arXiv:2110.12962v3 [cs.CV] 27 Jun 2025

Article

Event Data Association via Robust Model Fitting for Event-based Visual Object Tracking

Haosheng Chen [1]*, Yue Wu [1] and Yidong Peng [1]

[1] The Chongqing Key Laboratory of Image Cognition, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; chenhs@cqupt.edu.cn, wuyue@cqupt.edu.cn and pengyd@cqupt.edu.cn

* Correspondence: chenhs@cqupt.edu.cn

Abstract

Event-based approaches, which are based on bio-inspired asynchronous event cameras, have achieved promising performance on various computer vision tasks. However, the study of the fundamental event data association problem is still in its infancy. In this paper, we propose a novel Event Data Association approach (called EDA) to explicitly address the data association problem. The proposed EDA seeks for event trajectories that best fit the event data, in order to perform unifying data association. In EDA, we first asynchronously gather the event data, based on its information entropy. Then, we introduce a deterministic model hypothesis generation strategy, which effectively generates model hypotheses from the gathered events, to represent the corresponding event trajectories. After that, we present a two-stage weighting algorithm, which robustly weighs and selects true models from the generated model hypotheses, through multi-structural geometric model fitting. Meanwhile, we also propose an adaptive model selection strategy to automatically determine the number of the true models. Finally, we use the selected true models to associate the event data, without being affected by sensor noise and irrelevant structures. We evaluate the performance of the proposed EDA on the remote object tracking task. The experimental results show the effectiveness of EDA under challenging scenarios, such as high speed, motion blur, and high dynamic range conditions.

Keywords: event camera; data association; object tracking; model fitting

1. Introduction

As one of the most delicate neural systems, biological retinas can precisely and efficiently capture object motions [1]. Inspired by the biological retina, asynchronous event cameras, such as DVS [2], DAVIS [3] and ATIS [4], have been proposed to mimic it. These bio-inspired silicon retina vision sensors have a very high dynamic range (higher than 100 dB) and an ultra-low latency (less than 1 ms). In particular, each pixel of the event camera can independently and asynchronously emit binary (i.e., *On* or *Off*) retinal events in responding to pixel intensity changes in the current environment. Pixel intensity changes are usually triggered by object and camera motions, under relatively smooth external illumination variations. Thus, the retinal events contain clear motion information. Moreover, thanks to the high dynamic range and low latency, event cameras are not affected by those challenges that inherently affect conventional frame-based cameras, such as motion blur and pixel saturation. As a result, event cameras are well suited for motion-related computer vision tasks, such as remote visual tracking.

Event-based methods have achieved promising performance on various tasks [5]. However, the study of the fundamental event data association problem is still challenging and in its infancy. Unlike a traditional camera, an event camera only sparsely emits binary (i.e., *On* and *Off*) retinal events at the edges within a scene, in responding to camera and object motions. Therefore, without the pixel intensity information, it is difficult to calculate event correspondences (i.e., event trajectories) in the temporal domain for event-based data association. Since data association is the key to various

