ESVIO: Event-Based Stereo Visual Inertial Odometry

Peiyu Chen, Weipeng Guan, and Peng Lu

***Abstract*—Event cameras that asynchronously output low-latency event streams provide great opportunities for state estimation under challenging situations. Despite event-based visual odometry having been extensively studied in recent years, most of them are based on the monocular, while few research on stereo event vision. In this letter, we present ESVIO, the first event-based stereo visual-inertial odometry, which leverages the complementary advantages of event streams, standard images, and inertial measurements. Our proposed pipeline includes the ESIO (purely event-based) and ESVIO (event with image-aided), which achieves spatial and temporal associations between consecutive stereo event streams. A well-design back-end tightly-coupled fused the multi-sensor measurement to obtain robust state estimation. We validate that both ESIO and ESVIO have superior performance compared with other image-based and event-based baseline methods on public and self-collected datasets. Furthermore, we use our pipeline to perform onboard quadrotor flights under low-light environments. Autonomous driving data sequences and real-world large-scale experiments are also conducted to demonstrate long-term effectiveness. We highlight that this work is a real-time, accurate system that is aimed at robust state estimation under challenging environments.**

***Index Terms*—Visual-Inertial SLAM, sensor fusion, aerial systems: perception and autonomy.**

I. Introduction

Event cameras are novel bio-inspired sensors [1], which have a high dynamic range (140 dB compared to 60 dB of standard cameras) to handle broad illumination conditions. Unlike standard cameras that output fixed-rate image frames, event cameras respond to pixel-level intensity changes and output asynchronous event streams at the latency of microsecond level, which endows these novel sensors to tackle high-speed motion without motion blur.

Most of the existing event-based visual odometers (VO) use monocular event camera [2], [3], [4], while few research on visual odometry based on stereo event cameras [5], [6]. Since event cameras output asynchronous event streams rather than fixed-rate image frames, the traditional image-based instantaneous matching cannot be directly implemented on event streams. For consecutive stereo event streams, merely relying on temporal constraints to extract and match event-corner features might lead to many false correspondences. Temporal deviations between event streams, noise impacts, different contrast sensitivity of sensors, etc. cause the above problem. Therefore, it is crucial to extract apposite event-corner features and design proper constraints to achieve data association between stereo event-corner features.

Compared with standard cameras, event cameras do not suffer from motion blur under aggressive motion. However, when the relative motion between event cameras and scenes is restricted, e.g. in the stationary state, event streams might not be reliably generated and transmitted, whereas standard cameras are able to provide rich information most of the time (e.g. low-speed motion and well-lit scenes).Observing this complementarity, leveraging both of the advantages of the aforementioned different sensors in combination with an inertial measurement unit (IMU) results in a robust and accurate visual-inertial odometry (VIO) pipeline [3], [4].

In this letter, we propose, to the best of our knowledge, the first published event-based stereo visual-inertial odometry (ESVIO). Our contributions are summarized as follows:

1) In order to achieve robust state estimation under aggressive motion and low-light scenarios, we propose the first purely event-based stereo inertial odometry (ESIO) pipeline with sliding windows graph-based optimization, and further extend it with image-aided (ESVIO) which tightly integrates stereo event streams, stereo image frames, and IMU together.
2) To tackle the problem of event-based stereo feature tracking and matching, we design geometry-based spatial and temporal data associations in consecutive stereo event streams. The spatial and temporal constraints ensure accurate and reliable state estimation. Moreover, a motion compensation method is designed to emphasize the edge of scenes by warping each event.
3) We evaluate that our ESVIO can achieve state-of-the-art performance on publicly available datasets. We also release a very challenging event-based VO/VIO dataset, featuring aggressive motion and HDR scenarios. Finally, we perform onboard closed-loop quadrotor flight using our ESVIO as the estimator.

The remainder of the letter is organized as follows: Section II introduces the related works. Section III introduces the methodology of our methods. Section IV presents the experiments and results. Finally, the conclusion is given in Section V.

II. Related Works

A. Event-Based Monocular Visual Odometry

Event-based monocular VO has been intensively researched for challenging scenarios in recent years. [7] is the first work using feature tracks to achieve event-based VO, which

Manuscript received 28 December 2022; accepted 13 April 2023. Date of publication 25 April 2023; date of current version 3 May 2023. This letter was recommended for publication by Associate Editor N. Radwan and Editor S. Behnke upon evaluation of the reviewers' comments. This work was supported by General Research Fund under Grant 17204222 and in part by the Seed Funding for Strategic Interdisciplinary Research Scheme and Platform Technology Fund. *(Peiyu Chen and Weipeng Guan contributed equally to this work.) (Corresponding author: Peng Lu.)*

The authors are with the Adaptive Robotic Controls Lab (ArcLab), Department of Mechanical Engineering, The University of Hong Kong, Hong Kong SAR 999077, China (e-mail: chenpyhk@connect.hku.hk; guanwp@hku.hk; lupeng@hku.hk).

Digital Object Identifier 10.1109/LRA.2023.3269950

detects features firstly from grayscale frames and then uses event streams tracked features asynchronously. EVO [2] proposed a monocular event-based parallel tracking-and-mapping philosophy which applies the image-to-model alignment for tracking and Event-based Multi-View Stereo (EMVS) [8] for mapping. [9] proposed the first event-based VIO that tackles the incomplete estimation of scale and provides accurate 6-DoF state estimation based on Extended Kalman Filter (EKF). [10] obtains a discrete number of states based on a spatio-temporal window of event streams, and introduces virtual event frames to achieve nonlinear optimization that refines estimated poses. Ultimate SLAM [3] furthered the aforementioned research by combining event streams, image frames, and IMU measurements with nonlinear optimization, which leverages the complementary advantages of event cameras and standard cameras. [11] adopted a continuous-time framework based on cubic spline for smooth trajectory estimation and fused both event streams and IMU together. DEVO [12] proposed a novel VO based on a hybrid setup of depth and event cameras, which construct a semi-dense depth map by thresholding time-surface maps. EKLT-VIO [13] integrated an accurate state-of-the-art event-based feature tracker EKLT [14] with EKF backend to achieve event-based state estimation on Mars-like datasets. [15] proposed a real-time monocular event-based VIO based on graph optimization, which directly utilizes the asynchronous raw events for feature detection. PL-EVIO [4] extended the above method to leverage the complementary advantages of standard and event cameras, which tightly combined event-based point features, event-based line features, image-based point features, and IMU measurements together.

