

FAST-LIEO2: Fast and Tightly-Coupled LiDAR-Inertial-Event Odometry

Jiang Wu , Zirui Wang , Jing Wu , I-Ming Chen , *Fellow, IEEE*, Zhenzhong Jia , *Member, IEEE*, and Boyu Zhou 

Abstract—In geometrically degenerate environments such as corridors and tunnels, LiDAR–Inertial odometry often drifts due to insufficient constraints, while conventional cameras can provide unreliable visual constraints under HDR lighting and severe motion blur. Event cameras offer microsecond-level temporal resolution and a high dynamic range, providing an additional information source for state estimation under challenging illumination and fast motion. In this work, we propose FAST-LIEO2 (and FAST-LIEVO2 with optional RGB fusion), a tightly-coupled LiDAR–inertial–event(–RGB) odometry framework that fuses LiDAR, event, and optional visual measurements via a sequentially updated error-state iterated Kalman filter (ESIKF) for efficient and stable multi-modal state estimation. To mitigate over-accumulation in highly active regions while preserving structure in low-activity areas, we design an activity-adaptive smoothed time surface (AS-TS) that adjusts the temporal decay parameter based on event activity. We further maintain two event submaps and design a coarse-to-fine hierarchical event update strategy, performing frame-to-frame geometric alignment followed by frame-to-map edge alignment, to improve prior pose correction and enlarge the convergence basin. Experiments on the public benchmark datasets and a self-collected dataset show that FAST-LIEO2 outperforms the compared multi-sensor SLAM baselines in accuracy and robustness across diverse structural and illumination conditions, while achieving strong real-time performance.

Index Terms—SLAM, sensor fusion, localization.

I. INTRODUCTION

RECENT advances in SLAM have significantly improved the ability of robots to localize and build maps

Received 23 March 2026; accepted 26 June 2026. Date of publication 20 July 2026; date of current version 3 August 2026. This article was recommended for publication by Associate Editor Y. Zhou and Editor J. Civera upon evaluation of the reviewers' comments. This work was supported in part by the National Key Research and Development Program of China under Grant 2024YFF0506704, in part by the National Natural Science Foundation of China under Grant 62503377, and in part by the Application Scenario Unveiling Support Action Project of Nanshan District in Shenzhen, under Project 20251128936000002. (Corresponding author: Jing Wu.)

Jiang Wu, Jing Wu, Zhenzhong Jia, and Boyu Zhou are with the Department of Mechanical and Energy Engineering, Southern University of Science and Technology (SUSTech), Shenzhen 518055, China (e-mail: wuj@sustech.edu.cn).

Zirui Wang is with the Department of Mechanical and Energy Engineering, Southern University of Science and Technology (SUSTech), Shenzhen 518055, China, and also with the Thrust of Robotics and Autonomous Systems, Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511453, China.

I-Ming Chen is with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798.

The source code of FAST-LIEO2 and our dataset are available at <https://github.com/Empty-Wu/FAST-LIEO2>.

This article has supplementary downloadable material available at <https://doi.org/10.1109/LRA.2026.3715038>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2026.3715038

in previously unknown environments [1]. However, existing single-sensor systems are often unable to provide accurate and robust pose estimation under challenging conditions. For example, LiDAR often lacks sufficient geometric constraints and tends to drift in degenerate environments such as corridors, tunnels, or single-wall scenes [2], while RGB cameras often lose effective visual constraints under high dynamic range (HDR) lighting and severe motion blur [3]. To overcome the limitations of single-sensor systems, multi-sensor fusion SLAM has attracted growing attention [4]. By combining complementary modalities, fusion systems can improve observability and mitigate temporary degradation of individual sensors. Most existing approaches use RGB cameras as the primary visual sensor and perform data association and optimization under the photometric constancy assumption [5]. However, this assumption often breaks down under complex illumination and fast motion.

In contrast, event cameras produce an asynchronous stream of events triggered by brightness changes [6]. With microsecond-level temporal resolution and a high dynamic range, they can provide reliable information under poor illumination, strong lighting contrast, and fast motion, offering strong potential for state estimation in challenging lighting and aggressive motion [7]. However, because event data are generated asynchronously, their noise characteristics and data association strategies differ from conventional vision. As a result, the key challenge in integrating event cameras into multi-sensor fusion SLAM is to develop a computationally efficient, optimization-friendly event representation together with reliable measurement models and update strategies that remain stable across different event activity levels.

To address these challenges, we propose FAST-LIEO2 (and FAST-LIEVO2 with optional RGB fusion), a fast and tightly-coupled LiDAR–Inertial–event (–RGB) odometry framework that exploits the complementary strengths of each sensor to improve robustness in challenging environments. Our main contributions are summarized as follows:

- 1) We propose an activity-adaptive smoothed time surface (AS-TS) for event measurement alignment, which uses count-based local activity to adjust the temporal decay parameter and balance edge stability and responsiveness.
- 2) Leveraging the stable edge representation provided by AS-TS, we maintain two event submaps and design a coarse-to-fine event update, combining a frame-to-frame alignment for early pose correction with a frame-to-map edge alignment for refinement.

- 3) We integrate IMU, LiDAR, event, and optional RGB measurements into a tightly coupled multi-sensor filtering framework, enabling unified fusion of measurements with different dimensions.
- 4) We evaluate the proposed method on public benchmark datasets and a self-collected dataset, demonstrating high accuracy and robustness across diverse scenarios.

The remainder of this paper is organized as follows. Section II reviews related work on LiDAR-visual-inertial SLAM and event-based SLAM. Section III presents the proposed framework in detail. Section IV reports experimental results on public benchmarks and our self-collected dataset. Section V concludes the paper.

II. RELATED WORK

A. Lidar-Visual-Inertial SLAM

LiDAR-visual-inertial SLAM/odometry has been studied to improve robustness in challenging environments, with existing methods mainly categorized as optimization-based or filtering-based. Optimization-based approaches such as LVI-SAM [4] formulate LiDAR-inertial and visual-inertial constraints in a factor graph and solve them with sliding-window and global optimization, often with degeneracy handling. VILENS [8] jointly optimizes LiDAR primitives (e.g., lines and planes), visual features, and IMU preintegration to maintain long-term consistency. Filtering-based methods perform high-rate updates in the error-state form. R2LIVE [9] fuses LiDAR, vision, and IMU using an iterated Kalman filter and optionally refines keyframes with low-rate graph optimization. FAST-LIVO [10] combines geometric residuals with sparse photometric residuals through LiDAR-image associations. R3LIVE [5] adopts a two-stage visual update with frame-to-frame tracking followed by frame-to-map photometric alignment, and R3LIVE++ [11] further improves robustness via photometric calibration and online exposure-time estimation. Building on these ideas, FAST-LIVO2 [12] improves efficiency using sequential measurement updates and a shared map structure. Although these methods effectively exploit complementary LiDAR, visual, and inertial constraints, their reliance on RGB images and photometric consistency makes them sensitive to illumination changes, motivating the use of event cameras in this work.

