2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

# IMU-Aided Event-based Stereo Visual Odometry

Junkai Niu*, Sheng Zhong*, Yi Zhou

***Abstract*— Direct methods for event-based visual odometry solve the mapping and camera pose tracking sub-problems by establishing implicit data association in a way that the generative model of events is exploited. The main bottlenecks faced by state-of-the-art work in this field include the high computational complexity of mapping and the limited accuracy of tracking. In this paper, we improve our previous direct pipeline *Event-based Stereo Visual Odometry* in terms of accuracy and efficiency. To speed up the mapping operation, we propose an efficient strategy of edge-pixel sampling according to the local dynamics of events. The mapping performance in terms of completeness and local smoothness is also improved by combining the temporal stereo results and the static stereo results. To circumvent the degeneracy issue of camera pose tracking in recovering the yaw component of general 6-DoF motion, we introduce as a prior the gyroscope measurements via pre-integration. Experiments on publicly available datasets justify our improvement. We release our pipeline as an open-source software for future research in this field.**

## Multimedia Material

Code: https://github.com/NAIL-HNU/ESVIO_AA.git

## I. Introduction

Neuromorphic event-based cameras are bio-inspired visual sensors with asynchronous pixels that report only intensity changes (called "events"). Endowed with microsecond temporal resolution and up to 160 dB dynamic range [1], event cameras are qualified to deal with challenging scenarios that are inaccessible to standard cameras, such as high-speed and/or high-dynamic-range (HDR) tracking [2]–[8], control [9, 10] and Simultaneous Localization and Mapping (SLAM) [11]–[17].

Like its standard-vision counterparts, event-based visual odometry (VO) or SLAM also aims at solving simultaneously the mapping and tracking sub-problems in a recursive manner. The main challenge therein is to extract and maintain effective data association in the event stream, from which the depth information and ego motion can be inferred. From the perspective of how such data association is established, existing methods, including event-based visual inertial odometry, can be divided into two categories: feature-based methods and direct methods.

**Feature-based Methods:** To build on top of existing feature-based VO/SLAM pipelines (*e.g.*, [18, 19]) using standard cameras, researchers have resorted to developing hand-crafted features from event data, such as event corners [20]–[23], which are typically adapted from the original Harris [24] and FAST [25] methods. Additionally, strategies for tracking event corners are presented by [22, 26]. Such event features enable straightforward application of epipolar geometry [27, 28] and Perspective-$n$-Point (PnP) methods [29, 30]. Although the success of these feature-based solutions (*e.g.*, [31]) has been witnessed to some extent, event features, however, are not as theoretically robust as their standard-vision counterparts. This is due to the camera-velocity-dependent nature of event data, which sometimes leads to incomplete observation of junctions. Consequently, feature matching can easily fail in a sudden variation of the event camera's velocity. Another mainstream strategy for feature detection and tracking is inspired by the motion compensation method [32], a unified pipeline for event-based geometric model fitting. This strategy is widely witnessed in event-based VIO pipelines [15, 33], which typically build features from motion-compensated event sets [4] or event images [32], and furthermore, fuse with inertial measurements by means of either the Kalman filter [34] or the keyframe-based nonlinear optimization [35].

**Direct Methods:** Unlike feature-based methods, direct methods refer to those that implicitly establish data associations by exploiting the generative model of events in some ways. Based on the constant-brightness assumption in the

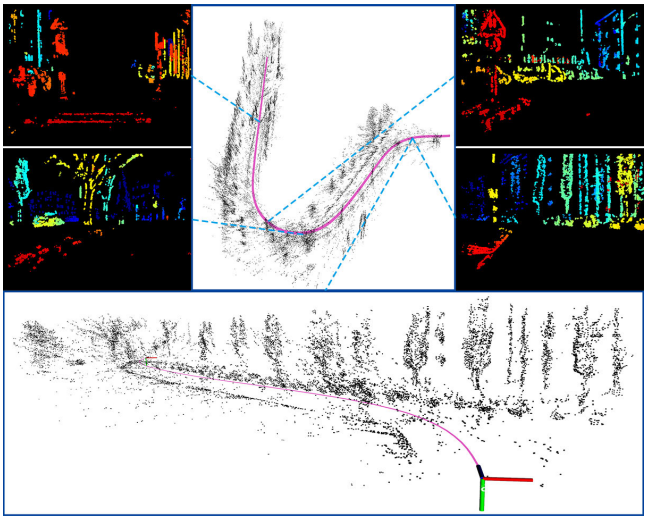


Fig. 1: Illustration of running the proposed system on *DSEC* [36] dataset (seq. *dsec_city04_a*). Middle and bottom: The reconstructed point cloud and resulting trajectory seen from different view points. Left and right: Recovered inverse depth maps at reference perspectives.

