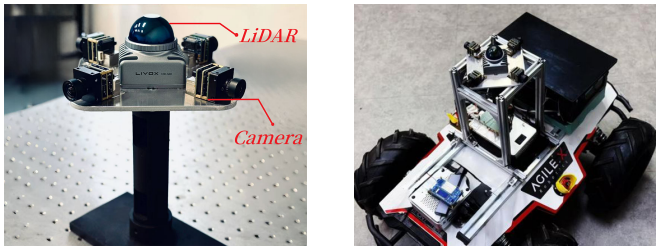
To demonstrate the effectiveness of the proposed multi-camera LIVO framework, we conduct comprehensive experiments evaluating trajectory accuracy, ablation of key components, computational efficiency, and qualitative mapping performance across diverse environments.

Evaluation is conducted on Hilti SLAM Challenge 2022/2023 and Newer College Dataset. We additionally collected a custom dataset comprising multiple sequences using LIVOX MID360 LiDAR (360° horizontal FoV, 10Hz) and four cameras (1024×768, 10Hz) in cross-pattern configuration with handheld and robotic platforms (Fig. 6).

We compare against five methods: FAST-LIVO2, FAST-LIVO, R^{3}LIVE (LiDAR-inertial-visual), FAST-LIO2 (LiDAR-inertial), and OpenMAVIS (multi-camera visual-inertial). All LIVO systems use 10Hz for both camera and LiDAR data acquisition. FAST-LIVO2 and our system use identical LIO and IMU parameters. Ablation studies are conducted using five system variants: *w/o ac* (Adaptive Covariance disabled), *w/o cvtm* (Cross-View Temporal Migration disabled), *1-Cam*, *2-Cam*, and *Ours* (complete system with three cameras).

TABLE II
ABSOLUTE TRAJECTORY ERRORS (RMSE IN METERS) ON MULTIPLE BENCHMARK DATASETS

Dataset	Sequence	Baseline Methods					Omni-LIVO Variants				
		FAST-LIVO	FAST-LIVO2	R^{3}LIVE	FAST-LIO2	OpenMAVIS	*w/o ac*	*w/o cvtm*	1-Cam	2-Cam	Ours
Hilti 2022	Attic to Upper Gallery	1.650	0.150	×	×	0.103	0.082	0.089	0.176	0.085	**0.069**
	Attic to Upper Gallery 2	1.165	0.123	×	3.591	0.255	0.129	0.141	0.122	0.134	**0.120**
	Basement 2	0.129	0.043	0.097	0.136	0.082	0.038	0.036	0.049	**0.031**	**0.031**
	Construction Ground	0.016	**0.010**	0.015	0.018	0.094	**0.010**	0.011	0.011	**0.010**	0.011
	Construction Multilevel	0.155	**0.020**	0.021	0.038	0.933	0.021	**0.020**	0.022	**0.020**	**0.020**
	Construction Stairs	0.291	0.039	0.611	0.625	0.054	**0.028**	0.032	0.042	0.031	0.030
	Construction Upper 1	0.032	0.030	0.028	0.124	0.083	0.029	0.029	0.029	0.029	**0.027**
	Construction Upper 2	0.089	0.041	0.041	0.100	0.040	0.019	0.020	0.037	0.022	**0.009**
	Construction Upper 3	0.091	0.045	0.049	×	0.075	0.027	0.025	0.047	0.025	**0.011**
	Cupola	0.270	0.130	×	×	0.140	0.133	0.129	0.144	0.131	**0.126**
	Cupola 2	×	0.185	×	×	0.136	0.127	0.146	0.187	0.121	**0.098**
	Long Corridor	0.072	0.056	0.069	0.077	0.098	0.052	0.055	0.056	0.053	**0.051**
	Lower Gallery	0.065	0.033	0.010	0.056	0.064	0.009	0.011	0.021	0.011	**0.006**
	Lower Gallery 2	0.387	0.155	×	×	0.178	0.109	0.119	0.158	0.108	**0.083**
	Outside Building	0.047	0.031	0.033	0.029	0.425	0.020	0.018	0.022	**0.016**	**0.016**
Hilti 2023	Floor 0	0.030	0.023	0.045	0.068	0.060	0.021	**0.020**	0.030	0.024	**0.020**
	Floor 1	0.029	0.020	0.045	0.055	0.127	0.018	0.019	0.020	0.019	**0.017**
	Floor 2	0.093	0.022	0.121	0.121	0.178	**0.012**	0.022	0.020	0.018	**0.012**
	Basement	0.072	0.019	0.070	0.190	0.246	0.021	0.020	0.020	**0.016**	**0.016**
	Stairs	0.230	0.020	0.202	0.154	0.225	0.010	0.013	0.019	0.013	**0.008**
Newer College	Quad easy	0.099	0.082	0.086	0.092	0.084	0.077	0.081	0.080	0.077	**0.071**
	Quad medium	1.530	0.073	**0.061**	0.101	0.113	0.069	0.074	0.075	0.070	0.065
	Quad hard	1.757	0.083	0.071	0.146	0.124	0.073	0.075	0.075	0.072	**0.057**
	Stairs	0.124	0.057	0.224	×	×	0.049	0.051	**0.046**	0.061	0.048
	Underground easy	0.125	0.049	**0.037**	0.156	0.120	0.051	0.050	0.049	0.051	0.049
	Underground medium	0.088	0.043	0.047	0.153	0.130	0.037	0.041	0.045	0.039	**0.034**
	Underground hard	0.157	0.241	0.069	0.141	0.255	0.068	0.072	0.096	0.061	**0.055**