B. Event-Based Stereo Visual Odometry

In contrast to monocular VO, which requires sufficient parallax to recover the depth of corner features, stereo VO can directly obtain the depth of features at the current timestamp. This solves scale uncertainty and even tracking failure caused by insufficient parallax. However, most of the recent research on stereo event cameras has focused on depth estimation and constructing semi-dense or dense maps [16], [17], with less research on VO/SLAM fields. ESVO [5] proposed the first event-based stereo VO pipeline, which achieves parallel 3D semi-dense mapping thread and tracking thread by maximizing the spatio-temporal consistency of stereo event streams. [6] adopted stereo feature detection and matching with the geometry method, which adopts reprojection error minimization to achieve pose estimation. However, these algorithms merely use events to estimate the state, which might result in tracking failure when the system is stationary. In addition, these aforementioned approaches without combining IMU might lead to losses of visual tracks under textureless areas. Our work fills a gap based on combining stereo event streams, stereo image frames, and IMU together, which can operate in real-time under high-resolution event streams with better performance than previously proposed methods.

III. Methodology

Since the procedure of the image measurement is very similar to that of event streams, we only introduce the ESIO in this section. The core of our framework lies in the pre-processing of raw event streams using motion compensation (Section III-A) and the data association between consecutive stereo event streams in temporal and spatial (Section III-B). After that, we design the event-based constraint for graph optimization (Section III-C). The pipeline of ESIO can be represented by the ESVIO (shown in Fig. 2) without image-based processing.

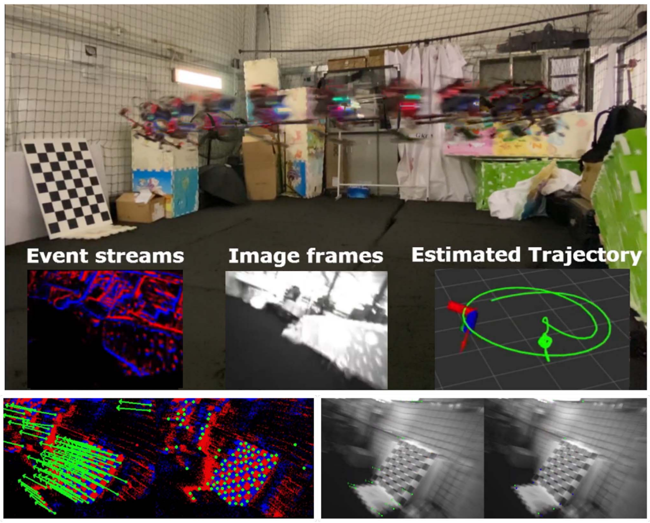

Fig. 1. Our ESVIO provides robust and accurate, real-time pose feedback for drones under aggressive motion. Events provide rich and reliable features, while only a few features are tracked in image frames in high-speed motion. **Left bottom:** stereo event-based feature tracking. **Right bottom:** stereo image-based feature tracking.

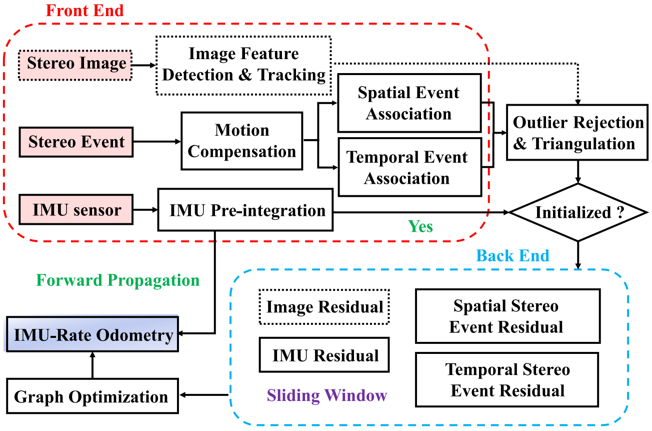

Fig. 2. The structure of our ESVIO and ESIO pipeline.

A. Motion Compensation for Event Streams

Motion compensation corrects the curved event streams by aligning events corresponding to the same scene edge. [18] only uses the angular velocity of IMU to achieve rotational compensation while ignoring the effect of translation. [19] achieve the rotational and translational compensation by IMU and depth camera respectively. We design a motion compensation approach to correct each raw event position, which uses the angular velocity from the IMU sensor and the linear velocity from our ESVIO back-end to achieve rotational and translational compensation

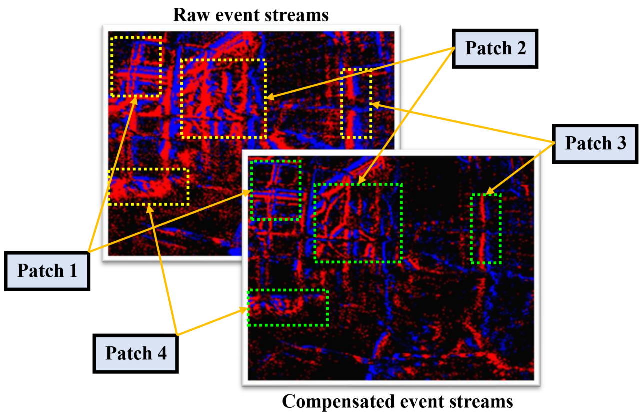

Fig. 3. The event streams without and with motion compensation.

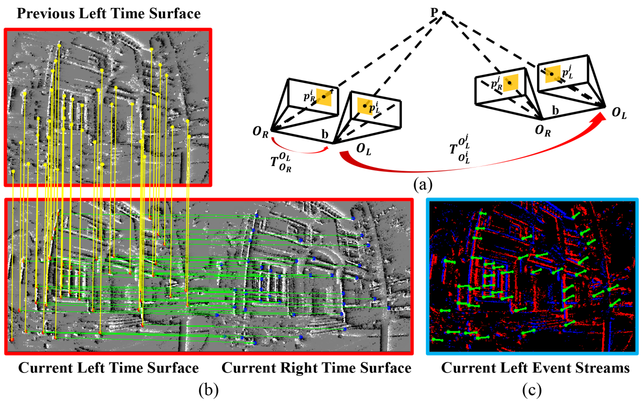
Previous Left Time Surface
P
p_R^i
p_L^i
p_R^j
p_L^j
O_R
b
O_L
$T_{O_R}^{O_L}$
$T_{O_L^i}^{O_L^j}$
O_R
b
O_L
(a)
Current Left Time Surface
Current Right Time Surface
(b)
Current Left Event Streams
(c)

Fig. 4. Stereo event-corner features: (a) Geometry principle; (b) Temporally and spatially associating the event-corner features on the time surface; (c) Event-corner features tracking on the event streams.

respectively. Since our motion compensation utilizes the estimated velocity from the back-end, it works after the successful initialization.