All authors are with the Neuromorphic Automation and Intelligence Lab (NAIL) at School of Robotics, Hunan University, Changsha, China.
* denotes equal contribution.
Corresponding author: Yi Zhou. Email: eeyzhou@hnu.edu.cn.
This work was supported by the National Key Research and Development Project of China under Grant 2023YFB4706600.

log intensity domain, Kim *et al*. [12] propose the first direct method that implements three interleaved probabilistic filters solving the sub-problems of mapping, camera pose tracking, and additionally, recovering the log intensity information. To justify that recovering the intensity information is not mandatory, Rebecq *et al*. [14] propose a pure geometric method. The proposed mapping approach determines the 3D location of structures by searching the maximum ray intersection in the disparity space image (DSI), and the camera pose is estimated through a 3D-2D registration process, in which the 3D edge map is aligned to the synthesized event map. These two pioneering frameworks are, however, limited by the requirements of gentle motion in the initialization and slow expansion of the local map. Hence, neither of them has been justified on data collected using a mobile platform. To overcome these issues, Zhou *et al*. present the first event-based VO pipeline (ESVO) using a stereo event camera [17]. The method exploits spatio-temporal consistency of the events across the image planes of the cameras to solve both localization and mapping sub-problems of visual odometry. Nevertheless, ESVO dose not achieve near real-time performance once the spatial resolution of event cameras is $640 \times 480$ pixels or larger. This is mainly due to the large number of redundant operations in the mapping, which is originally caused by the way that edge pixels are determined. Besides, we observe that ESVO's tracking method cannot fully recover the yaw component in general 6-DoF motion, which degrades the accuracy of the recovered trajectory.

The goal of this paper is to lift the above-mentioned limitations of the original ESVO framework. We extend ESVO and present a direct method for visual-inertial odometry with a stereo event camera, , as illustrated in 1. The proposed system achieves better performance of mapping and tracking than ESVO in terms of accuracy and efficiency, due to the following efforts.

***Contribution.***

- A novel image-like representation of events, called adaptive accumulation (AA) of events, which is used for efficient determination of pixel locations associated to instantaneous edges.
- An improved solution to the mapping sub-problem by leveraging both the temporal stereo and static stereo configurations;
- An IMU-aided solution to the camera pose tracking sub-problem that overcomes the insensitivity to the yaw component of general 6-DoF motion in the 3D-2D spatio-temporal registration.

The remaining paper is organized as follows. We first discuss our method by detailing each item listed in the contribution (Sec. II). Then the experimental evaluation is provided in Sec. III, and finally the conclusion is made in Sec. IV.

## II. Methodology

We detail our method in this section. First, we present a pre-processing method for event data that samples efficiently a number of edge-pixel candidates (Sec. II-A). Second, we discuss how to recover the depth information of missing structures in the original ESVO pipeline and further achieve a more complete mapping result by merging results of temporal stereo and static stereo methods (Sec. II-B). Third, we discuss how to improve the 3D-2D spatio-temporal registration by leveraging inertial measurements as a motion prior (Sec. II-C). Finally, we overview the proposed system and discuss the implementation detail (Sec. II-D).

### A. Adaptive Accumulation of Events

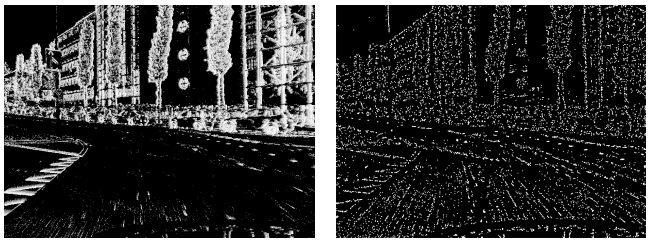
The computational efficiency of ESVO's mapping method is limited by several aspects. One of them is the way that the pixel locations of instantaneous edges are determined. In ESVO, the instantaneous edge map in a virtual reference frame is created by applying motion compensation to events occurred within a short time interval (*e.g*., 10 ms). This operation will become too computationally expensive as a pre-processing step when the event streaming rate exceeds a certain range[1]. Besides, we observe that the extracted events are concentrated in the regions of significant optical flow. To alleviate such an uneven distribution of edge pixels, it is necessary to sample a large number of points, which is redundant and in turn becomes a computation burden to mapping. Therefore, a more efficient way is needed to determine pixels of instantaneous edges.

We propose a novel method inspired by [14, 37]. In [14], the synthesized event map obtained by a naive accumulation[2] of events can be used as an approximate edge map. However, this approximation can become severely inaccurate (*i.e*., blurred edges or invisible edges) when there is a significant parallax in the scene. This is because a global threshold for accumulation cannot handle different local dynamics of events. To this end, we propose a method called adaptive accumulation (AA) that can control the amount of events to be accumulated according to the local event dynamics. Intuitively, the more intensive the local event dynamics, the shorter the time interval for event accumulation. Different from [37] which directly uses the number of events as the metric to control the time length for event accumulation, we apply the contrast of event image [32]. The contrast of image can be quantified by a variety of dispersion metrics, and we simply use the variance loss because of its advantageous accuracy and computation complexity over other alternatives [38]. Although the contrast of event image [32] monotonically increases as the number of accumulated events increases, we observe that, for a fixed-size local patch, the accumulation result by a certain contrast value ($\beta$) can deliver enough visual information before getting obviously blurred.