B. Event-Based SLAM

Event-based SLAM relies on robust event representations and multi-sensor fusion [7]. Existing methods can be broadly divided into direct alignment and feature-based constraint construction. Direct methods, such as ESVO [13], align event representations with 3D structure and are robust under fast motion and HDR scenes. However, its mapping becomes costly on long or high-resolution sequences, and the fixed temporal decay may cause spatially uneven edge representations. IMU-aided ESVO [14] improves convergence with inertial priors, and ESVO2 [15] introduces an adaptive accumulation (AA) event representation that, inspired by Adaptive Global Decay Process (AGDP) [16], monitors local event dynamics and terminates regional event

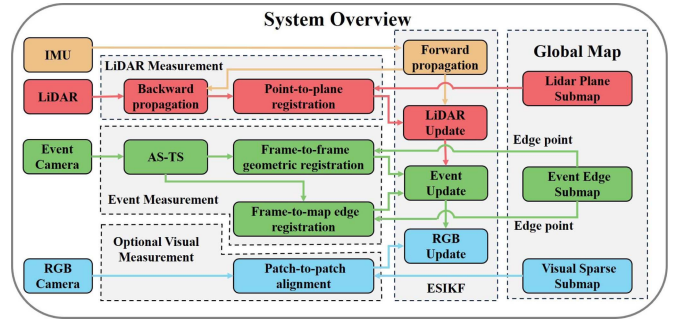


Fig. 1. System overview of FAST-LIEO2. FAST-LIEO2 is configurable as LIEO2 or LIEVO2 (with optional visual measurement).

accumulation for efficient contour-point sampling and stereo mapping. In contrast, feature-based methods, such as ES-VIO [17] and Ultimate SLAM [18], construct constraints from event features and usually rely on sliding-window or factor-graph optimization. Beyond camera-IMU fusion, FAST-LIEO [19] incorporates TS-based event observations into a filtering-based LiDAR-IMU-event pipeline, but its frame-to-map alignment is sensitive to prior pose errors.

Motivated by these observations, we propose AS-TS to obtain a stable activity-adaptive event representation for tightly coupled LiDAR-IMU-event odometry under spatially non-uniform event activity. Existing activity-adaptive event representations estimate and use event activity in different ways. AGDP [16] adapts temporal decay and event weights using global activity estimated from inter-event timestamp intervals, while the AA in ESVO2 [15] extends the event-activity-based adaptation idea in AGDP to regional event accumulation, where local event activity is used to terminate accumulation and generate event-count maps for contour-point sampling. Following this activity-adaptive idea, AS-TS estimates local activity from regional event counts within fixed filter-synchronized fusion intervals and maps it to cell-wise adaptive decay parameters, forming an edge-consistency field for cross-modal ESIKF residuals. Based on AS-TS, we further maintain event submaps and design a coarse-to-fine event update with frame-to-frame correction followed by frame-to-map alignment.

III. SYSTEM DESCRIPTION

Fig. 1 shows the architecture of the FAST-LIEO2 framework. After IMU propagation, LiDAR, event, and optional visual measurements are fused sequentially within the ESIKF in the order of LiDAR, event, and visual updates. To support modality-specific updates, all association submaps are selected from the global LiDAR map using modality-specific criteria. Since LiDAR, event, and visual measurements rely on different measurement models, data-association criteria, and map maintenance strategies, separate submaps are maintained for their respective measurement models. The LiDAR measurement uses a dense planar submap and updates the state using point-to-plane residuals. The event measurement uses AS-TS to represent event edges, maintains global and local event submaps for registration, and performs a coarse-to-fine update with frame-to-frame geometric

registration followed by frame-to-map edge registration. The optional visual measurement maintains a sparse submap and refines the state via sparse direct patch alignment.

In our system, the system state is defined as:

$$\mathbf{x} \triangleq \begin{bmatrix} {}^G\mathbf{R}_I^T & {}^G\mathbf{p}_I^T & {}^G\mathbf{v}_I^T & \mathbf{b}_g^T & \mathbf{b}_a^T & {}^G\mathbf{g}^T \end{bmatrix}^T \quad (1)$$

which contains the IMU orientation ${}^G\mathbf{R}_I^T$, position ${}^G\mathbf{p}_I^T$, and velocity ${}^G\mathbf{v}_I^T$ in the global frame, together with gyroscope $\mathbf{b}_g^T$ and accelerometer biases $\mathbf{b}_a^T$ and the gravity vector ${}^G\mathbf{g}^T$.

To apply perturbations and compute differences on this non-Euclidean state space, we adopt the $\boxplus/\boxminus$ operators following [20]. For $\mathcal{M} = SO(3) \times \mathbb{R}^n$, the operators used in this work are defined as follows:

$$(\mathbf{R}, \mathbf{a}) \boxplus (\mathbf{r}, \mathbf{b}) \triangleq (\mathbf{R} \text{Exp}(\mathbf{r}), \mathbf{a} + \mathbf{b}) \quad (2)$$

$$(\mathbf{R}_1, \mathbf{a}) \boxminus (\mathbf{R}_2, \mathbf{b}) \triangleq (\text{Log}(\mathbf{R}_2^T \mathbf{R}_1), \mathbf{a} - \mathbf{b}) \quad (3)$$

where $\mathbf{r} \in \mathbb{R}^3$ and $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$. $\text{Exp}(\cdot)$ and $\text{Log}(\cdot)$ denote the exponential and logarithm maps between $\mathfrak{so}(3)$ and $SO(3)$.

A. Sequentially Updated Error-State Iterated Kalman Filter

We assume that the LiDAR, IMU, event camera, and RGB camera are temporally synchronized and extrinsically calibrated offline. We first perform IMU forward propagation to obtain the state prior $p(\mathbf{x})$:

$$\mathbf{x} \boxminus \hat{\mathbf{x}} \sim \mathcal{N}(\mathbf{0}, \hat{\mathbf{P}}) \quad (4)$$

where $\hat{\mathbf{P}}$ is the propagated covariance. The measurement model of each modality is written as:

$$\begin{bmatrix} \mathbf{y}_l \\ \mathbf{y}_{eg} \\ \mathbf{y}_{ee} \\ \mathbf{y}_c \end{bmatrix} = \begin{bmatrix} \mathbf{h}_l(\mathbf{x}, \epsilon_l) \\ \mathbf{h}_{eg}(\mathbf{x}, \epsilon_{eg}) \\ \mathbf{h}_{ee}(\mathbf{x}, \epsilon_{ee}) \\ \mathbf{h}_c(\mathbf{x}, \epsilon_c) \end{bmatrix} \quad (5)$$

where $\mathbf{y}_l$, $\mathbf{y}_{eg}$, $\mathbf{y}_{ee}$, and $\mathbf{y}_c$ denote the LiDAR, frame-to-frame event, frame-to-map event, and optional RGB measurements, respectively, and ϵ_l , ϵ_{eg} , ϵ_{ee} , and ϵ_c are their measurement noises. Since the modalities provide conditionally independent measurements with different dimensions, we employ a sequential-update ESIKF to avoid dimension mismatch and allow greater flexibility in module design.