Given the kth event as $e_k = \{l_k, t_k, p_k\}$, where $l_k = \{x_k, y_k\}$ represents the pixel location of the event e_k. t_k is the timestamp and p_k represents its polarity. The event e_k is warped from t_k to t_{ref}, and the compensated location ${}^{ref}l_k$ define as

$$ {}^{ref}l_k = \mathcal{M}[x_k, y_k, t_k, \Theta] \tag{1} $$

where $\mathcal{M}$ is the motion compensation function. Θ represents compensation parameters. Since the short time interval Δt between t_{ref} and t_k, we assume that the motion during this period is uniform motion, thereby the ego-motion estimation of each event can be formulated as follow:

$$ {}^{ref}\mathbf{R}_k = \exp\left((\tilde{\boldsymbol{\omega}}_k - \mathbf{b}_g(t_k) - \mathbf{n}_g(t_k))\Delta t\right) \tag{2a} $$

$$ {}^{ref}\mathbf{L}_k = {}^{ref}\mathbf{R}_k \mathbf{L}_k + \mathbf{v}_{ref}\Delta t \tag{2b} $$

where $\mathbf{exp}$ denotes the exponential map $se(3)\rightarrow SE(3)$. ${}^{ref}\mathbf{R}_k$ is the rotation matrix converted from the Euler angle $\boldsymbol{\omega_k}\Delta t$, $\boldsymbol{\omega}_k = \tilde{\boldsymbol{\omega}}_k - \mathbf{b}_g(t_k) - \mathbf{n}_g(t_k)$. $\tilde{\boldsymbol{\omega}}_k$ is the measurement of gyroscope, while $\mathbf{b}_g(t_k)$ and $\mathbf{n}_g(t_k)$ are bias and noise variable of gyroscope respectively. $\mathbf{L}_k = \{x_k, y_k, 1\}$ is homogeneous matrix that extended from l_k. $\mathbf{v}_{ref}$ represents the velocity of our ESVIO back-end at t_{ref} timestamp. Finally, we convert ${}^{ref}\boldsymbol{L}_k$ to homogeneous matrix and obtain the compensated location ${}^{ref}l_k$. Fig. 3 compares raw event streams and motion-compensated event streams, where raw event streams produce a certain extent of distortion while the compensated event streams show clear contours of scenes.

B. Event-Based Spatial and Temporal Data Associations

After the motion compensation, the stereo event streams are fed to generate two (positive and negative) surface-of-active-event (SAE) which store the event pixels and timestamps. For the new arrival event streams, the existing event-corner features are firstly temporally tracked by the LK optical approach [20] and then spatially matched in left and right event streams. The event-corner features that are not successfully tracked and matched in the current timestamp would be discarded immediately. While new event-corner features are extracted on the motion-compensated event streams to maintain a minimum number (100-200) of features in each timestamp. Modified from the publicly available implementation of the Arc* algorithm [21] for event-based corner detection, we extract the event corners on the individual event by leveraging the SAE. We only select those events whose timestamps are within a short interval from that of the current time surface, thereby retaining event-corner features at the dense event streams. To reduce the influence of noisy events, we further apply time surface (TS) with polarity as a mask to filter effective event-corner features, and the TS is converted from the SAE in real time. Meanwhile, TS is also used to distribute adjacent event-corner features uniformly by setting a minimum distance $d_{\min}$ value.

The geometry principle of temporal and spatial event-based associations is depicted in Fig. 4(a). To ensure that the matched features between the left and right event streams lie along the epipolar line, we instantaneously match stereo-rectified time surfaces for the spatial association. Stereo event-corner features, $\mathcal{F}_l^i$ and $\mathcal{F}_r^i$, are instantaneously matched by forward and inverse LK optical flow between the left and right time surface at the current timestamp i. Our ESVIO executes spatial and temporal association at each frame. Meanwhile, the temporal association also uses forward and inverse optical flow to track event-corner features of left event streams, $\mathcal{F}_l^i$ and $\mathcal{F}_l^j$, at consecutive two timestamps i and j. In Fig. 4(b), red and blue dots represent event-corner features extracted on the left and right time surface at timestamp i respectively, while yellow dots denote the features on the left time surface at the previous timestamp j. Note that the left bottom and right bottom pictures form the spatial event-based association, while the left bottom and left top pictures form the temporal event-based association. Green lines connect matched event-corner features between the left and right time surfaces at the same timestamp, while yellow lines connect tracked features with two consecutive left time surfaces at i and j. Fig. 4(c) and the bottom of Fig. 1 show our proposed method can achieve correct temporal and spatial association in consecutive stereo event-corner features, even when the image-based tracking is failed caused by the motion blur.

Finally, we recover the inverse depth of event-corner features by RANSAC outlier rejection and triangulation. As depicted in Fig. 4(a), the matched corner features of the 3D point P on the imaging plane of left and right event cameras at timestamp i are p_L^i and p_R^i respectively. The inverse depth of P can be formulated through epipolar geometry and triangulation between p_L^i and p_R^i. Similarly, p_L^i and p_L^j can also be used to recover the inverse depth. Based on these associated event-corner features, corresponding residual constraints can be constructed to conduct graph-based optimization.

C. The Construction of Event-Based Residual Constraint for the Graph-Based Optimization

The full state vector in the sliding window is defined as

$$\boldsymbol{\chi} = \left[\mathbf{x}_{b_0}, \ldots, \mathbf{x}_{b_n}, \mathbf{x}_e^b, \mathbf{x}_c^b, \boldsymbol{\Lambda}_{es}, \boldsymbol{\Lambda}_{et}, \boldsymbol{\Lambda}_c\right] \tag{3a}$$

$$\mathbf{x}_{b_k} = \left[\mathbf{p}_{b_k}^w, \mathbf{q}_{b_k}^w, \mathbf{v}_{b_k}^w, \mathbf{b}_{a_k}, \mathbf{b}_{g_k}\right] \quad k \in [0, n] \tag{3b}$$

where $\mathbf{x}_{b_k}$ is the state of IMU at timestamp k in the world frame, which consists of the position $\mathbf{p}_{b_k}^w$, the orientation quaternion $\mathbf{q}_{b_k}^w$, the velocity $\mathbf{v}_{b_k}^w$, the accelerometer bias $\mathbf{b}_{a_k}$ and the gyroscope bias $\mathbf{b}_{g_k}$. $\mathbf{x}_e^b$ and $\mathbf{x}_c^b$ are the extrinsic transformation from event cameras and standard cameras to IMU respectively. $\boldsymbol{\Lambda}_{es} = [\lambda_{es_0}, \lambda_{es_1}, \ldots, \lambda_{es_l}]$, $\boldsymbol{\Lambda}_{et} = [\lambda_{et_0}, \lambda_{et_1}, \ldots, \lambda_{et_l}]$, $\lambda_{es_l}, \lambda_{et_l}$ represent the inverse depth of the event-corner features es_l, et_l respectively. $\boldsymbol{\Lambda}_c = [\lambda_{c_0}, \lambda_{c_1}, \ldots, \lambda_{c_l}]$, λ_{c_l} is the inverse depth of the image-based features c_l. n is the total number of keyframes, and l is the total number of features in the sliding window.