As shown in Algorithm 1, an AA map is generated as follows. First, we divide the image plane evenly to several small blocks (e.g., block size $w \times w$), and the accumulation

[1]The event streaming rate is in proportion to the scene dynamics, the scene texture and the spatial resolution of the event camera

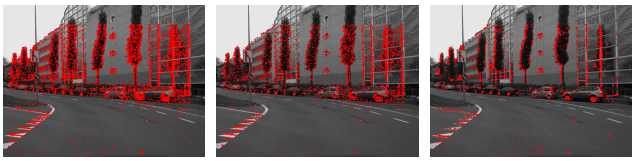
[2]Naive accumulation of events refers to plainly visualizing events' spatial information (*i.e*. image coordinates) on the image plane without any transformation in the spatio-temporal domain.



(a) Time Surface [39] (b) TOS [41]

(c) SILC [40] (d) Adaptive Accumulation

Fig. 2: *Comparison of resulting edge maps from TS [39], TOS [41], SILC [40], and AA, respectively.*



(a) ESVO (10k pts.) (b) ESVO (5k pts.) (c) Ours (5k pts.)

Fig. 3: *Point sampling results of ESVO and our method.* (a) and (b) are ESVO's results that have 10k points and 5k points sampled, respectively. (c) is our sampling result.

of events is carried out in each block independently. For each block area, the contrast of event image [32] is evaluated at a fixed time interval $\delta t$, and the computation terminates if the contrast reaches a certain threshold $\beta$. In this way, patches with different local event dynamics will possibly have a similar number of accumulated events.

The pixel value in an AA map represents the number of events that have been accumulated at this pixel over the locally, adaptively controlled time interval. These pixels with higher values are more likely to be associated to instantaneous edge pixels. To evaluate the effectiveness of the proposed AA method, we compare its result against another three image-like representations, including time surfaces (TS) [39] and two speed-invariant representations [40, 41]. As illustrated in Fig. 2, the result of AA preserves relatively complete edges with the least redundant points while keeping the highest signal-to-noise ratio. Edge pixels used by the following mapping operation are sampled from each patch independently. In general, the bigger the AA pixel value in each patch, the higher possibility the pixel is selected. We further shuffle the sampling pixels to assure an even sampling. We compare the sampling result with edge pixels obtained from ESVO. As shown in Fig. 3, our sampled pixels are more evenly and continuously distributed on the edge

**Algorithm 1** Adaptive Accumulation of Events

**Input:** $N_e$ Events $\{e_k \doteq (x_k, y_k, t_k, p_k)\}_{k=1}^{N_e}$, parameters $\beta$.
**Output:** Adaptive accumulation map $\mathbf{A}(x, y)$.

```
Initialize A(x, y) with all zero elements, and t_last = 0.
Divide A(x, y) to N blocks {A_i | i = 1, 2, ··· N}, and
    assign each block with a boolean flag F_i.
Set {F_i = True | i = 1, 2, ··· N}.
for k = 1, ··· , N_e do
    Get block A_i according to the coordinate of e_k.
    if F_i != True then
        continue.
    end if
    A_i(x_k, y_k)++.
    if ImageContrast(A_i) > β then
        F_i = False.
    end if
end for
```

structures.

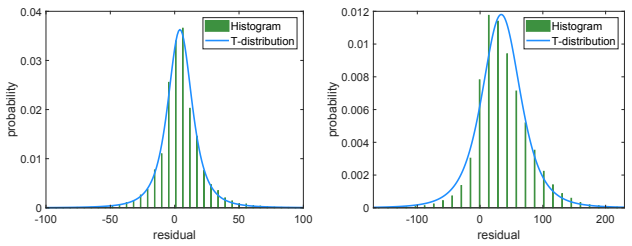
### B. Mapping

The static stereo method in [17] can hardly recover accurate depth of structures that are parallel to the baseline of the stereo camera. This is because the spatio-temporal profile of these structures are only distinctive in one direction, and thus, many false-positive matches will be witnessed during the epipolar-line searching. On the contrary, temporal stereo methods are unlikely affected by this issue. As long as the stereo camera does not move along the baseline, epipolar lines defined between a temporal stereo pair are no longer parallel to the baseline of the static stereo. This assumption always holds for forward looking stereo cameras, *e.g.*, those used in driving scenes. Inspired by [42], we introduce the temporal stereo method to solve this problem.

Let's consider the normally applied horizontal stereo configuration. We divide the sampled edge pixels from Sec. II-A into two groups according to their gradient direction on the corresponding TS. In the first group, we collect pixels at which the magnitude ratio ($\eta$) between the horizontal gradient and the vertical gradient is smaller than a threshold. The remaining sampled pixels make up the second group, and we feed them to the mapping method of the original ESVO. The first group is fed to the proposed temporal stereo method as discussed in the following.

The key to the event-based temporal stereo is effectively exploiting appearance similarity in the spatio-temporal profile. This requires the event representation, on which the stereo data association is established, to possess the speed-invariant property. Thus, time surfaces used in ESVO are no longer applicable in this context. We investigate the temporal stereo matching performance on three representations, including TOS [41], SILC [40] and our AA. We find AA is the optimal choice because of its higher signal-noise ratio.