Specifically, we obtain the posterior over $\mathbf{x}$ via four sequential Bayesian updates. Starting from the IMU-propagated prior $p(\mathbf{x})$, we incorporate the LiDAR measurement $\mathbf{y}_l$, the frame-to-frame geometric measurement $\mathbf{y}_{eg}$, the frame-to-map edge measurement $\mathbf{y}_{ee}$, and the optional RGB measurement $\mathbf{y}_c$ in order, yielding $p(\mathbf{x} | \mathbf{y}_l, \mathbf{y}_{eg}, \mathbf{y}_{ee}, \mathbf{y}_c)$:

$$\begin{aligned} p(\mathbf{x} | \mathbf{y}_l, \mathbf{y}_{eg}, \mathbf{y}_{ee}, \mathbf{y}_c) &\propto p(\mathbf{y}_c | \mathbf{x}) p(\mathbf{x} | \mathbf{y}_l, \mathbf{y}_{eg}, \mathbf{y}_{ee}) \\ &\propto p(\mathbf{y}_c | \mathbf{x}) p(\mathbf{y}_{ee} | \mathbf{x}) p(\mathbf{y}_{eg} | \mathbf{x}) p(\mathbf{y}_l | \mathbf{x}) p(\mathbf{x}) \end{aligned} \quad (6)$$

Following the sequential-update ESIKF in [12], we denote the κ -th iterated estimate as $\hat{\mathbf{x}}^\kappa$ with $\hat{\mathbf{x}}^0 = \hat{\mathbf{x}}$. We linearize the nonlinear measurement function in (5) around $\hat{\mathbf{x}}^\kappa$ using a first-order Taylor expansion, yielding the following form (for LiDAR,

event, or RGB measurements):

$$\mathbf{y} | \mathbf{x} \simeq \underbrace{\mathbf{h}(\hat{\mathbf{x}}^\kappa, \mathbf{0})}_{\mathbf{r}^\kappa} + \mathbf{H}^\kappa \delta \mathbf{x}^\kappa + \mathbf{L}^\kappa \mathbf{v} \quad (7)$$

where $\delta \mathbf{x}^\kappa = \mathbf{x} \boxminus \hat{\mathbf{x}}^\kappa$, $\mathbf{r}^\kappa$ is the residual between the observation and the predicted measurement, and $\mathbf{L}^\kappa \mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ is the measurement noise. $\mathbf{H}^\kappa$ and $\mathbf{L}^\kappa$ are the Jacobians with respect to $\delta \mathbf{x}^\kappa$ and $\mathbf{v}$, respectively.

Combining the prior $p(\mathbf{x})$ with the above likelihood and applying the maximum a-posterior estimation yields the standard error-state iterated Kalman filter update, given by:

$$\mathbf{K} = \left((\mathbf{H}^\kappa)^T \mathbf{R}^{-1} \mathbf{H}^\kappa + \hat{\mathbf{P}}^{-1} \right)^{-1} (\mathbf{H}^\kappa)^T \mathbf{R}^{-1} \quad (8)$$

$$\hat{\mathbf{x}}^{\kappa+1} = \hat{\mathbf{x}}^\kappa \boxminus (-\mathbf{K} \mathbf{r}^\kappa - (\mathbf{I} - \mathbf{K} \mathbf{H}^\kappa)(\hat{\mathbf{x}}^\kappa \boxminus \hat{\mathbf{x}})) \quad (9)$$

After convergence, $\hat{\mathbf{x}}^{\kappa+1}$ and its covariance give the posterior mean and covariance of $p(\mathbf{x} | \mathbf{y})$.

B. LiDAR Measurement

To compensate for motion distortion during a LiDAR scan, we apply backward propagation to compensate each point to the scan end time [21]. The undistorted LiDAR point ${}^L\mathbf{p}_i^m$ is then transformed to the global frame using the state estimate at the k -th LiDAR iterative update, denoted by $\hat{\mathbf{x}}^\kappa$:

$${}^G\hat{\mathbf{p}}_i^k = {}^G\hat{\mathbf{T}}_I^k {}^I\mathbf{T}_L {}^L\mathbf{p}_i^m \quad (10)$$

For LiDAR measurement modeling, we maintain a dense voxel map, where each planar voxel stores the plane center $\mathbf{q}_i$, normal $\mathbf{n}_i$, and their covariance $\Sigma_{\mathbf{n}, \mathbf{q}}$. For each point in the global frame, we query its corresponding root or child voxel in the dense voxel map. When no corresponding voxel is found, or when the retrieved voxel does not represent a plane, the point is discarded. The measurement model $\mathbf{h}_l(\mathbf{x}, \epsilon_l)$ for ${}^L\mathbf{p}_i^m$ is then defined by the point-to-plane consistency:

$$0 = (\mathbf{n}_i \boxminus \delta \mathbf{n}_i)^T ({}^G\mathbf{T}_I^k {}^I\mathbf{T}_L ({}^L\mathbf{p}_i^m - \delta {}^L\mathbf{p}_i^m) - (\mathbf{q}_i - \delta \mathbf{q}_i)) \quad (11)$$

where the measurement uncertainty ϵ_l arises from the LiDAR point measurement noise $\delta {}^L\mathbf{p}_i^m$ and the uncertainty in the estimated plane parameters, including the plane normal noise $\delta \mathbf{n}_i$ and the plane center noise $\delta \mathbf{q}_i$.

C. Event Measurement

1) *Event Representation*: Most event-camera SLAM systems use the time surface (TS) [22], where each pixel stores an exponentially decayed value with a fixed temporal decay parameter. However, using a constant temporal decay parameter often leads to spatial imbalance in real scenes: regions with high event rates tend to produce overly thick edges and redundant accumulation, while low-activity regions may suffer from broken edges and insufficient response. To address this issue, we construct an activity-adaptive smoothed time surface (AS-TS), which estimates local activity from event counts within the current filter-synchronized fusion interval and maps the activity to cell-wise adaptive decay parameters. The resulting AS-TS is then used as an edge-consistency field for cross-modal ESIKF measurement updates.

Specifically, the image plane is partitioned into square cells of size $s \times s$, indexed by j . At the k -th event update, the number of events in cell j within the counting interval Δt is denoted as N_k^j , and the corresponding local event rate is $r_k^j = N_k^j / \Delta t$. We smooth the activity estimate by an exponential moving average (EMA):

$$\lambda_k^j = (1 - \alpha)\lambda_{k-1}^j + \alpha r_k^j \quad (12)$$

where k indexes the event update, and α is the EMA coefficient. Based on λ_k^j , we define an adaptive temporal decay parameter τ_k^j for each cell, which is set inversely proportional to the activity and clamped to a bounded range:

$$\tau_k^j = \text{clip}\left(\frac{\gamma}{\lambda_k^j + \varepsilon}, \tau_{\min}, \tau_{\max}\right) \quad (13)$$

where $\tau_{\min}$ and $\tau_{\max}$ are the lower and upper bounds of the decay parameter, respectively. The scalar γ controls the overall sensitivity of the mapping from the local event activity λ_k^j to the temporal decay parameter τ_k^j . The constant ε is a small positive stabilizer (with the same unit as λ_k^j) to avoid numerical issues when λ_k^j is close to zero. In this paper, the default AS-TS configuration is set to $s = 16$ pixels, $\alpha = 0.1$, $\gamma = 200$, $\varepsilon = 1$, $\tau_{\min} = 20$ ms, and $\tau_{\max} = 50$ ms. The default AS-TS parameters are selected as a robust operating point based on the functional role of each parameter and preliminary empirical checks, rather than being tuned for a specific experimental setting. The same parameters are used in all experiments except for the parameter sensitivity analysis.