(a) Handheld device setup (b) Robotic platform setup

Fig. 6. SLAM data acquisition platforms used in our custom dataset. (a) Handheld configuration. (b) Robotic configuration.

B. Quantitative Results

Table II summarizes absolute trajectory errors (bold: best, underline: second-best) across all sequences. Omni-LIVO variants achieve the lowest error in 25 out of 27 sequences and successfully complete all evaluated tracks, outperforming FAST-LIVO2, R^{3}LIVE, FAST-LIO2, and OpenMAVIS across diverse environments including construction sites, indoor corridors, and underground scenarios. Compared to FAST-LIVO2, our system achieves an average 30% error reduction across all 27 sequences, with improvements up to 82% on Lower Gallery.

Ablation analysis demonstrates that Cross-View Temporal Migration improves accuracy in sequences with large viewpoint changes (Cupola 2: 0.098 m vs. 0.146 m *w/o cvtm*), while Adaptive Covariance enhances robustness under illumination variations (Lower Gallery 2: 0.083 m vs. 0.109 m *w/o ac*). Camera number comparison shows progressive gains from 1-Cam to 3-Cam (Ours), with the largest improvements in complex environments (Underground hard: 0.096 m to 0.055 m). This trend indicates that multi-camera fusion provides greater benefits in geometrically challenging scenarios.

C. Computational Performance

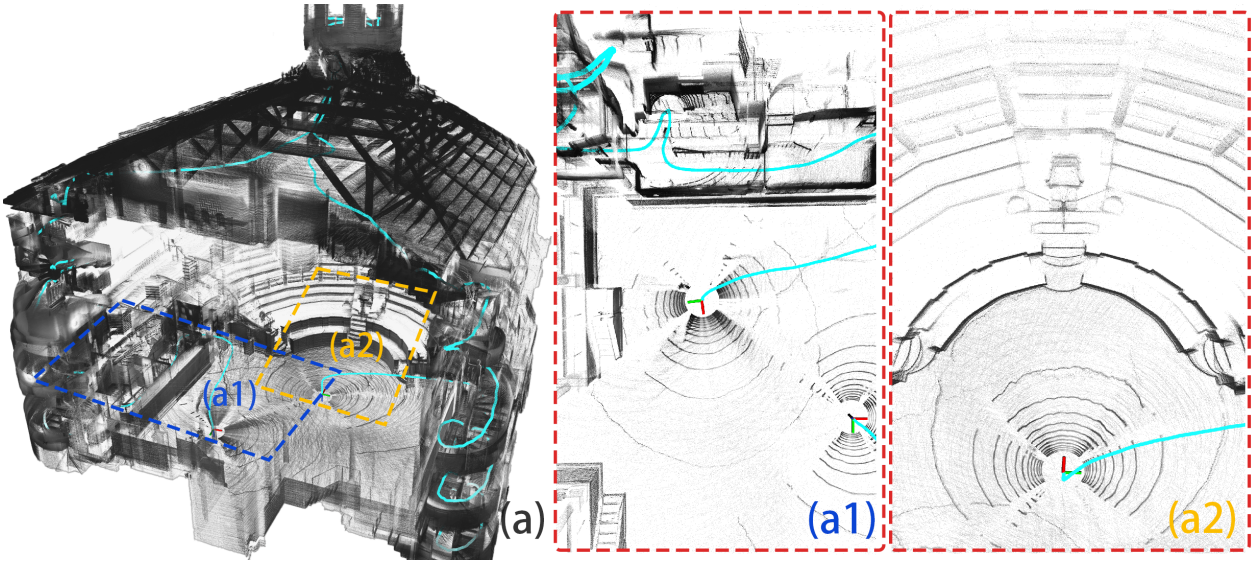
Computational complexity is $\mathcal{O}(N_p \cdot P \cdot L)$ for single-camera and $\mathcal{O}((N_p + N_{\text{cross}}) \cdot P \cdot L)$ for multi-camera, where N_p is visual points, P is patch size, L is pyramid levels, and N_{cross} is cross-camera observations.