Given as prior the relative pose between a temporal stereo pair, the proposed temporal stereo method imitates ESVO's




(a) Temporal stereo on AA. (b) Static stereo on TS.

Fig. 4: *Probability distribution (PDF) of the temporal stereo residuals* $r_{temporal}$ *and static stereo residuals* $r_{static}$. The curves (blue) are the *Student's t* fitting results from the empirical histogram (green).

mapping strategy in the sense of applying a block matching plus a nonlinear refinement. To fuse multiple temporal stereo estimates, we follow the way in [17] to obtain the probabilistic characteristics of the temporal stereo results. As shown in Fig. 4, both the temporal stereo residuals $r_{\text{temporal}}$ evaluated on AAs and the static stereo residuals $r_{\text{static}}$ evaluated on TSs approximately obey the *student's t* distribution

$$r \sim St(\mu_r, s_r, \nu_r), \tag{1}$$

where $\mu_r$, $s_r$, $\nu_r$ are the model parameters, namely the mean, scale and degree of freedom. Although the depth estimates from the two methods are probabilistically compatible, they cannot be fused straightforwardly. This is due to the uncertainty of the temporal stereo's result is always much smaller, and it is just caused by the different nature of the heterogeneous representations. To obtain a more complete depth map, we simply merge the results of the two stereo methods. We show the mapping results in Sec. III-A.

### C. Tracking

Our tracking module basically follows the ESVO framework, which takes as input the events, a TS and a local 3D map, and computes the pose of the stereo rig with respect to the map. Let $\mathcal{S}^{\mathcal{F}_{\mathbf{ref}}} = \{\mathbf{x}_i\}$ represent a set of pixels with inverse depth values in the reference frame, and $\mathcal{T}(\mathbf{x}, t)$ and $\overline{\mathcal{T}}(\mathbf{x}, t) = 1 - \mathcal{T}(\mathbf{x}, t)$ be the TS and negative TS at time $t$. The purpose of tracking is, identical to [17], to determine the optimal motion parameters $\boldsymbol{\theta}$ by solving

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{\mathbf{x} \in \mathcal{S}^{\mathcal{F}_{\mathbf{ref}}}} \overline{\mathcal{T}}_{\mathbf{left}}(\boldsymbol{W}(\mathbf{x}, \rho; \boldsymbol{\theta})), \tag{2}$$

where the warp function $\boldsymbol{W}(\mathbf{x}, \rho; \boldsymbol{\theta})$ denotes the transformation from local depth points to the latest negative TS image. And $\boldsymbol{\theta} \doteq (\mathbf{c}^\mathsf{T}, \mathbf{t}^\mathsf{T})^\mathsf{T}$ are the motion parameters, where $\mathbf{c} = (c_1, c_2, c_3)^\mathsf{T}$ are the Cayley parameters [43] for rotation and $\mathbf{t} = (t_x, t_y, t_z)^\mathsf{T}$ is the translation.

To improve the accuracy of pose estimation, we leverage the pre-integration of gyroscope measurements to provide an initial value of rotation in Eq. 2. For an IMU with a 3-axis gyroscope, the measurement of IMU angular velocity can be

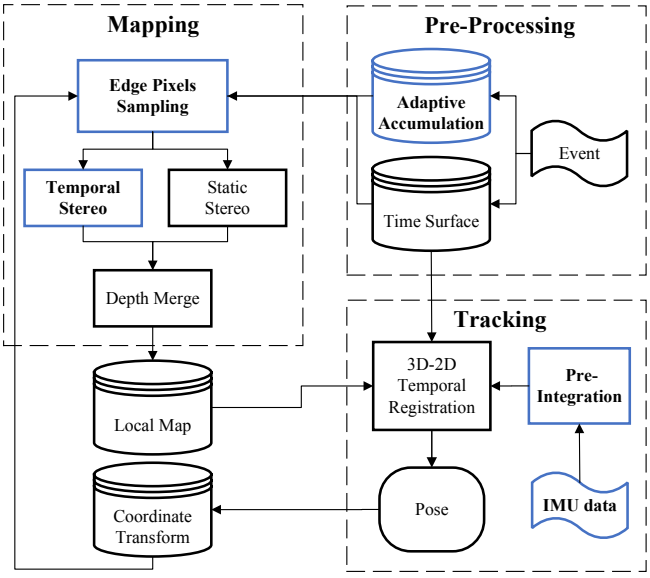



Fig. 5: *Flowchart of the proposed system*. New functions added to the original *ESVO* framework are highlighted in blue. Each module enclosed by a dashed box is executed independently and occupies at least one thread.