Combining event, image, and IMU residual terms, the event-visual-inertial odometry can be formulated as the joint nonlinear optimization problem as follow

$$\min_{\boldsymbol{\chi}} \left(\sum_{k \in b} \left\| \mathbf{r}_b(\hat{\mathbf{z}}_{b_{k+1}}^{b_k}, \boldsymbol{\chi}) \right\|_{\Omega_b}^2 + \sum_{(l,k) \in es} \left\| \mathbf{r}_{es}(\hat{\mathbf{z}}_{es_k}^l, \boldsymbol{\chi}) \right\|_{\Omega_{es}}^2 \right.$$
$$\left. + \sum_{(l,k) \in et} \left\| \mathbf{r}_{et}(\hat{\mathbf{z}}_{et_k}^l, \boldsymbol{\chi}) \right\|_{\Omega_{et}}^2 + \sum_{(l,k) \in c} \left\| \mathbf{r}_c(\hat{\mathbf{z}}_{c_k}^l, \boldsymbol{\chi}) \right\|_{\Omega_c}^2 \right) \tag{4}$$

where $\mathbf{r}_b(\hat{\mathbf{z}}_{b_{k+1}}^{b_k}, \boldsymbol{\chi})$ is the residual for IMU measurement with information matrix Ω_b. $\mathbf{r}_{es}(\hat{\mathbf{z}}_{es_k}^l, \boldsymbol{\chi})$ and $\mathbf{r}_{et}(\hat{\mathbf{z}}_{et_k}^l, \boldsymbol{\chi})$ are the residuals for event-based spatial and temporal association measurement, with corresponding information matrix Ω_{es} and Ω_{et} respectively. $\mathbf{r}_c(\hat{\mathbf{z}}_{c_k}^l, \boldsymbol{\chi})$ represents the residuals for standard cameras measurement with information matrix Ω_c. The definition of event-based residuals will be presented below, while other residual terms can be found in [4], [22].

For the stereo event cameras, we construct the spatial association factor $\mathbf{r}_{es}(\hat{\mathbf{z}}_{es_k}^l, \boldsymbol{\chi})$ between the left and right event streams and the temporal association factor $\mathbf{r}_{et}(\hat{\mathbf{z}}_{et_k}^l, \boldsymbol{\chi})$ in the consecutive event streams. Consider the lth feature that is observed in the ith right event stream, the residual for the event-corner feature observation in the ith left event stream is defined as:

$$\mathbf{r}_{es} = \begin{bmatrix} u_{les_i}^l \\ v_{les_i}^l \end{bmatrix} - \pi_e \left(\mathbf{T}_{re}^{le} \pi_e^{-1} \left(\frac{1}{\lambda_{es}}, \begin{bmatrix} u_{res_i}^l \\ v_{res_i}^l \end{bmatrix} \right) \right) \tag{5}$$

where $[u_{les_i}^l, v_{les_i}^l]$ is the observation of the lth event-corner feature in the ith left event stream. $[u_{res_i}^l, v_{res_i}^l]$ is the same event-corner feature in the ith right event stream. π_e and π_e^{-1} are the projection and back-projection functions of the event camera respectively. $\mathbf{T}_{re}^{le}$ represents the extrinsic transformation from the right to left event camera.

Consider the lth feature that is first observed in the ith left event stream, the residual for the event-corner feature observation in the kth left event stream is defined as:

$$\mathbf{r}_{et} = \begin{bmatrix} u_{et_k}^l \\ v_{et_k}^l \end{bmatrix}$$
$$- \pi_e \left((\mathbf{T}_{le}^b)^{-1} \mathbf{T}_w^{b_k} \mathbf{T}_{b_i}^w \mathbf{T}_{le}^b \pi_e^{-1} \left(\frac{1}{\lambda_{et}}, \begin{bmatrix} u_{et_i}^l \\ v_{et_i}^l \end{bmatrix} \right) \right) \tag{6}$$

where $[u_{e_k}^l, v_{e_k}^l]$ is the observation of the lth event-corner feature in the kth event stream. $[u_{e_i}^l, v_{e_i}^l]$ is the same event-corner feature in the ith event stream. $\mathbf{T}_{le}^b$ represents the extrinsic transformation from the left event camera to the body coordinate. $\mathbf{T}_{b_i}^w$ indicates the pose of the body center related to the world frame at timestamp i, $\mathbf{T}_w^{b_k}$ is the transpose of the pose of the body coordinate in the world frame at the kth keyframe.

IV. Evaluation

We perform both dataset and real-world experiments to evaluate our proposed methods. We first evaluate our proposed ESIO and ESVIO in the self-collected dataset which is acquired by two DAVIS346 (346 × 260, event-sensor, image-sensor, IMU sensor) and VICON. It contains extremely fast 6-Dof motion and scenes with HDR. In Section IV-B, we compare our methods with other event-based and image-based methods on two publicly available datasets: MVSEC [23] and VECtor [24]. We perform quantitative analysis to evaluate the accuracy of our system. The accuracy is measured with mean position error (MPE, %) and mean rotation error (MRE, °/m) aligning the estimated trajectory with ground truth using 6-DOF transformation (in SE3), which is calculated by the tool [25]. Finally, in Section IV-C we evaluate our ESVIO in the onboard quadrotor flighting. While Sections IV-D1 and IV-D2 perform the evaluation of the autonomous-driving dataset and outdoor large-scale environment, respectively. All experiments run in real-time on an Intel NUC computer equipped with Intel i7-1260P, 32 GB RAM, and Ubuntu 20.04 operation system.

A. Evaluation of Our ESVIO in Challenging Situations

1) Experiment Data Description: The dataset contains stereo event data at 60 Hz and stereo image frames at 30 Hz with resolution in 346×260, as well as IMU data at 1000 Hz. Timestamps between all sensors are synchronized in hardware. We also provide ground truth poses from a motion capture system VICON at 50 Hz for each sequence, which can be used for accuracy comparison. The dataset consists of handheld sequences including rapid motion and HDR scenarios. The full setup including the attached infrared filter can be seen in Fig. 5. These two DAVIS346 are rigidly attached with a baseline of 6.0 cm and USB 3.0 interfaces are used to transmit sensor measurements to the NUC. However, since the limitation of our hardware and cost, we use DAVIS346-COLOR and DAVIS346-MONO for the data collection. Although this might introduce some artificial inconsistency, we think it is acceptable for the method evaluation. The DAVIS comprises an image camera and event camera on the same pixel array, thus calibration can be