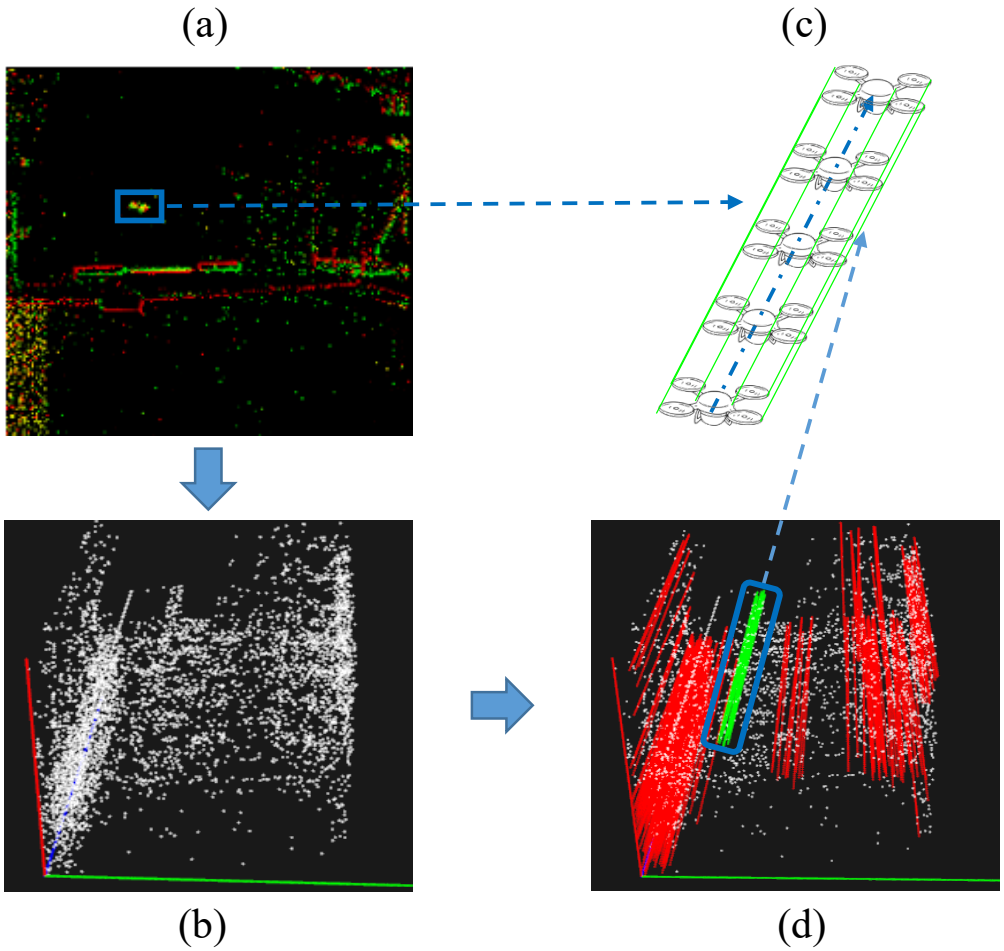


Figure 1. An illustration of event trajectories triggered by a flying drone. (a) The triggered retinal events on the image plane. The retinal events marked by the blue rectangle are triggered by the moving drone. (b) The triggered retinal events in the spatio-temporal domain. (c) The event trajectories triggered by the moving drone. (d) The event trajectories calculated by the proposed EDA in the 3D spatio-temporal domain. The green and red lines in (d) are triggered by the drone motion and the camera motion, respectively.

fundamental tasks (such as object tracking), the problem of event-based data association needs further investigation.

As the pioneering work on event-based data association, [6] proposes a contrast maximization framework to implicitly handle the data association problem based on the corresponding motion parameters. In that framework, the initial motion parameters for each event are empirically set to a range of predefined values for optimization. Since the predefined motion range may not cover all possible event motions, the final result may not be the optimal estimate.

However, [6] and other event-based data association methods [7,8] show that the events triggered by the same edge in the scene can be associated with each other using an event trajectory. Furthermore, they also show that the event trajectories, triggered at different edges of the scene, are nearly 3D lines over a short time interval in the spatio-temporal domain (see an example in Fig. 1). Based on these observed facts, we explicitly formulate the event-based data association problem as a 3D event trajectory estimation problem in the spatio-temporal domain. Then, a novel event-based data association approach (called EDA) is proposed to perform explicit and accurate event-based data association. In the proposed EDA, the event trajectory estimation problem is effectively solved through the theory of robust multi-structural model fitting. To the best of our knowledge, it is the first time to introduce robust model fitting for event-based data association. The estimated multiple structures correspond to the event trajectories triggered by different motions, and they are used to associate the event data. To evaluate the effectiveness of the proposed EDA, we use EDA to deal with the problem

of event-based data association in remote object tracking scenarios. Overall, this paper makes the following contributions:

- We introduce a deterministic model hypothesis generation strategy to effectively generate model hypotheses in the spatio-temporal domain, spanned by retinal events. The generated model hypotheses are considered as event trajectory candidates for event data association.
- We propose a two-stage weighting algorithm, which robustly weighs the generated model hypotheses and selects true event trajectories from them through multi-structural model fitting, without being affected by sensor noise and irrelevant structures.
- We present an adaptive model selection strategy to automatically determine the number of the true event trajectories, based on the elbow value of the corresponding model hypothesis weights calculated by the proposed two-stage weighting algorithm.

We extensively evaluate the proposed EDA on object tracking. The experimental results demonstrate the superiority of EDA over other state-of-the-art event-based tracking methods and several popular conventional tracking methods. In addition, the estimated true event trajectories corresponding to object motions are also provided for visualization.

2. Related Work

As mentioned previously, event cameras have shown their great potential on various computer vision tasks [5]. Different from conventional cameras that are synchronized to the global camera shutter, these bio-inspired vision sensors work in an asynchronous way with ultra low microsecond latencies in responding to object motions, which is inspired by the biological retina.

Thanks to the asynchronous nature, the high spatio-temporal resolution, and the superior HDR properties of event cameras, various event-based methods have been proposed for many computer vision tasks, especially for object tracking. In the field of visual remote tracking, conventional methods can be roughly categorized into three groups: UAV tracking methods[9–12], RGB tracking methods[13–17], hyperspectral tracking methods[18,19], and RGB-T tracking methods[20]. Compared with conventional methods, event-based tracking methods [21–23] show their superiority under fast motions and HDR scenes. The recently proposed event-based tracking methods have achieved state-of-the-art performance by leveraging motion compensation[24], asynchronous tracking-by-detection [25], and deep regression networks [26]. Other notable motion-related event-based tasks include optical flow estimation [27], camera motion estimation [28], and motion segmentation [29]. In addition, there are also some works in relation to event-based feature extraction [30], classification [31], and stereo vision [29]. From these event-based studies, we can see that the bio-inspired event camera can provide an effective and efficient way to solve various challenging vision problems, by exploiting the event data.