represented by

$$\tilde{\boldsymbol{\omega}}^b = \boldsymbol{\omega}^b + \mathbf{b}_g^b + \mathbf{n}_g^b, \tag{3}$$

where $\mathbf{b}_g^b$, $\mathbf{n}_g^b$ are the bias and noise of the gyroscope expressed in the IMU frame. The relative rotation between two successive tracking estimates, *e.g.*, from time $t_i$ to $t_{i+1}$, expressed in the IMU frame $b_i$, can be calculated by

$$\boldsymbol{\gamma}_{b_{i+1}}^{b_i} = \int_{t \in [t_i, t_{i+1}]} \frac{1}{2} \boldsymbol{\Omega}(\tilde{\boldsymbol{\omega}}^{b_t} - \mathbf{b}_g^{b_t} - \mathbf{n}_g^{b_t}) \boldsymbol{\gamma}_{b_t}^{b_i} dt, \tag{4}$$

where $\boldsymbol{\gamma}_{b_t}^{b_i}$ is the quaternion representation of the relative rotation, and $\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor \boldsymbol{\omega} \rfloor_\times & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^\mathsf{T} & 0 \end{bmatrix}$. The bias $\mathbf{b}_g^b$ is initialized empirically and not updated throughout this work. The initial rotation parameter $\mathbf{c}_0$ is obtained via a quaternion-Cayley transformation. We show the benefit brought by using the pre-integration result as a motion prior in Sec. III-B.

### D. System

We extend ESVO [17] with several additional modules. As shown in Fig. 5, the whole system takes as input the events from a stereo event camera and gyroscope measurements from an IMU. In the pre-processing module, the AA and TS are generated at a fixed rate (e.g., 100 Hz), and they are fed to the mapping module together with the tracking results. In the mapping module, the sampled edge pixels are divided according to the classification parameter $\eta$, and the results of the temporal stereo method and static stereo method are merged and inserted to the local map. Given the local map, the most recent TS, and also the initial rotation guess from IMU pre-integration, the tracking module calculates the camera pose. All hyper parameters used are set as in Table. I.

TABLE I: Settings of hyper parameters.

| Parameter | $w$ | $\delta t$ | $\beta$ | $\eta$ |
|---|---|---|---|---|
| Value | 80 pixel (for *DSEC*)<br>30 pixel (for *rpg*) | 2 ms | 0.5 | 0.2 |

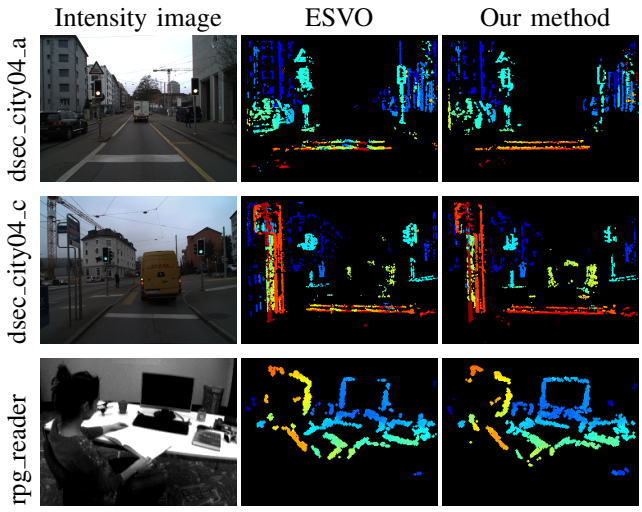



Fig. 6: *Qualitative comparison of mapping results*. Intensity images in the first column are used only for visualization. The second and third columns show the estimated inverse depth map by ESVO and our method, respectively. Note that our method returns more accurate and complete reconstruction results for the horizontal edges. Depth maps are color coded, from red (close) to blue (far) over a black background, in the range 1 m-50 m for the *DSEC* dataset and 0.5 m-10 m for the *rpg* dataset.

## III. Experiments

We evaluate our method in this section using two publicly available datasets, collected using a hand-held stereo event camera (*rpg* dataset) [44] and a mobile-mounted stereo event camera (*DSEC* dataset) [36], respectively. In this section, we first evaluate our solution to the mapping sub-problem quantitatively and qualitatively (Sec. III-A). Second, we test the full system by evaluating the recovered trajectories (Sec. III-B). Finally, we present an analysis on the computational efficiency (Sec. III-C).

### A. Comparison of Mapping: ESVO vs Ours

We compare our mapping results against those of the original ESVO pipeline [17]. As shown in Fig. 6, our results achieve better performance in terms of reconstruction completeness and local depth smoothness. In the results of ESVO, we observe that horizontal structures are typically neither recovered nor estimated accurately. This is due to the aperture problem encountered by the original static stereo matching operation. Compared to the original ESVO pipeline [17], our solution to the mapping sub-problem introduces the temporal stereo estimation between successive observations of the left event camera. This additional stereo matching operation recovers 3D information missed (or inaccurately estimated) by the static stereo operation. We also carry out a quantitative evaluation on the mapping results and use the relative depth error [44] as the evaluation metric. As shown in Table. II, our method outperforms ESVO in terms of mean error, median error, and average relative error in depth estimation.