At the reference time t_k , let $t_{\text{last}}(\mathbf{u})$ denote the timestamp of the most recent event at pixel $\mathbf{u}$. The AS-TS is defined as:

$$\mathcal{S}_k(\mathbf{u}, t_k) = \exp\left(-\frac{t_k - t_{\text{last}}(\mathbf{u})}{\tau_k^j}\right) \quad (14)$$

For subsequent image processing, we construct a polarity-free negative AS-TS and remap its pixel values to $[0, 255]$. The polarity-free negative AS-TS is defined as:

$$\mathcal{S}_k^n(\mathbf{u}, t_k) \triangleq 255.0 (1 - \mathcal{S}_k(\mathbf{u}, t_k)) \quad (15)$$

Finally, $\mathcal{S}_k^n$ is Gaussian-smoothed to improve convergence in iterative state updates, and the AS-TS generation procedure is summarized in Algorithm 1.

2) *Event Submap Maintenance*: We maintain a global semi-dense event submap and a local event submap for event alignment, both constructed as event-selected subsets of the global LiDAR map. Instead of embedding event-association states into the dense LiDAR map, we maintain the event submaps separately to decouple event-specific map management from the LiDAR point-to-plane map. The event submaps retain only LiDAR points associated with recently activated AS-TS pixels, excluding event-irrelevant geometric points from event alignment. After each state update at time t_k , the latest LiDAR points are projected onto the negative AS-TS image $\mathcal{S}_k^n$, and points on recently activated pixels are inserted into the global event submap, with only the closest point retained for each pixel.

At each event update, global event-submap points are projected onto the current negative AS-TS image using the current state estimate, and the points within the event camera field of view are further verified by the latest LiDAR scan. Only points

Algorithm 1: AS-TS Generation.

Input: Event stream in interval Δt , previous activity λ_{k-1}^j .

Output: Negative AS-TS map $\mathcal{S}_k^n$.

Partition image plane into cells j of size $s \times s$;

for each cell j do

 Count events N_k^j within Δt , and compute local rate

$r_k^j = N_k^j / \Delta t$;

 Update smoothed activity: $\lambda_k^j = (1 - \alpha)\lambda_{k-1}^j + \alpha r_k^j$;

 Compute adaptive decay:

$\tau_k^j = \text{clip}\left(\frac{\gamma}{\lambda_k^j + \varepsilon}, \tau_{\min}, \tau_{\max}\right)$;

for each pixel $\mathbf{u} \in \text{cell } j$ do

$\mathcal{S}_k(\mathbf{u}, t_k) = \exp\left(-\frac{t_k - t_{\text{last}}(\mathbf{u})}{\tau_k^j}\right)$;

$\mathcal{S}_k^n(\mathbf{u}, t_k) \triangleq 255.0 \cdot (1 - \mathcal{S}_k(\mathbf{u}, t_k))$;

Apply Gaussian smoothing to $\mathcal{S}_k^n$;

return $\mathcal{S}_k^n$

with valid and consistent LiDAR depth are retained to form the local event submap. To mitigate error accumulation in the global event submap, we perform consistency-based pruning after each update using the current local event submap. The local event submap points are reprojected onto the negative AS-TS with the updated pose. Since low pixel values indicate the most recent edge locations, points falling on high-value pixels are treated as noisy or dynamic outliers, and their corresponding points in the global event submap are removed.

3) *Frame-to-Frame Geometric Measurement Model*: FAST-LIEO [19] performs event updates via direct frame-to-map alignment. As a result, the update depends strongly on the pose prior. When the initial error is large, projected map edges can be far from the correctly corresponding event edges, and the local linearization becomes invalid, which limits the effective convergence range. To reduce sensitivity to large initial errors while maintaining real-time performance, we design a coarse-to-fine two-stage event update that is performed at every event update. The first stage uses frame-to-frame geometric constraints between consecutive AS-TS frames to provide a coarse correction and establish reliable correspondences. The second stage performs frame-to-map edge alignment on the event submap to refine the estimate.

Assume that at time t_{k-1} , we have a set of successfully tracked map points $\mathcal{P} = \{^G\mathbf{p}_1^e, ^G\mathbf{p}_2^e, \dots, ^G\mathbf{p}_i^e\}$. We project them onto the previous negative AS-TS image to obtain the corresponding pixel coordinates $\mathbf{u}_{1k-1}, \mathbf{u}_{2k-1}, \dots, \mathbf{u}_{ik-1}$. Then, on two consecutive negative AS-TS images $\mathcal{S}_{k-1}^n$ and $\mathcal{S}_k^n$, we track these pixels using Lucas-Kanade optical flow to estimate their locations in the current frame, denoted by $\mathbf{u}_{1k}^f, \mathbf{u}_{2k}^f, \dots, \mathbf{u}_{ik}^f$. Based on these correspondences and considering the measurement uncertainty, we construct a frame-to-frame geometric measurement model $\mathbf{h}_{eg}(\mathbf{x}, \epsilon_{eg})$ using the geometric reprojection error of the tracked points:

$$0 = \left(\mathbf{u}_{ik}^f - \delta \mathbf{u}_{ik}^f\right) - \left(\pi_e \left(^E\mathbf{T}_I \ ^G\mathbf{T}_{I_k}^{-1} \ ^G\mathbf{p}_i^e\right) - \delta \mathbf{u}_{ik}^e\right) \quad (16)$$

where $^G\mathbf{T}_{I_k}$ denotes the coordinate transformation from the IMU frame I_k to the global frame G at time t_k (its inverse

transforms a point from G to I_k) and ${}^E\mathbf{T}_I$ is the fixed IMU–event camera extrinsic. The projection operator $\pi_e(\cdot)$ projects a 3D point expressed in the event camera frame E onto the AS-TS image plane. The measurement noise ϵ_{eg} arises from the optical-flow tracking noise $\delta\mathbf{u}_{i_k}^f$ and the projection uncertainty of the camera model $\delta\mathbf{u}_{i_k}$.