We use NVIDIA Jetson AGX Orin with 12-core ARM CPU (2.2GHz) and 64GB RAM running Ubuntu 20.04. System parameters: LiDAR voxel size 1.0 m, surface filtering threshold 0.5 m, maximum 150 visual points per frame.

Table III presents processing time and memory consumption. Compared to FAST-LIVO2, Omni-LIVO introduces 1.16× overhead on Hilti (60.3 ms vs. 52.1 ms, 3 cameras) and 1.21× on custom sequences (40.6 ms vs. 33.6 ms, 4 cameras), primarily due to cross-camera constraint computation, while maintaining real-time 10Hz performance with 2906 MB (Hilti) and 2359 MB (Custom) memory consumption, demonstrating practical viability for embedded platforms.

D. Qualitative Results

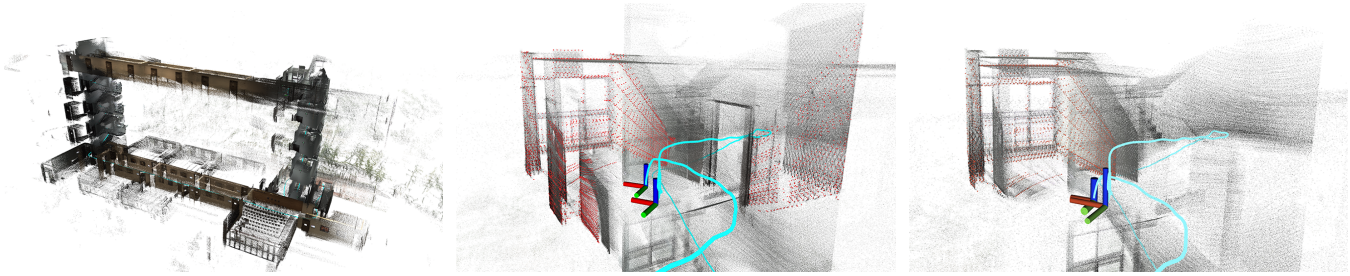
Fig. 7 demonstrates Omni-LIVO's mapping quality on 2 sequences with multi-camera configurations. Fig. 8 validates the system's ability to maintain global consistency and achieve precise loop closure, with Omni-LIVO showing superior alignment compared to FAST-LIVO2's visible drift. The improved consistency is attributed to enhanced observability from multi-view constraints. Table IV shows that Omni-LIVO produces



(a) Cupola 2

(b) Underground Hard

Fig. 7. Mapping quality comparison on two challenging sequences. (a) Cupola 2 scene with overall map and detailed regions. (b) Underground Hard scene with overall map and detailed regions. The colored regions are grayscale LiDAR point clouds colorized using multi-camera gray images.



(a) Overall map (b) Omni-LIVO: Start/End alignment (c) FAST-LIVO2: Start/End misalignment

Fig. 8. Loop closure comparison on staircase sequence. (a) Overall map. (b) Omni-LIVO: precise start/end alignment. (c) FAST-LIVO2: visible drift.

more RGB-colored points than competing methods, with particularly significant improvements in complex environments (3.5× more than FAST-LIVO2 in basement3).