TABLE II: *Quantitative comparison of mapping result.* The depth range refers to the average of true depth from points evaluated.

| Sequence (depth range) | | ESVO | Ours |
|---|---|---|---|
| dsec_city04_a (9.95 m) | Mean error | 0.66 m | **0.41 m** |
| | Median error | 0.43 m | **0.32 m** |
| | Relative error | 7.8% | **4.3%** |
| dsec_city04_c (6.86 m) | Mean error | 0.83 m | **0.69 m** |
| | Median error | 0.33 m | **0.28 m** |
| | Relative error | 15.3% | **11.6%** |
| dsec_city04_d (14.61 m) | Mean error | 1.01 m | **0.65 m** |
| | Median error | 0.65 m | **0.58 m** |
| | Relative error | 11.2% | **7.1%** |

TABLE III: Translation [%] and rotation [°/m] evaluation results of the proposed method compared to ESVO using relative pose RMSE.

| Sequence | ESVO | | Ours | |
|---|---|---|---|---|
| | **R** | **t** | **R** | **t** |
| rpg_box | 1.92 | 3.79 | **0.82** | **1.88** |
| rpg_monitor | 3.30 | 5.62 | **1.93** | **3.05** |
| rpg_reader | **3.32** | 11.98 | 3.68 | **8.46** |
| dsec_city04_a | 0.10 | 8.12 | **0.08** | **3.46** |
| dsec_city04_b | **0.13** | 8.46 | 0.21 | **6.53** |
| dsec_city04_c | **0.04** | 8.75 | 0.05 | **6.47** |
| dsec_city04_d | 0.08 | 16.17 | **0.05** | **4.48** |
| dsec_city04_e | 0.16 | 21.17 | **0.08** | **5.57** |
| dsec_city11_a | **0.06** | 9.79 | 0.07 | **2.96** |

### B. Full System Evaluation: ESVO vs Ours

We evaluate the full system using two publicly available datasets. The first one is the *rpg* dataset, which features a hand-held stereo event camera moving in a small-scale indoor environment. We report ego-motion estimation results using two standard metrics: relative pose error (RPE) and absolute trajectory error (ATE) [45], and the results are given as root-mean-square errors (RMSEs). As shown in Table. III and Table. IV, our new pipeline outperforms ESVO in terms of these two metrics.

The second dataset used is *DSEC*, which is a stereo event camera dataset for large-scale driving scenarios. We first demonstrate the benefit brought by using inertial measurements as a prior in the camera pose tracking sub-problem. As discussed in Sec. II-C, the introduction of inertial measurements alleviates the problem of 3D-2D spatio-temporal registration being insensitive to recovering rotation in general 6-DoF motion. This improvement is clearly witnessed when the stereo event camera undergoes a translation plus a rotation in the yaw axis. This is justified by the relative pose RMSE results shown in Table. V, where two configurations (with and without IMU) are compared. Additionally, we compare extensively our results against that of ESVO pipeline on the *DSEC* dataset. We apply the evaluation tool provided in [46]

TABLE IV: Absolute trajectory RMSE [t:cm]

| Sequence | ESVO | Ours |
|---|---|---|
| rpg_box | 9.5 | **5.0** |
| rpg_monitor | 5.8 | **2.8** |
| rpg_reader | 6.6 | **3.7** |
| dsec_city04_a | 371.1 | **105.0** |
| dsec_city04_b | 116.6 | **66.7** |
| dsec_city04_c | 1357.1 | **637.9** |
| dsec_city04_d | 2676.6 | **699.8** |
| dsec_city04_e | 794.9 | **130.3** |
| dsec_city11_a | 364.0 | **92.7** |

TABLE V: Translation [%] and rotation [°/m] evaluation results of the proposed method compared to w/o IMU using relative pose RMSE.

| Sequence | w/o IMU | | w/ IMU | |
|---|---|---|---|---|
| | **R** | **t** | **R** | **t** |
| dsec_city04_a | 0.13 | 4.06 | **0.11** | **3.84** |
| dsec_city04_c | 0.06 | 7.71 | **0.05** | **6.47** |

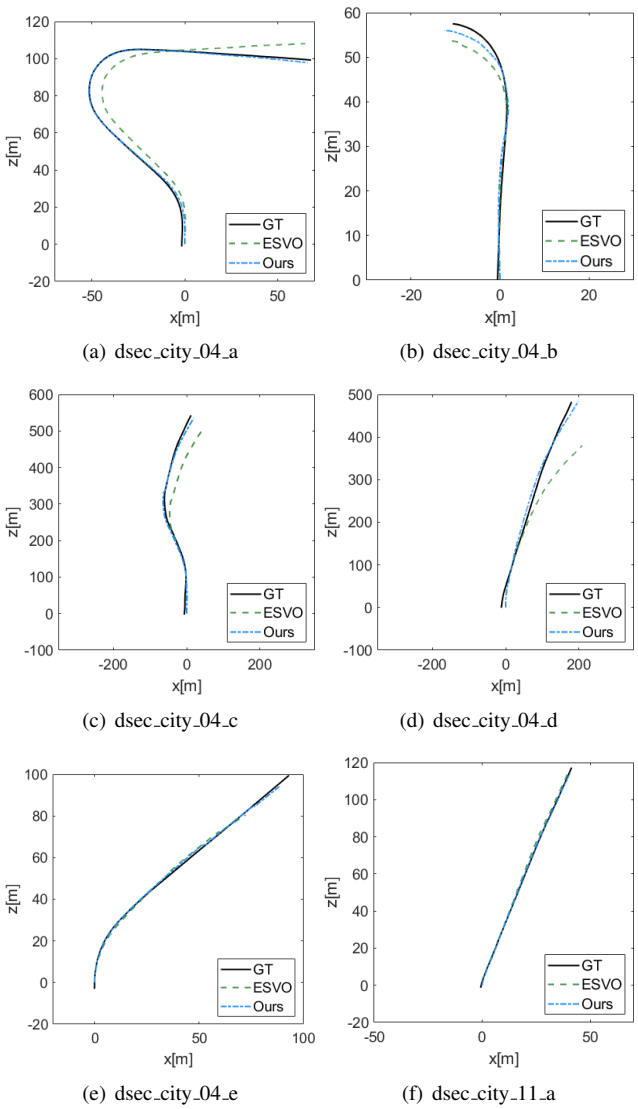
TABLE VI: Computational performance [time:ms]