4) *Frame-to-Map Edge Measurement Model*: After the frame-to-frame update, each event update performs frame-to-map edge alignment using the local event submap. When constructing the frame-to-map event measurement model, we avoid explicit feature extraction and matching by treating the AS-TS image as a differentiable edge-consistency field. Since newly triggered event edges appear as stable low-value bands in the image, the 3D edge points ${}^G\mathbf{p}_i^e$ maintained in the event submap should align with these bands after projection onto the image plane under the current pose estimate. For each valid map point ${}^G\mathbf{p}_i^e$, the frame-to-map edge measurement $\mathbf{h}_{ee}(\mathbf{x}, \epsilon_{ee})$ is defined as follows, with ϵ_{ee} arising from the AS-TS image uncertainty δS_k^n :

$$0 = S_k^n(\pi_e({}^E\mathbf{T}_I \cdot {}^G\mathbf{T}_{I_k}^{-1} \cdot {}^G\mathbf{p}_i^e), t_k) - \delta S_k^n \quad (17)$$

D. Visual Measurement

The visual measurement follows the sparse direct photometric alignment in FAST-LIVO [10]. A sparse visual submap is maintained by dividing the image into grids and selecting representative projected LiDAR points with high image gradients, each associated with a reference patch in the image pyramid. For each visible map point, the closest-viewpoint patch $\mathbf{Q}_n$ is selected, and the RGB measurement model $\mathbf{h}_c(\mathbf{x}, \epsilon_c)$ minimizes the photometric residual between the reference patch and the current image:

$$0 = (I_k(\pi_c({}^I\mathbf{T}_C^{-1} {}^G\mathbf{T}_{I_k}^{-1} {}^G\mathbf{p}_i^v)) - \delta I_k) - \mathbf{A}_n(\mathbf{Q}_n - \delta\mathbf{Q}_n) \quad (18)$$

where ${}^G\mathbf{p}_i^v$, ${}^I\mathbf{T}_C$, $\pi_c(\cdot)$, $I_k(\cdot)$, and $\mathbf{A}_n$ denote the visual submap point, camera–IMU extrinsic, camera projection model, current image pyramid, and affine warp from $\mathbf{Q}_n$ to the image pyramid, respectively. The noise ϵ_c comes from the current image uncertainty δI_k and patch uncertainty $\delta\mathbf{Q}_n$.

IV. EXPERIMENTS AND RESULTS

We evaluate FAST-LIEO2 against state-of-the-art multi-sensor SLAM systems on public benchmarks and a self-collected dataset. We further use the self-collected dataset to assess performance in challenging scenarios. All experiments are conducted on an Ubuntu 20.04 desktop equipped with an AMD Ryzen 7 9700X CPU and 32 GB RAM.

A. Benchmark Experiments

1) *Experiment Setup*: We evaluate the proposed system on ECMD [23] and VECtor [24]. ECMD experiments use *Tunnel_e_a/b/c*, *Urban_day_e_b*, and *Urban_night_e_a* with the left DVXplorer event camera (640×480), left RGB camera (1920×1200), Velodyne HDL-32E LiDAR, and Xsens-MTi-30 IMU, while VECtor experiments use *Units_dolly1*,

School_dolly1, and *Corridors_dolly1* with the left Prophesee Gen3 CD event camera, left FLIR Grasshopper3 camera, Ouster OS0-128 LiDAR, and Xsens MTi-30 AHRS IMU. We compare against FAST-LIEO [19], FAST-LIVO2 [12], FAST-LIO2 [21], R3LIVE [5], ESVIO [17], ESVO2 [15], and Ultimate SLAM [18], with a TS-based ablation experiment. FAST-LIEO is selected as the direct baseline instead of FAST-LIEVO because it shares the same LiDAR–IMU–event configuration as FAST-LIEO2, enabling a controlled evaluation of the proposed event-related improvements. Since FAST-LIEVO and FAST-LIEVO2 adopt similar optional sparse direct visual updates, the influence of visual fusion is analyzed through the FAST-LIEO2/FAST-LIEVO2 comparison, and FAST-LIEVO is not included as an additional baseline. All methods are evaluated using EVO [25] against the reference trajectories provided by each dataset. ATE(RMSE) and ARE(RMSE) are reported in Table I. FAST-LIEO2/FAST-LIEVO2 achieve the best ATE on most sequences and competitive ARE, while maintaining stable operation across both ECMD and VECtor.

For ECMD, although PTP/GNSS synchronization and camera–IMU temporal-offset calibration are reported, residual temporal inconsistency and non-uniform visual sampling are observed in some released streams, which can destabilize filtering-based multi-sensor fusion. Fixed offline time-offset compensation is therefore applied as an empirical correction, initialized from released timestamp relationships and refined according to cross-modal consistency and filter stability. The RGB–IMU offsets are 0.14 s, 0.13 s, and 0.12 s for *Tunnel_e_a/b/c*, and 0 s for *Urban_day_e_b* and *Urban_night_e_a*. The DVXplorer (left)–IMU offset is set to 1.5 s for all ECMD sequences and is treated as an empirical stream-alignment offset, rather than physical event-camera latency or an official hardware calibration result. Due to more pronounced RGB sampling irregularity and residual timestamp inconsistency on *Urban_day_e_b* and *Urban_night_e_a*, FAST-LIVO2 becomes unstable in LIVO mode and is therefore evaluated in LIO mode on these two sequences. For VECtor, since reliable synchronization configurations are provided, all experiments follow the released dataset configuration.

2) *Evaluation Results*: The ECMD sequences start with a high initial speed of about 20 m/s, making initialization and early-stage estimation challenging. As shown in Fig. 2, FAST-LIEO2 improves over FAST-LIEO by using AS-TS and the coarse-to-fine event update to provide stable edge constraints and early pose correction. FAST-LIO2 and FAST-LIVO2 show larger short-term drift, while R3LIVE, ESVIO, ESVO2, and Ultimate SLAM are prone to failure once initialization is inaccurate. On VECtor, FAST-LIEO2 remains stable on all three indoor sequences, whereas ESVIO shows larger drift than LiDAR-based baselines, ESVO2 drifts severely, and Ultimate SLAM fails throughout, mainly due to low-texture or weak-edge regions that degrade event–visual association or direct event alignment. The comparison between FAST-LIEO2 and FAST-LIEVO2 further shows that visual measurements are conditionally beneficial. On ECMD, high-speed motion, coarse synchronization, and non-uniform visual sampling can degrade sparse direct visual residuals, whereas on VECtor, with smoother motion,

TABLE I
RMSE ON ECMD AND VECTOR BENCHMARK SEQUENCES. EACH ENTRY REPORTS ATE(RMSE, M) / ARE(RMSE, DEG)

Dataset	Sequence	FAST-LIEO2(ours)	FAST-LIEO2(TS)	FAST-LIEVO2(ours)	FAST-LIEO	FAST-LIVO2	FAST-LIO2	R3LIVE	ESVIO	ESVO2	Ultimate SLAM
ECMD	Tunnel_e_a	0.806 / 3.81	1.009 / 3.76	0.721 / 5.23	2.249 / 4.57	4.850 / 5.72	5.282 / 5.02	fail	fail	fail	fail
	Tunnel_e_b	0.476 / 3.45	0.825 / 3.57	1.270 / 3.67	23.065 / 4.67	12.366 / 5.43	fail	fail	fail	fail	fail
	Tunnel_e_c	1.177 / 6.45	1.208 / 6.73	1.389 / 8.51	2.839 / 9.85	27.500 / 10.78	47.700 / 9.56	fail	fail	fail	fail
	Urban_day_e_b	2.920 / 3.72	3.687 / 3.77	3.097 / 5.63	3.933 / 5.08	3.577 / 5.68	5.197 / 6.65	fail	fail	fail	fail
	Urban_night_e_a	0.726 / 4.05	0.749 / 4.06	0.857 / 4.35	1.234 / 4.67	1.747 / 4.05	1.835 / 6.32	fail	fail	fail	fail
VECToR	Units_dolly1	0.200 / 1.66	0.202 / 1.64	0.148 / 1.61	0.207 / 1.34	0.249 / 1.86	0.305 / 2.20	0.179 / 1.25	1.515 / 4.57	14.174 / 121.20	fail
	School_dolly1	0.170 / 1.65	0.172 / 1.65	0.167 / 1.64	0.269 / 1.94	0.189 / 1.70	0.180 / 1.20	0.203 / 1.40	0.341 / 2.36	20.776 / 134.85	fail
	Corridors_dolly1	0.165 / 1.85	0.150 / 2.10	0.143 / 1.81	0.169 / 1.97	0.171 / 2.89	0.196 / 3.38	0.149 / 1.85	0.958 / 3.13	8.816 / 109.46	fail