E. Discussion

Experimental results show that Omni-LIVO achieves improved accuracy over baseline methods across evaluated sequences with moderate computational overhead. The multi-camera configuration provides extended spatial coverage and denser scene colorization. Ablation studies demonstrate that Cross-View constraints improve performance in sequences

TABLE III
COMPUTATIONAL PERFORMANCE COMPARISON

Sequence	#Cam	FAST-LIVO2		Omni-LIVO	
		Time (ms)	Mem (MB)	Time (ms)	Mem (MB)
Construction Upper	3	50.7	1901	58.1	2017
Cupola 2	3	51.2	3477	57.0	3498
Outside Building	3	54.5	3261	65.9	3203
Avg. (Hilti)	3	52.1	2880	60.3	2906
Corridor	4	17.3	1054	21.0	1197
Basement3	4	49.9	3473	60.1	3520
Avg. (Custom)	4	33.6	2264	40.6	2359

TABLE IV
RGB POINT CLOUD STATISTICS (NUMBER OF COLORED POINTS)

Method	stairs	classroom	basement3	corridor
Omni-LIVO	**14,184,500**	**4,484,889**	**33,513,608**	**8,715,790**
FAST-LIVO2	5,539,700	3,111,504	9,519,125	4,863,001
FAST-LIVO	3,827,955	1,611,539	5,019,200	2,543,585
R^3LIVE	393,242	619,310	1,213,929	1,007,194

with large viewpoint changes, while adaptive covariance enhances robustness under illumination variations.

V. CONCLUSION

This letter presents Omni-LIVO, a tightly coupled multi-camera LiDAR-inertial-visual odometry system that extends direct visual methods to multi-view configurations through Cross-View temporal migration and adaptive ESIKF integration. Evaluation on public benchmarks and custom datasets shows that the system achieves lower trajectory errors compared to state-of-the-art LIVO, LIO, and visual-inertial methods. The multi-camera configuration provides extended spatial coverage and produces denser RGB-colored point clouds. The quantitative and qualitative improvements demonstrate the effectiveness of the proposed multi-camera fusion approach. Future work will explore online loop closure and investigate multi-sensor calibration refinement.

REFERENCES

[1] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Zurich, Switzerland, Sep. 2014, pp. 834–849, doi: 10.1007/978-3-319-10605-2_54.

[2] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017, doi: 10.1109/TRO.2017.2705103.

[3] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot.: Sci. Syst.*, Berkeley, CA, USA, Jul. 2014, pp. 1–9, doi: 10.15607/RSS.2014.X.007.

[4] 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, doi: 10.1109/TRO.2018.2853729.

[5] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, Dec. 2021, doi: 10.1109/TRO.2021.3075644.

[6] T. Shan, B. Englot, C. Ratti, and D. Rus, "LVI-SAM: Tightly-coupled Lidar-Visual-Inertial Odometry via Smoothing and Mapping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Xi'an, China, May 2021, pp. 5692–5698.

[7] C. Zheng, Q. Zhu, W. Xu, X. Liu, Q. Guo, and F. Zhang, "FAST-LIVO: Fast and tightly-coupled sparse-direct LiDAR-inertial-visual odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Kyoto, Japan, Oct. 2022, pp. 4003–4009, doi: 10.1109/IROS47612.2022.9981107.

[8] C. Zheng, W. Xu, Q. Guo, and F. Zhang, "FAST-LIVO2: Fast, direct LiDAR-inertial-visual odometry," *IEEE Trans. Robot.*, vol. 41, pp. 326–346, 2025, doi: 10.1109/TRO.2024.3502198.

[9] J. Kuo, M. Muglikar, Z. Zhang, and D. Scaramuzza, "Redesigning SLAM for arbitrary multi-camera systems," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Paris, France, May 2020, pp. 2116–2122, doi: 10.1109/ICRA40945.2020.9196730.

[10] Y. Wang, Y. Ng, I. Sa, A. Parra, C. Rodriguez-Opazo, T. Lin, and H. Li, "MAVIS: Multi-camera augmented visual-inertial SLAM using $SE_2(3)$-based exact IMU pre-integration," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Yokohama, Japan, May 2024, pp. 1694–1700, doi: 10.1109/ICRA57147.2024.10609982.