TABLE I
THE ACCURACY COMPARISON OF OUR ESVIO WITH OTHER IMAGE-BASED OR EVENT-BASED METHODS ON HKU DATASET

Sequence	ORB-SLAM3 [26] Stereo VIO	VINS-Fusion [22] Stereo VIO	USLAM [10] Mono EIO	USLAM [3] Mono EVIO	PL-EVIO [4] Mono EVIO	**Our ESIO** **Stereo EIO**	**Our ESIO+** **Stereo EIO**	**Our ESVIO** **Stereo EVIO**
	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE
hku_agg_translation	0.15 / 0.075	0.11 / 0.019	16.22 / 0.45	0.59 / 0.020	**0.07** / 0.091	0.59 / 0.16	0.55 / 0.16	0.10 / **0.016**
hku_agg_rotation	0.35 / 0.11	1.34 / 0.024	*failed*	3.14 / 0.026	0.23 / 0.12	1.33 / 0.048	0.78 / 0.045	**0.17** / **0.015**
hku_agg_flip	**0.36** / 0.39	1.16 / 2.02	11.15 / 2.11	6.86 / 2.04	0.39 / 2.23	3.79 / 0.23	3.17 / 0.23	**0.36** / **0.12**
hku_agg_walk	*failed*	*failed*	*failed*	2.00 / 0.16	0.42 / 0.14	1.49 / 0.23	1.30 / 0.23	**0.31** / **0.026**
hku_hdr_circle	0.17 / 0.12	5.03 / 0.60	0.92 / 0.58	1.32 / 0.54	**0.14** / 0.62	1.38 / 0.10	0.46 / 0.099	0.16 / **0.035**
hku_hdr_slow	0.16 / 0.058	0.13 / **0.026**	*failed*	2.80 / 0.099	0.13 / 0.068	0.29 / 0.38	0.31 / 0.39	**0.11** / 0.028
hku_hdr_tran_rota	0.30 / 0.042	0.11 / 0.021	*failed*	2.64 / 0.13	0.10 / 0.064	0.84 / 0.30	0.91 / 0.31	**0.10** / **0.018**
hku_hdr_agg	0.29 / 0.085	1.21 / 0.27	*failed*	2.47 / 0.27	0.14 / 0.30	2.33 / 0.16	1.41 / 0.14	**0.10** / **0.021**
hku_dark_normal	*failed*	0.86 / 0.028	*failed*	2.17 / 0.031	1.35 / 0.081	**0.30** / 0.12	0.35 / 0.12	0.42 / **0.015**
Average	0.16 / 0.12	0.76 / 0.38	5.06 / 1.05	1.69 / 0.39	0.26 / 0.41	0.89 / 0.19	0.66 / 0.19	**0.14** / **0.033**

**EIO means event-based inertial odometry, EVIO means event-based VIO with image-aided*

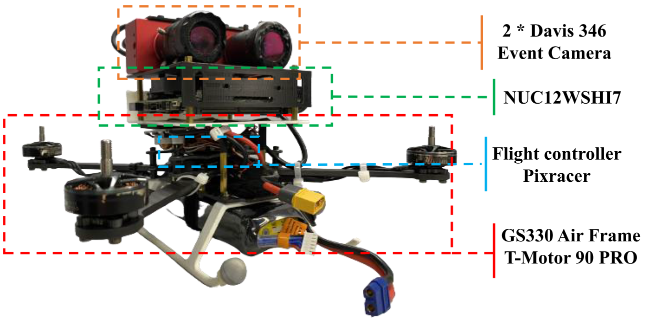

Fig. 5. Our self-designed quadrotor platform.

done using standard image-based methods, such as Kalibr[1], on the image frames and then are applied to the event camera. For the benefit of the research community, we also release the dataset and the configuration files on our project website.

2) Methods Evaluation: Table I compares the performance of our ESIO and ESVIO with the other state-of-the-art event-based or image-based systems. Our ESIO has good performance, especially for the sequence *hku_agg_walk* and *hku_dark_normal*, our ESIO still can produce reliable and accurate pose estimation even when the state-of-the-art image-based VIO method, ORB-SLAM3, fails. Due to motion blur, both ORB-SLAM3 and VINS-Fusion fail to extract reliable features in *hku_agg_walk* sequence, resulting in system failure. In the *hku_dark_normal* sequence, ORB-SLAM3 cannot extract any feature due to poor light conditions (more qualitative details is in the supplementary material). As for the MRE evaluation criterion, our ESVIO shows significant improvement compared to other advanced algorithms, e.g. the average MRE of ESVIO is 0.033°/m while the value of ORB-SLAM3 is 0.12°/m. Even if it does not perform too much improvement in MPE compared with our previous work PL-EVIO [4] in most sequences, ESIO and ESVIO are still a breakthrough for event-based stereo VIO. For the motion compensation version (ESIO+), it shows effective improvement in most of the data sequences compared to ESIO, e.g. the average MPE of ESIO+ is 0.66% compared to 0.89% of ESIO, which is opposite to the conclusion from Ref. [4]. This might be thanks to the well-designed motion-compensated methods that process the reliable and optimized compensated measurements from the ESVIO back-end.

Note that we also evaluate EVO [2] and ESVO [5] in our self-collected datasets, but they failed in all sequences. This might be caused by three factors: Firstly, both EVO and ESVO have strict initialization requirements. For example, EVO requires running in a uniform scene for a few seconds to boost the system. Secondly, they are sensitive to parameter tuning, even in their open-source project, they use different parameters for different sequences in the same scenarios. We might fail to correctly tune parameters for their successful running. Finally, our dataset is so challenging that only reliable methods can perform well.

B. Evaluation of Our ESVIO on Public Datasets

In this section, we evaluate our ESVIO on publicly available datasets. The VECtor [24] dataset consists of a hardware-synchronized sensor suite that includes stereo event cameras, stereo standard cameras, an RGB-D sensor, a LiDAR, and an IMU. It covers the full spectrum of 6 DoF motion dynamics, environment complexities, and illumination conditions for both small and large-scale scenarios. To the best of our knowledge, we provide the first results on this new event-based dataset. For the MVSEC [23], we select the sequence captured in the indoor flying room. We use the stereo event camera (640 × 480) and the regular stereo camera (1224 × 1024) from the VECtor, and the DAVIS346 (346 × 260 for both event and image) from the MVSEC, for evaluation, respectively.

As can be seen in Table II, our proposed ESVIO can achieve fairly good results in most of the sequences. Although the MPE criterion of ORB-SLAM3 is slightly better than ours in some sequences (e.g. *robot-normal*, *desk-normal*, *mountain-normal*), our ESVIO provides more reliable and accurate results in most of the sequences under harsh situations with HDR or aggressive motion. The proposed ESVIO is more precise than our previous PL-EVIO, especially in large-scale environments, this might be due to the better event-corner depth estimation. While the traditional event-based methods [2], [3], [5] failed in most of the sequences in these two datasets.