| Node (#Threads) | Function | ESVO | Ours |
|---|---|---|---|
| Pre-processing (1) | TS | 27 (∼70k) | **3 (∼70k)** |
| | AA | - | 8 (∼70k) |
| Tracking (2) | Non-linear solver | 8 (∼2k) | **7 (∼2k)** |
| Mapping (4) | Event matching | 36 (∼10k) | **18 (∼2.5k)** |
| | Depth optimization | 82 (∼4.5k) | **8 (∼0.9k)** |
| | Depth fusion | 23 (∼140k) | **4 (∼12k)** |
| | Regularization (optional) | 296 (∼25k) | - |
| | Total (w/ optional) | 141 (437) | **30** |

and also report the absolute trajectory error and relative pose error. As shown in Table. III and Table. IV, our new pipeline outperforms ESVO in terms of both ATE and RPE. We also illustrate the resulting trajectories against the groundtruth (GT). As shown in Fig. 7, our trajectories (Ours) are typically more consistent with the GT compared to those of ESVO.

### C. Computational Efficiency

As shown in Table. VI, We compare the computational performance between ESVO and our method using a desktop with Intel Core i7-12700K on the *DSEC*. The reported runtime for each functionality is evaluated on a rough number of points, denoted by (∼). Both systems are accelerated using hyper-threading technology, and the number of threads occupied by each node are declared inside the parentheses. We improve the generation method of TS by reorganizing data structure, making it 9 times faster than that in ESVO. The efficiency of tracking is also slightly improved due to initializing the pose parameters with relative rotation from gyroscope pre-integration. Note that the number of points used for mapping is different between two systems. The minimum number of points required by ESVO's mapping for a normal performance is 10k, and our mapping method requires only 2.5k. In this way, our mapping takes almost one fifth of the time of ESVO and still has remarkably




(a) dsec_city_04_a (b) dsec_city_04_b

(c) dsec_city_04_c (d) dsec_city_04_d

(e) dsec_city_04_e (f) dsec_city_11_a

Fig. 7: Illustration of recovered trajectories.

better tracking performance. Additionally, regularization is very time-consuming and has very little impact on tracking performance, and thus, it is used optionally.

## IV. Conclusion

We present an IMU-aided event-based stereo visual odometry system in this work. Built on top of ESVO, a state-of-the-art event-based visual odometry pipeline, our framework additionally introduces three modules, namely the efficient edge-pixel sampling strategy, the temporal stereo mapping operation, and the usage of inertial measurements. The first two modules lead to more complete and accurate mapping results, and the third module improves the accuracy of camera pose tracking. Besides, our framework scales better with event streaming rate featured by modern event cameras with a high spatial resolution (*e.g.*, $640 \times 480$ pixel), indicating a step forward of direct methods for practical applications.

## REFERENCES


[1] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128×128 120 dB 15 $\mu$s latency asynchronous temporal contrast vision sensor," *IEEE J. Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.

[2] E. Mueggler, B. Huber, and D. Scaramuzza, "Event-based, 6-DOF pose tracking for high-speed maneuvers," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2014, pp. 2761–2768.

[3] X. Lagorce, C. Meyer, S.-H. Ieng, D. Filliat, and R. Benosman, "Asynchronous event-based multikernel algorithm for high-speed visual features tracking," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 8, pp. 1710–1720, Aug. 2015.

[4] A. Z. Zhu, N. Atanasov, and K. Daniilidis, "Event-based feature tracking with probabilistic data association," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2017, pp. 4465–4470.

[5] G. Gallego, J. E. A. Lund, E. Mueggler, H. Rebecq, T. Delbruck, and D. Scaramuzza, "Event-based, 6-DOF camera tracking from photometric depth maps," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 10, pp. 2402–2412, Oct. 2018.

[6] G. Gallego and D. Scaramuzza, "Accurate angular velocity estimation with an event camera," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 632–639, 2017.