[11] L. T. Zhang, D. Wisth, M. Camurri, and M. Fallon, "Balancing the budget: Feature selection and tracking for multi-camera visual-inertial odometry," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1182–1189, Apr. 2022, doi: 10.1109/LRA.2021.3137910.

[12] A. Fontan, L. Oliva, J. Civera, and R. Triebel, "Model for multi-view residual covariances based on perspective deformation," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1960–1967, Apr. 2022, doi: 10.1109/LRA.2022.3142905.

[13] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Madrid, Spain, Oct. 2018, pp. 4758–4765, doi: 10.1109/IROS.2018.8594299.

[14] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022, doi: 10.1109/TRO.2022.3141876.

[15] W. Xu and F. Zhang, "FAST-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021, doi: 10.1109/LRA.2021.3064227.

[16] J. Lin, C. Zheng, W. Xu, and F. Zhang, "R^2LIVE: A robust, real-time, LiDAR-inertial-visual tightly-coupled state estimator and mapping," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7469–7476, Oct. 2021, doi: 10.1109/LRA.2021.3095515.

[17] J. Lin and F. Zhang, "R^3LIVE: A robust, real-time, RGB-colored, LiDAR-inertial-visual tightly-coupled state estimation and mapping package," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Philadelphia, PA, USA, May 2022, pp. 10672–10678, doi: 10.1109/ICRA46639.2022.9811935.

[18] K. Eckenhoff, P. Geneva, and G. Huang, "MIMC-VINS: A versatile and resilient multi-IMU multi-camera visual-inertial navigation system," *IEEE Trans. Robot.*, vol. 37, no. 5, pp. 1360–1380, Oct. 2021, doi: 10.1109/TRO.2021.3049445.

[19] W. Zhang, S. Wang, X. Dong, R. Guo, and N. Haala, "BAMF-SLAM: Bundle adjusted multi-fisheye visual-inertial SLAM using recurrent field transforms," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, London, UK, May 2023, pp. 6232–6238, doi: 10.1109/ICRA48891.2023.10160905.

[20] Z. Wang, K. Yang, H. Shi, P. Li, F. Gao, and K. Wang, "LF-VIO: A visual-inertial-odometry framework for large field-of-view cameras with negative plane," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2022, pp. 4423–4430.

[21] Z. Wang, K. Yang, H. Shi, P. Li, F. Gao, J. Bai, and K. Wang, "LF-VISLAM: A SLAM framework for large field-of-view cameras with negative imaging plane on mobile agents," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 4, pp. 6321–6335, Oct. 2024.

[22] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Barcelona, Spain, Nov. 2011, pp. 2320–2327, doi: 10.1109/ICCV.2011.6126513.

[23] R. Wang, M. Schworer, and D. Cremers, "Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 3903–3911, doi: 10.1109/ICCV.2017.421.

[24] S. Urban and S. Hinz, "MultiCol-SLAM: A modular real-time multi-camera SLAM system," arXiv preprint, Oct. 2016. [Online]. Available: https://arxiv.org/abs/1610.07336

[25] T.-M. Nguyen, S. Yuan, M. Cao, Y. Lyu, T. H. Nguyen, and L. Xie, "MILIOM: Tightly coupled multi-input lidar-inertia odometry and mapping," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 5573–5580, Jul. 2021, doi: 10.1109/LRA.2021.3080633.

[26] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, UK: Cambridge University Press, 2003.

[27] C. Yuan, W. Xu, X. Liu, X. Hong, and F. Zhang, "Efficient and probabilistic adaptive voxel mapping for accurate online LiDAR odometry," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 8518–8525, Jul. 2022, doi: 10.1109/LRA.2022.3187250.

[28] L. Zhang, M. Helmberger, L. F. T. Fu, D. Wisth, M. Camurri, D. Scaramuzza, and M. Fallon, "Hilti Oxford dataset: A millimetre accurate benchmark for simultaneous localization and mapping," *IEEE Robot. Autom. Lett.*, vol. 8, no. 1, pp. 408–415, Jan. 2023, doi: 10.1109/LRA.2022.3226077.

[29] L. Zhang, M. Camurri, D. Wisth, and M. Fallon, "Multi-Camera LiDAR Inertial Extension to the Newer College Dataset," arXiv preprint, Dec. 2021. [Online]. Available: https://arxiv.org/abs/2112.08854