Please note that we think that parameter tuning is infeasible. Therefore, we evaluate our methods using fixed parameters for all sequences during the evaluations. However, the generalization capability of [2] and [5] is slightly poor. Although we have put the utmost effort to tune parameters, they fail in most sequences. Besides, we emphasize real-time performance when

[1] [Online]. Available: https://github.com/ethz-asl/kalibr

TABLE II
THE ACCURACY COMPARISON OF OUR ESVIO WITH OTHER IMAGE-BASED OR EVENT-BASED METHODS ON PUBLIC DATASET

Sequence		ORB-SLAM3 [26] Stereo VIO	VINS-Fusion [22] Stereo VIO	EVO [2] Mono EO	ESVO [5] Stereo EO	Ultimate SLAM [3] Mono EVIO	PL-EVIO [4] Mono EVIO	**Our ESVIO** **Stereo EVIO**
		MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE	MPE / MRE
VECtor [24]	corner-slow	1.49 / 14.28	1.61 / 14.06	4.33 / 15.52	4.83 / 20.98	4.83 / 14.42	2.10 / 14.21	**1.49** / **14.03**
	robot-normal	0.73 / 1.18	**0.58** / 1.18	3.25 / 2.00	*failed*	1.18 / **1.11**	0.68 / 1.25	1.08 / 1.17
	robot-fast	0.71 / 0.70	*failed*	*failed*	*failed*	1.65 / 0.56	**0.17** / 0.74	0.20 / **0.56**
	desk-normal	**0.46** / 0.41	0.47 / **0.36**	*failed*	*failed*	2.24 / 0.56	3.66 / 0.45	0.61 / 0.38
	desk-fast	0.31 / 0.41	0.32 / 0.33	*failed*	*failed*	1.08 / 0.38	0.14 / 0.48	**0.13** / **0.32**
	sofa-normal	**0.15** / 0.41	0.13 / 0.40	*failed*	1.77 / 0.60	5.74 / **0.39**	0.19 / 0.46	0.16 / 0.40
	sofa-fast	0.21 / 0.43	0.57 / **0.34**	*failed*	*failed*	2.54 / 0.36	**0.17** / 0.47	0.17 / 0.35
	mountain-normal	**0.35** / 1.00	4.05 / 1.05	*failed*	*failed*	3.64 / 1.06	4.32 / **0.76**	0.59 / 0.77
	mountain-fast	2.11 / 0.64	*failed*	*failed*	*failed*	4.13 / 0.62	**0.13** / 0.56	0.16 / **0.45**
	hdr-normal	0.64 / 1.20	1.27 / 1.10	*failed*	*failed*	5.69 / 1.65	4.02 / 1.52	**0.57** / **1.06**
	hdr-fast	0.22 / 0.45	0.30 / 0.34	*failed*	*failed*	2.61 / 0.34	**0.20** / 0.50	0.21 / **0.33**
	corridors-dolly	**1.03** / 1.37	1.88 / 1.37	*failed*	*failed*	*failed*	1.58 / 1.37	1.13 / **1.33**
	corridors-walk	1.32 / 1.31	0.50 / 1.31	*failed*	*failed*	*failed*	0.92 / **1.31**	**0.43** / 1.32
	school-dolly	0.73 / 1.02	1.42 / 1.06	*failed*	10.87 / 1.08	*failed*	2.47 / 0.97	**0.42** / **0.73**
	school-scooter	0.70 / **0.49**	**0.52** / 0.61	*failed*	9.21 / 0.63	6.40 / 0.61	1.30 / 0.54	0.59 / 0.56
	units-dolly	7.64 / 0.41	4.39 / 0.42	*failed*	*failed*	*failed*	5.84 / 0.44	**3.43** / **0.022**
	units-scooter	6.22 / **0.22**	4.92 / 0.24	*failed*	*failed*	*failed*	5.00 / 0.42	**2.85** / 0.39
MVSEC [23]	Indoor Flying 1	5.31 / 0.37	1.50 / 0.13	5.09 / 0.92	4.00 / 0.50	*failed*	1.35 / **0.11**	**0.94** / 0.14
	Indoor Flying 2	5.65 / 0.41	6.98 / 0.15	*failed*	3.66 / 0.43	*failed*	1.00 / 0.16	**1.00** / **0.11**
	Indoor Flying 3	2.90 / 0.30	0.73 / 0.048	2.58 / 1.25	1.71 / 0.18	*failed*	0.64 / 0.065	**0.47** / **0.043**
	Indoor Flying 4	6.99 / 0.79	3.62 / 0.39	*failed*	*failed*	**2.77** / **0.14**	5.31 / 0.23	5.55 / 0.21

The bold values highlight the best accuracy result among the evaluated algorithms in the sequence.

TABLE III
RUNNING TIME OF OUR MODULES IN DIFFERENT RESOLUTION EVENT CAMERAS (MS)

Modules	346 × 260	640 × 480
Motion compensation	2.70	11.11
Creation of event representation	5.12	15.47
Spatial event association	0.82	2.57
Temporal event association	0.83	3.31
The whole process of front-end	10.44	35.69
Back-end optimization	19.30	35.59

evaluating our methods. Table III illustrates the running time of our modules under different resolutions. We also provide a qualitative comparison between our method and the other methods in the accompanying video[2] and supplementary material.

Last but not the least, although our proposed ESVIO achieves satisfactory results, it still has limitations in the low-texture environment. For example, the scenarios in sequence *units-dolly* and *units-scooter* are so special that the visual-only method might be easy to degenerate or mismatch during the loop-closure detection. This also indicates that either the event camera or the standard camera has limitations. Although event cameras play a complementary role to the traditional image-based method, event-based multi-sensor fusion, especially combining non-vision-based sensors (such as GPS, and lidar), should be further developed to exploit the advantage of different sensors.

C. Indoor Quadrotor Flight Evaluation

To further demonstrate the practicability of our methods, we perform real-world experiments on a self-designed quadrotor platform. We choose Pixracer autopilot as our flight platform with T-Motor F90 PRO, as shown in Fig. 5. The states estimate from our ESVIO is used to provide onboard pose feedback control for the quadrotor. The quadrotor is commanded to follow a circular pattern eight times continuously during the experiment. The robust and accurate onboard state estimates of our ESVIO enable real-time feedback control. Meanwhile, we also record the ground truth from VICON for further quantitative evaluation.