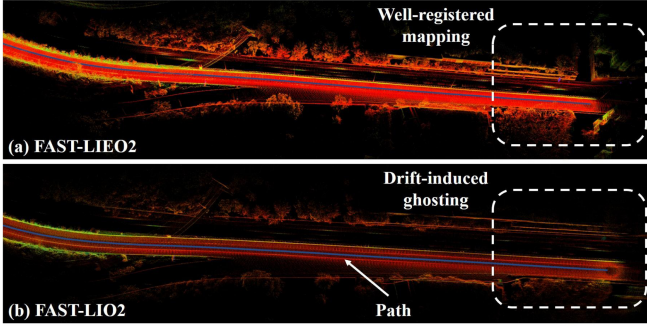


Fig. 2. Mapping results of FAST-LIEO2 and FAST-LIO2 on the ECMD Urban_day_e_b sequence. The blue lines are the estimated path.

TABLE II
PARAMETER SENSITIVITY OF AS-TS ON VECTOR. EACH ENTRY REPORTS ATE(RMSE, M) / ARE(RMSE, DEG).

Parameter	Setting	Units	School	Corridors	Avg
Default	Default	0.200/1.66	0.170/1.65	0.165/1.85	0.178/1.72
$[\tau_{\min}, \tau_{\max}]$	[5, 20]	0.185/1.64	0.470/1.79	0.165/1.97	0.273/1.80
	[20, 80]	0.197/1.64	0.173/1.58	0.161/2.48	0.177/1.90
	[50, 100]	4.639/3.35	0.166/1.99	1.147/3.46	1.984/2.93
γ	100	0.162/1.65	0.182/1.62	0.148/1.79	0.164/1.68
	300	0.202/1.64	0.173/1.57	0.154/2.11	0.176/1.78
ε	0.1	0.202/1.70	0.173/1.57	0.170/1.73	0.182/1.67
	10	0.201/1.63	0.176/1.58	0.133/1.89	0.170/1.70
s	8	0.194/1.61	0.175/1.58	0.162/1.78	0.177/1.66
	24	0.200/1.65	0.173/1.58	0.147/1.89	0.173/1.71
α	0.05	0.238/2.49	0.177/1.58	0.145/1.82	0.187/1.96
	0.5	0.203/1.64	0.175/1.57	0.143/1.92	0.174/1.71

better synchronization, and regular illumination, visual measurements provide effective complementary constraints. Thus, FAST-LIEVO2 can improve accuracy under reliable temporal and photometric conditions.

3) *Parameter Sensitivity Analysis*: To analyze AS-TS parameter sensitivity, we conduct a one-factor-at-a-time study on three VECToR sequences. As shown in Table II, the default setting is not always optimal for each sequence, but consistently achieves near-best accuracy. The temporal decay range $[\tau_{\min}, \tau_{\max}]$ is most influential: too short a temporal decay range may lead to broken edges and insufficient responses, as on *School_dolly1* under [5, 20], whereas too long a temporal decay range preserves stale events and thickens edges, causing degradation on *Units_dolly1* and *Corridors_dolly1* under [50, 100], where corridor-like or wall-facing segments make LiDAR constraints weak and event measurements more critical. In contrast, γ and ε are less sensitive due to the clipping of τ_k^j , while s and α only slightly affect accuracy through local activity estimation.

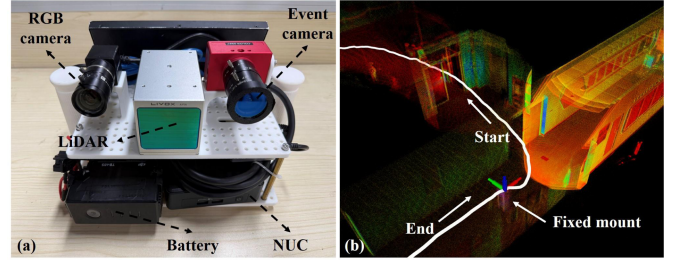


Fig. 3. Self-collected dataset setup: (a) handheld platform; (b) loop sequence with coincident start/end poses.

Overall, AS-TS is stable under reasonable hyperparameter changes, while an inappropriate temporal decay range can degrade performance when event constraints are important.

B. Self-Collected Dataset Experiments

1) *Self-Collected Dataset*: We collect data in a campus-scale environment. The dataset contains nine sequences, including three indoor, four outdoor, and two indoor-outdoor-indoor(ioi) transition sequences. Sequences with prefixes *ug*, *coe*, *cob*, *trb*, and *cn2* were collected in the underground parking garage, the College of Engineering, the College of Business, the Third Research Building and the area near the Second Canteen, respectively. These sequences are used to evaluate robustness under varying structural and illumination conditions. As shown in Fig. 3(a), the data collection platform is equipped with a DAVIS346 event camera (346×260), an HKVision MV-CA013-A0UC RGB camera (1280×1024), and a Livox Avia LiDAR. As shown in Fig. 3(b), for sequences labeled with “loop”, we use a fixed mount to enforce consistent start and end poses, and quantify accumulated drift using end-to-end error.

2) *Evaluation Results*: Since none of the evaluated systems includes loop closure, we evaluate the end-to-end translation and rotation errors on our self-collected dataset, computed from the relative pose between the start and final frames. We compare FAST-LIEO2 with FAST-LIEO [19], FAST-LIVO2 [12], FAST-LIO2 [21], R3LIVE [5], and Ultimate SLAM [18], and include an ablation experiment that replaces AS-TS with standard TS. As shown in Table III, FAST-LIEO2 converges reliably on all sequences and shows the strongest overall performance, achieving the lowest translation error on six sequences and the lowest rotation error on five sequences. On the remaining sequences, it still attains competitive results close to the best-performing methods, demonstrating robustness under diverse structural and illumination conditions.