[7] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, "Continuous-time visual-inertial odometry for event cameras," *IEEE Trans. Robot.*, 2018.

[8] D. Gehrig, H. Rebecq, G. Gallego, and D. Scaramuzza, "Asynchronous, photometric feature tracking using events and frames," in *Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 766–781.

[9] J. Conradt, M. Cook, R. Berner, P. Lichtsteiner, R. J. Douglas, and T. Delbruck, "A pencil balancing robot using a pair of AER dynamic vision sensors," in *IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2009, pp. 781–784.

[10] T. Delbruck and M. Lang, "Robotic goalie with 3ms reaction time at 4% CPU load using event-based dynamic vision sensor," *Front. Neurosci.*, vol. 7, p. 223, 2013.

[11] H. Rebecq, G. Gallego, and D. Scaramuzza, "EMVS: Event-based multi-view stereo," in *British Mach. Vis. Conf. (BMVC)*, 2016.

[12] H. Kim, S. Leutenegger, and A. J. Davison, "Real-time 3D reconstruction and 6-DoF tracking with an event camera," in *Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 349–364.

[13] 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.*, pp. 1–21, Nov. 2017.

[14] 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. Autom. Lett.*, vol. 2, no. 2, pp. 593–600, 2017.

[15] H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization," in *British Mach. Vis. Conf. (BMVC)*, 2017.

[16] A. Rosinol 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. Autom. Lett.*, vol. 3, no. 2, pp. 994–1001, Apr. 2018.

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

[18] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE, 2007, pp. 225–234.

[19] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, 2015.

[20] V. Vasco, A. Glover, and C. Bartolozzi, "Fast event-based Harris corner detection exploiting the advantages of event-driven cameras," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2016.

[21] E. Mueggler, C. Bartolozzi, and D. Scaramuzza, "Fast event-based corner detection," in *British Mach. Vis. Conf. (BMVC)*, 2017.

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

[23] R. Li, D. Shi, Y. Zhang, K. Li, and R. Li, "FA-Harris: A fast and asynchronous corner detector for event cameras," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2019.

[24] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Fourth Alvey Vision Conf.*, vol. 15, 1988, pp. 147–151.

[25] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 430–443.

[26] I. Alzugaray and M. Chli, "ACE: An efficient asynchronous corner tracker for event cameras," in *3D Vision (3DV)*, 2018, pp. 653–661.

[27] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 6, pp. 580–593, 1997.

[28] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–770, 2004.

[29] S. Li, C. Xu, and M. Xie, "A robust o (n) solution to the perspective-n-point problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1444–1450, 2012.

[30] L. Kneip, H. Li, and Y. Seo, "Upnp: An optimal o (n) solution to the absolute pose problem with universal applicability," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*. Springer, 2014, pp. 127–142.

[31] A. Hadviger, I. Cvišić, I. Marković, S. Vražić, and I. Petrović, "Feature-based event stereo visual odometry," in *2021 European Conference on Mobile Robots (ECMR)*. IEEE, 2021, pp. 1–6.

[32] G. Gallego, H. Rebecq, and D. Scaramuzza, "A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 3867–3876.

[33] A. Z. Zhu, N. Atanasov, and K. Daniilidis, "Event-based visual inertial odometry," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017, pp. 5816–5824.

[34] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2007, pp. 3565–3572.

[35] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial slam using nonlinear optimization," *Proceedings of Robotis Science and Systems (RSS) 2013*, 2013.

[36] M. Gehrig, W. Aarents, D. Gehrig, and D. Scaramuzza, "Dsec: A stereo event camera dataset for driving scenarios," *IEEE Robotics and Automation Letters*, 2021.

[37] M. Liu and T. Delbruck, "Adaptive time-slice block-matching optical flow algorithm for dynamic vision sensors," in *British Mach. Vis. Conf. (BMVC)*, 2018.

[38] G. Gallego, M. Gehrig, and D. Scaramuzza, "Focus is all you need: Loss functions for event-based vision," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 12 272–12 281.

[39] T. Delbruck, "Frame-free dynamic digital vision," in *Proc. Int. Symp. Secure-Life Electron.*, 2008, pp. 21–26.

[40] J. Manderscheid, A. Sironi, N. Bourdis, D. Migliore, and V. Lepetit, "Speed invariant time surface for learning to detect corner points with event-based cameras," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019.

[41] A. Glover, A. Dinale, L. D. S. Rosa, S. Bamford, and C. Bartolozzi, "luvharris: A practical corner detector for event-cameras," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 10 087–10 098, 2021.

[42] J. Engel, J. Stueckler, and D. Cremers, "Large-scale direct SLAM with stereo cameras," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2015.

[43] A. Cayley, "About the algebraic structure of the orthogonal group and the other classical groups in a field of characteristic zero or a prime characteristic," in *Reine Angewandte Mathematik*, 1846.

[44] Y. Zhou, G. Gallego, H. Rebecq, L. Kneip, H. Li, and D. Scaramuzza, "Semi-dense 3D reconstruction with a stereo event camera," in *Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 242–258.

[45] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Oct. 2012.

[46] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7244–7251.