1) Quadrotor Flight in HDR Scenarios [3]: We show the relative pose error (RPE) of our ESVIO against the VICON in Fig. 6(a). The total trajectory length is 56.0 m. The boxplot [27] shows that the average relative error for the translation part is around 0.1 m. For the rotational part, the average relative error is around 7°. While there are many outliers from 50 to 60 seconds, which is caused by rapid change in yaw at that moment resulting in the estimated pose being slightly slower than VICON. The root-mean-square error (RMSE) of absolute trajectory error in HDR flight is 0.17 m. Fig. 1 qualitatively evaluates the moment when the yaw angle changes rapidly, where few features can be extracted and tracked by the image thread, while event-corner features can still be tracked well.

2) Quadrotor Flight in Aggressive Motion [4]: In this section, the yaw angle of the commanded pattern is changed drastically, for aggressive motion. The performance of our ESVIO is qualitatively demonstrated in Fig. 1 and quantitatively evaluated in Fig. 6(b). The RMSE of ATE in aggressive flight is 0.26 m. Note that it would have some outliers during the comparison with the VICON. For example, there are some rotation errors of more than 20° within 20-30 seconds. This is caused by VICON's ball is not well observed during the aggressive flight, resulting in an inaccurate measurement of the VICON at that moment. However, our reliable ESVIO state estimator still provides robust and accurate onboard pose feedback for the quadrotor.

[2] [Online]. Available: https://b23.tv/JTkDvCP

[3] [Online]. Available: https://b23.tv/wcZiKzG

[4] [Online]. Available: https://b23.tv/nQUbMjy

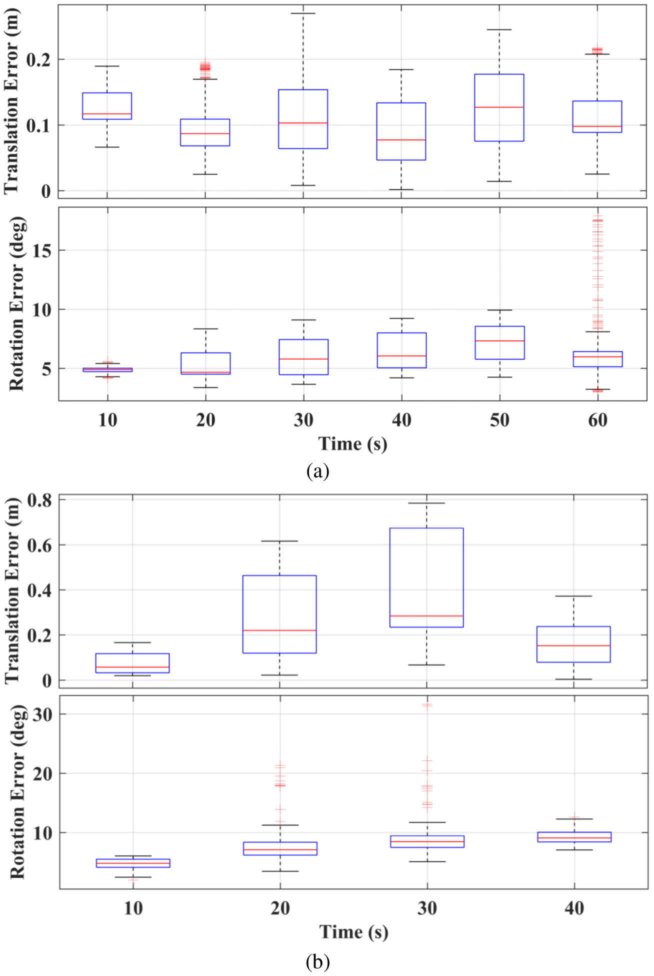

Fig. 6. The relative error comparison of our proposed ESVIO with the VICON: (a) Onboard quadrotor flight in low-illumination conditions; (b) Onboard quadrotor flight in aggressive motion.

D. Outdoor Large-Scale Evaluation

In this section, we evaluate our ESIO and ESVIO in outdoor large-scale environments, including the public-available autonomous driving dataset and the self-collected HKU campus dataset (More details can be seen on our website).

1) DSEC Dataset: DSEC [28] is collected by high-resolution stereo event cameras (640 × 480) under driving scenarios, which is challenging for event-based sensors, as forward motions typically produce considerably fewer events at the center. Qualitative evaluation can be seen in Fig. 7. Since the DSEC dataset does not provide the ground truth 6-DoF poses, we only show the estimated trajectory and the tracking performance of our event-based and image-based features. Both our ESIO (available in supplementary material) and ESVIO can achieve satisfactory results.

2) HKU Large-Scale Environment: This section carried out a large-scale experiment on the HKU campus to illustrate the long-time practicability of our ESVIO, features with large-scale, indoor-outdoor conversion, pedestrians in the scene generating outlier events, etc. The path length of the outdoor evaluation is around 1.8 km and the duration is 34.9 minutes. The evaluation covers the place around 310 m in length, 170 m in width, and 55 m in height changes. Since the VICON is not available outdoors, we only show the qualitative performance and the estimated trajectory overlaid with the Google map for visual comparison. As can be seen from Fig. 8, our ESVIO performs well in long-term motion evaluation, the estimated trajectory is aligned and almost coincide with the Google map.

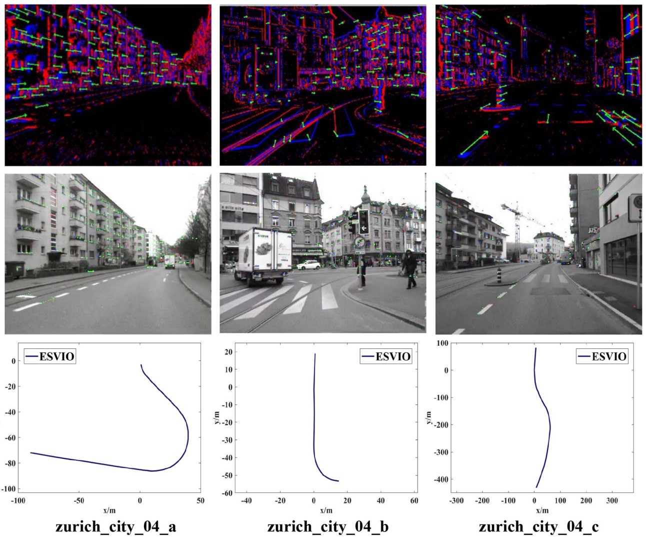

Fig. 7. The qualitative results of ESVIO in DSEC dataset for sequences zurich_city_04 (a) to (c). **Top:** The stereo event-corner feature tracking performance; **Middle:** The stereo image-based feature tracking performance; **Bottom:** The estimated trajectories produced by our ESVIO.