TABLE III
END-TO-END POSE ERROR ON EACH SELF-COLLECTED SEQUENCE. EACH ENTRY REPORTS TRANSLATION ERROR (M) / ROTATION ERROR (DEG)

Sequence	FAST-LIEO2(ours)	FAST-LIEO2(TS)	FAST-LIEVO2(ours)	FAST-LIEO	FAST-LIVO2	FAST-LIO2	R3LIVE	Ultimate SLAM
cob-2f-loop	0.004 / 0.89	5.530 / 9.70	2.274 / 1.28	0.006 / 1.11	0.029 / 1.45	0.131 / 1.11	2.208 / 3.32	36.734 / 123.87
ug-loop	0.003 / 0.06	0.004 / 0.06	0.003 / 0.04	0.108 / 0.40	0.052 / 0.11	0.219 / 0.40	0.166 / 0.41	16.389 / 115.05
coe-5f-loop	0.017 / 2.34	0.334 / 2.30	0.485 / 2.95	0.059 / 2.03	1.557 / 2.53	6.463 / 7.98	0.044 / 2.14	19.683 / 168.76
cob-ioi-loop	0.010 / 2.48	1.317 / 5.42	0.772 / 2.57	6.914 / 6.94	1.066 / 3.26	4.363 / 7.71	22.169 / 13.35	40.344 / 71.05
coe-ioi-loop	0.019 / 0.25	1.584 / 7.43	0.852 / 0.41	1.550 / 2.85	0.008 / 0.43	5.995 / 5.51	16.800 / 26.25	11.587 / 41.45
cn2-loop	0.038 / 0.83	3.183 / 1.52	0.030 / 1.05	26.152 / 9.86	1.387 / 2.11	3.612 / 0.91	7.858 / 11.69	fail
coe-loop	0.033 / 1.45	1.587 / 10.59	0.041 / 1.77	3.376 / 1.70	1.570 / 2.38	5.906 / 1.02	3.188 / 1.19	22.717 / 120.90
trb-loop	0.011 / 0.38	2.832 / 1.92	0.006 / 0.25	1.269 / 0.30	0.852 / 1.40	1.935 / 1.03	33.861 / 14.82	90.579 / 25.43
trb-deg-loop	0.007 / 0.24	0.007 / 0.25	0.016 / 0.25	1.396 / 2.13	2.761 / 1.33	18.842 / 8.16	11.125 / 33.47	19.893 / 111.65

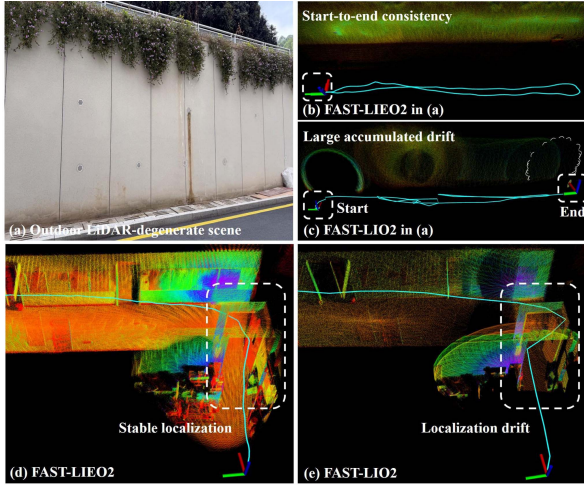


Fig. 4. Results in LiDAR-degenerate scenes: (a) outdoor degenerate scene; (b) and (c) compare FAST-LIEO2 and FAST-LIO2 in (a); (d) and (e) compare FAST-LIEO2 and FAST-LIO2 in an indoor degenerate scene.

For FAST-LIEVO2, the visual update is conditionally beneficial. It performs close to or better than FAST-LIEO2 on several sequences such as *ug-loop*, *cn2-loop*, *coe-loop*, *trb-deg-loop* and *trb-loop*, where the visual observations remain relatively reliable. However, in sequences with strong illumination changes, including *coe-5f-loop*, *cob-ioi-loop*, *coe-ioi-loop* and the semi-open *cob-2f-loop* with sunlit glass-wall corridors and dark enclosed corridors, exposure variation and photometric inconsistency can degrade sparse direct visual residuals. In such cases, FAST-LIEO2 avoids unreliable visual updates and achieves more stable odometry.

3) *AS-TS Ablation Experiment*: FAST-LIEO2(TS) in Tables I and III replaces AS-TS with the standard TS using a fixed 30 ms decay as used in FAST-LIEO [19]. On the public ECMD benchmark, AS-TS achieves lower ATE than the fixed-decay TS baseline on all five sequences, e.g., reducing the ATE from 0.825 m to 0.476 m on Tunnel_e_b and from 3.687 m to 2.920 m on Urban_day_e_b. On VECtor, the difference between AS-TS and TS is small, which is mainly attributed to the more regular motion and reliable synchronization of the VECtor sequences. On the self-collected dataset, AS-TS brings more evident gains under non-uniform event activity. For example, replacing AS-TS with TS increases the translation error from 0.017 m to 0.334 m on *coe-5f-loop*, from 0.038 m to 3.183 m on *cn2-loop*, from 0.033 m to 1.587 m on *coe-loop*, and from 0.004 m to 5.530 m on *cob-2f-loop*. These results show

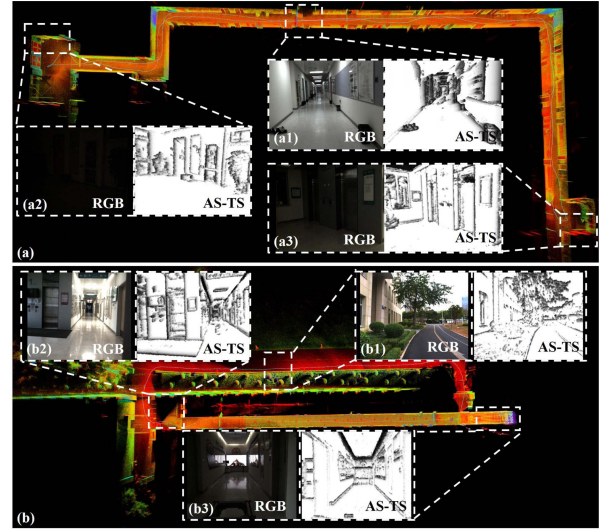


Fig. 5. FAST-LIEO2 mapping under challenging illumination: (a) coe-5f-loop, (b) coe-ioi-loop; (a1)–(a3) and (b1)–(b3) show lighting conditions.

that a fixed temporal decay can lead to spatially imbalanced event representations, weakening the alignment constraints.

4) *Challenging Scenarios*: We further evaluate FAST-LIEO2 in LiDAR-degenerate, HDR, and large-scale outdoor scenarios. In the LiDAR-degenerate *trb-deg-loop* sequence, FAST-LIEO2 reduces the end-to-end translation error from 18.842 m of FAST-LIO2 to 0.007 m, as shown in Table III and Fig. 4(b) and (c). Fig. 4(d) and (e) show an indoor LiDAR-degenerate scene, where FAST-LIO2 drifts significantly when facing a corridor wall, while FAST-LIEO2 remains stable by leveraging event measurements. Under challenging illumination, FAST-LIEO2 converges reliably on the HDR sequences, including *cob-2f-loop*, *coe-5f-loop*, *cob-ioi-loop*, and *coe-ioi-loop*. As shown in Table III and Fig. 5, it maintains low drift despite strong exposure changes, while several baselines suffer meter-level errors. This demonstrates the benefit of event measurements under high dynamic range conditions. For large-scale outdoor experiments, FAST-LIEO2 is evaluated on *coe-loop*, *cn2-loop*, and *trb-loop*, whose total trajectory length exceeds 2 km. Table III shows that FAST-LIEO2 converges reliably on all three sequences and achieves lower errors compared with the baselines. Fig. 6 further visualizes the FAST-LIEVO2 colored point clouds, demonstrating long-range consistency and mapping capability in large-scale scenarios.