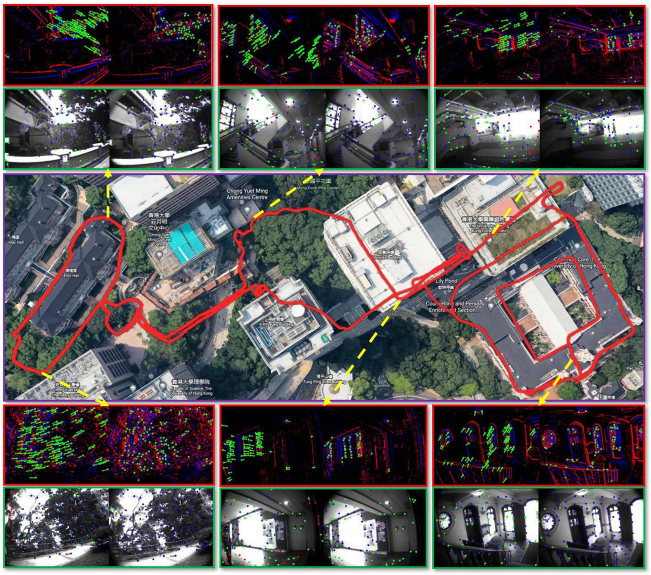

Fig. 8. The estimated trajectory of our ESVIO in the large-scale environment. The detection and tracking situation of the stereo event-corner features and stereo image features during the experiment are also visualized.

V. Conclusion

In this letter, we propose a robust, real-time event-based stereo VIO, ESVIO, which tightly fuses the stereo event streams, stereo image frames, and IMU measurements using sliding windows graph-based optimization. Geometry-based spatial and temporal associations between consecutive stereo event streams are designed to ensure robust state estimation. In addition, the motion compensation approach corrects the curved event streams with IMU and ESVIO back-end to emphasize the contour of scenes. Extensive evaluations demonstrate that our ESVIO achieves superior performance compared to other state-of-the-art algorithms on public datasets and our self-collected challenging datasets. Furthermore, we also perform various onboard closed-loop flights using the proposed ESVIO under low-light scenes and aggressive motion. In our future work, we might further explore the event-based mapping for the quadrotor system which should be able to support autonomous navigation and obstacle avoidance.

MULTIMEDIA MATERIAL

WEBSITE [Online]. Available: https://github.com/arclab-hku/Event_based_VO-VIO-SLAM.

Video Demo [Online]. Available: https://b23.tv/V23SVzC.

Supplementary Material [Online]. Available: https://github.com/arclab-hku/Event_based_VO-VIO-SLAM/tree/main/ESVIO/supply.

REFERENCES

[1] G. Gallego et al., "Event-based vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 44, no. 1, pp. 154–180, Jan. 2022.

[2] H. Rebecq, T. Horstschäfer, G. Gallego, and D. Scaramuzza, "EVO: A geometric approach to event-based 6-DOF parallel tracking and mapping in real time," *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 593–600, Apr. 2017.

[3] 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.

[4] W. Guan, P. Chen, Y. Xie, and P. Lu, "PL-EVIO: Robust monocular event-based visual inertial odometry with point and line features," 2022, *arXiv:2209.12160*.

[5] Y. Zhou, G. Gallego, and S. Shen, "Event-based stereo visual odometry," *IEEE Trans. Robot.*, vol. 37, no. 5, pp. 1433–1450, Oct. 2021.

[6] A. Hadviger, I. Cvišić, I. Marković, S. Vražić, and I. Petrović, "Feature-based event stereo visual odometry," in *Proc. Eur. Conf. Mobile Robots*, 2021, pp. 1–6.

[7] B. Kueng, E. Mueggler, G. Gallego, and D. Scaramuzza, "Low-latency visual odometry using event-based feature tracks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 16–23.

[8] H. Rebecq, G. Gallego, E. Mueggler, and D. Scaramuzza, "EMVS: Event-based multi-view stereo–3D reconstruction with an event camera in real-time," *Int. J. Comput. Vis.*, vol. 126, no. 12, pp. 1394–1414, 2018.

[9] A. Zihao Zhu, N. Atanasov, and K. Daniilidis, "Event-based visual inertial odometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5391–5399.

[10] H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization," in *Proc. Brit. Mach. Vis. Conf.*, 2017, pp. 16–1.

[11] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, "Continuous-time visual-inertial odometry for event cameras," *IEEE Trans. Robot.*, vol. 34, no. 6, pp. 1425–1440, Dec. 2018.

[12] Y. Zuo, J. Yang, J. Chen, X. Wang, Y. Wang, and L. Kneip, "DEVO: Depth-event camera visual odometry in challenging conditions," in *Proc. Int. Conf. Robot. Automat.*, 2022, pp. 2179–2185.

[13] F. Mahlknecht et al., "Exploring event camera-based odometry for planetary robots," *IEEE Robot. Automat. Lett.*, vol. 7, no. 4, pp. 8651–8658, Oct. 2022.

[14] D. Gehrig, H. Rebecq, G. Gallego, and D. Scaramuzza, "EKLT: Asynchronous photometric feature tracking using events and frames," *Int. J. Comput. Vis.*, vol. 128, no. 3, pp. 601–618, 2020.

[15] W. Guan and P. Lu, "Monocular event visual inertial odometry based on event-corner using sliding windows graph-based optimization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 2438–2445.

[16] S. Tulyakov, F. Fleuret, M. Kiefel, P. Gehler, and M. Hirsch, "Learning an event sequence embedding for dense event-based deep stereo," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1527–1537.

[17] Y. Nam, M. Mostafavi, K.-J. Yoon, and J. Choi, "Stereo depth from events cameras: Concentrate and focus on the future," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 6114–6123.

[18] D. Falanga, K. Kleber, and D. Scaramuzza, "Dynamic obstacle avoidance for quadrotors with event cameras," *Sci. Robot.*, vol. 5, no. 40, 2020, Art. no. eaaz9712.

[19] B. He et al., "FAST-dynamic-vision: Detection and tracking dynamic objects with event and depth sensing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 3071–3078.

[20] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. 7th Int/ Joint Conf. Artif. Intell.*, pp. 674–679, vol. 2, 1981.

[21] I. Alzugaray and M. Chli, "Asynchronous corner detection and tracking for event cameras in real time," *IEEE Robot. Automat. Lett.*, vol. 3, no. 4, pp. 3177–3184, Oct. 2018.

[22] T. Qin, J. Pan, S. Cao, and S. Shen, "A general optimization-based framework for local odometry estimation with multiple sensors," 2019, *arXiv:1901.03638*.

[23] A. Z. Zhu, D. Thakur, T. Özaslan, B. Pfrommer, V. Kumar, and K. Daniilidis, "The multivehicle stereo event camera dataset: An event camera dataset for 3D perception," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 2032–2039, Jul. 2018.

[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

[26] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tars, "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.

[27] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 7244–7251.

[28] M. Gehrig, W. Aarents, D. Gehrig, and D. Scaramuzza, "DSEC: A stereo event camera dataset for driving scenarios," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 4947–4954, Jul. 2021.