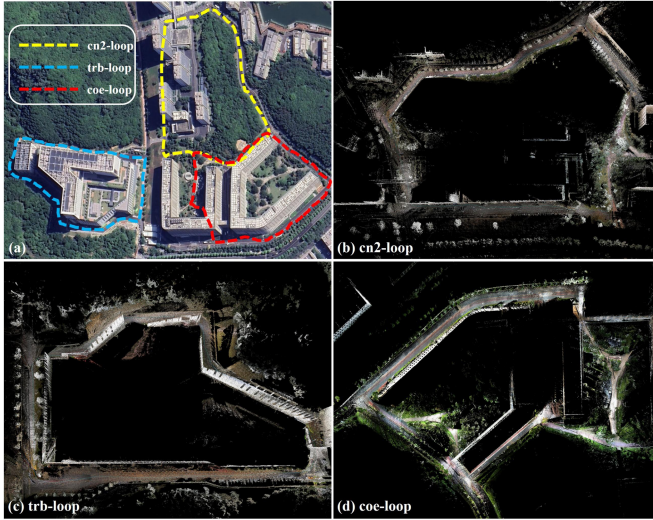


Fig. 6. Large-scale experiments: (a) satellite-view trajectories of cn2-loop, trb-loop, and coe-loop; (b)–(d) FAST-LIEVO2 colored point clouds.

TABLE IV
AVERAGE RUNTIME PER UPDATE (MEAN (STD), MS)

Event update		LiDAR update	Visual update
Event submap construction	0.74 (0.45)		
Event geometric update	0.38 (0.44)		
Event edge update	0.08 (0.02)		
Global map integration	0.27 (0.13)		
Outlier removal	0.02 (0.02)		
Total time	1.50 (1.02)	7.99 (6.99)	23.59 (12.99)

5) *Real-Time Ability Analysis*: Table IV reports the average per-update runtime over the nine sequences in Table III, where the values in parentheses denote the standard deviation across sequences. The event, LiDAR, and visual updates take 1.50 ms, 7.99 ms, and 23.59 ms on average, respectively, confirming real-time performance. Within the event update, most computation is spent on local map construction (0.74 ms) and event geometric update (0.38 ms), while the remaining components add little overhead. The larger standard deviation of LiDAR and visual updates reflects their sensitivity to scene structure and the complexity of motion and observations.

V. CONCLUSION

We present FAST-LIEO2/FAST-LIEVO2, a tightly coupled LiDAR-IMU-event(–RGB) odometry framework. AS-TS mitigates the spatial imbalance of fixed-decay TS by adapting the temporal decay parameter according to local event activity, and the global and local event submaps support coarse-to-fine event updates for frame-to-frame prior correction and frame-to-map refinement. LiDAR, event, and optional visual measurements are sequentially fused in an ESIKF. Experiments on public and self-collected datasets demonstrate robustness across structural degradation, HDR illumination, and large-scale outdoor environments. The system may degrade under weak event constraints and simultaneous multi-modal degradation. Future work will study degeneracy detection and adaptive weighting.

REFERENCES

- [1] C. Cadena et al., “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.
- [2] J. Hatleskog and K. Alexis, “Probabilistic degeneracy detection for point-to-plane error minimization,” *IEEE Robot. Automat. Lett.*, vol. 9, no. 12, pp. 11234–11241, Dec. 2024.
- [3] Z. Zhang, C. Forster, and D. Scaramuzza, “Active exposure control for robust visual odometry in HDR environments,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 3894–3901.
- [4] 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.*, 2021, pp. 5692–5698.
- [5] 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. Autom.*, 2022, pp. 10672–10678.
- [6] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, “A 240 × 180 130 dB 3 μ s latency global shutter spatiotemporal vision sensor,” *IEEE J. Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, Oct. 2014.
- [7] G. Gallego et al., “Event-based vision: A survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 154–180, Jan. 2022.
- [8] D. Wisth, M. Camurri, and M. Fallon, “VILENS: Visual, inertial, LiDAR, and leg odometry for all-terrain legged robots,” *IEEE Trans. Robot.*, vol. 39, no. 1, pp. 309–326, Feb. 2023.
- [9] J. Lin, C. Zheng, W. Xu, and F. Zhang, “R²LIVE: 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.
- [10] 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. Robots Syst.*, 2022, pp. 4003–4009.
- [11] J. Lin and F. Zhang, “R³LIVE++: A robust, real-time, radiance reconstruction package with a tightly-coupled LiDAR-inertial-visual state estimator,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 11168–11185, Dec. 2024.
- [12] C. Zheng et al., “FAST-LIVO2: Fast, direct LiDAR-inertial-visual odometry,” *IEEE Trans. Robot.*, vol. 41, pp. 326–346, 2025.
- [13] Y. Zhou, G. Gallego, and S. Shen, “Event-based stereo visual odometry,” *IEEE Trans. Robot.*, vol. 37, no. 5, pp. 1433–1450, Oct. 2021.
- [14] J. Niu, S. Zhong, and Y. Zhou, “IMU-aided event-based stereo visual odometry,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2024, pp. 11977–11983.
- [15] J. Niu, S. Zhong, X. Lu, S. Shen, G. Gallego, and Y. Zhou, “ESVO2: Direct visual-inertial odometry with stereo event cameras,” *IEEE Trans. Robot.*, vol. 41, pp. 2164–2183, 2025.
- [16] U. M. Nunes, R. Benosman, and S.-H. Ieng, “Adaptive global decay process for event cameras,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 9771–9780.
- [17] P. Chen, W. Guan, and P. Lu, “ESVIO: Event-based stereo visual inertial odometry,” *IEEE Robot. Automat. Lett.*, vol. 8, no. 6, pp. 3661–3668, Jun. 2023.
- [18] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, “Ultimate SLAM? Combining events, images, and IMU for robust visual SLAM in HDR and high-speed scenarios,” *IEEE Robot. Automat. Lett.*, vol. 3, no. 2, pp. 994–1001, Apr. 2018.
- [19] Z. Wang, Y. Ge, K. Dong, I.-M. Chen, and J. Wu, “FAST-LIEO: Fast and real-time LiDAR-inertial-event-visual odometry,” *IEEE Robot. Automat. Lett.*, vol. 10, no. 2, pp. 1680–1687, Feb. 2025.
- [20] C. Hertzberg, R. Wagner, U. Frese, and L. Schröder, “Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds,” *Inf. Fusion*, vol. 14, no. 1, pp. 57–77, 2013.
- [21] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, “FAST-LIO2: Fast direct LiDAR-inertial odometry,” *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.
- [22] X. Lagorce, G. Orchard, F. Galluppi, B. E. Shi, and R. B. Benosman, “HOTS: A hierarchy of event-based time-surfaces for pattern recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1346–1359, Jul. 2017.
- [23] P. Chen et al., “ECMD: An event-centric multisensory driving dataset for SLAM,” *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 407–416, Jan. 2024.
- [24] L. Gao et al., “VECToR: A versatile event-centric benchmark for multi-sensor SLAM,” *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 8217–8224, Jul. 2022.
- [25] M. Grupp, “evo: Python package for the evaluation of odometry and SLAM,” 2017. [Online]. Available: <https://github.com/MichaelGrupp/evo>