However, in the current event-based studies, most methods usually handle the fundamental event-based data association problem in implicit ways, which are designed for their specific tasks. As a result, event-based data association has not been effectively solved by the current event-based works. There are relatively few works focusing on this problem. As one of the pioneer works in this field, [7] proposes a probabilistic data association method based on the event-based features. Also, [6,8] present a unifying contrast maximization framework to implicitly perform data association. The retinal events are usually triggered by multiple motions, and they are greatly affected by a large amount of sensor noise. Due to these difficulties, the aforementioned methods use either indirect or implicit strategies to handle the event-based data association problem. In this paper, we propose a unifying event data association approach (EDA), by which we explicitly handle the event-based data association problem through a robust multi-structural model fitting.

Robust model fitting methods can accurately estimate the model instances (e.g., lines, homograph matrices, or fundamental matrices) in data contaminated with a large number of noise and outliers, and they have been applied in various areas. The most representative work in robust model fitting is the well-known RANSAC algorithm [32]. To improve the fitting performance, many robust fitting methods that use energy functions [33,34], or use preference analysis [35–37], or use hypergraphs

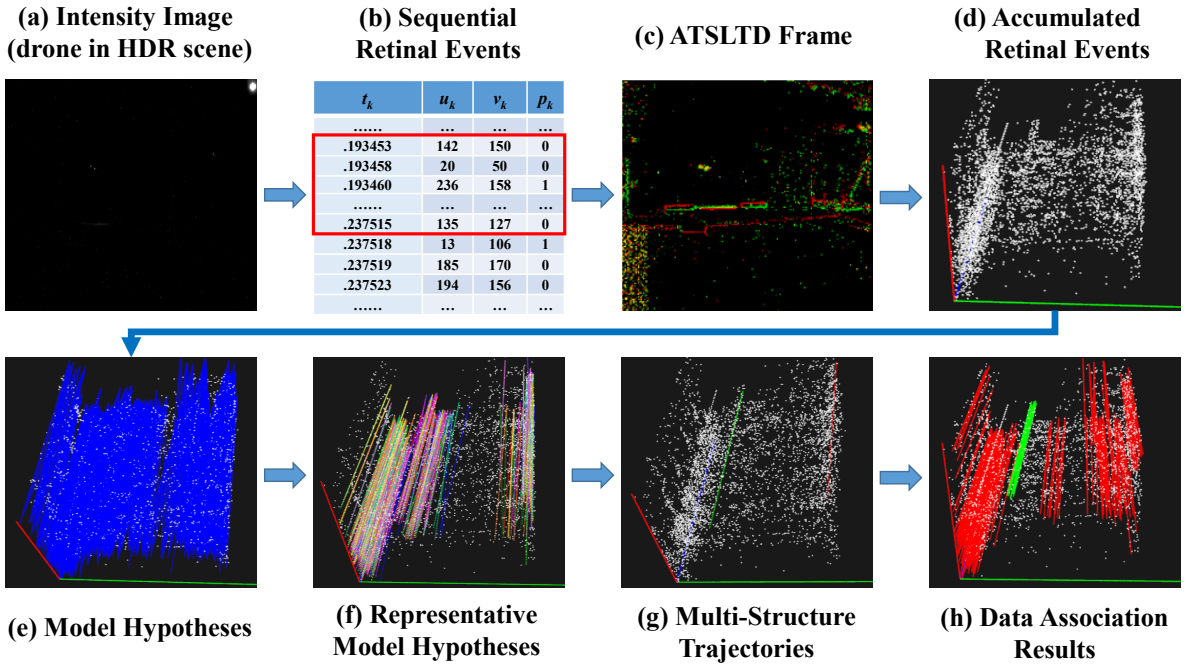


Figure 2. The pipeline of the proposed EDA. Note that the intensity image (a) is not used in EDA, and it is only for reference to show what conventional cameras would see in the low illumination scene. The three lines that are orthogonal to each other in (d)-(h) indicate the vertical (in red), horizontal (in green), and time (in blue) axes, respectively. The other lines in (e)-(h) are the event trajectory models that correspond to the their captions in (e)-(h).

[38,39], or use geometric priors [40–42], have been developed. However, there is no robust model fitting method proposed for event data. In this paper, we are the first to introduce robust model fitting to explicitly target the event association problem for object tracking.

3. Proposed EDA Method

3.1. Overview

In this section, we describe all the components of the proposed EDA approach. The pipeline of the proposed EDA is illustrated in Fig. 2. First, we describe the sequential retinal events, and introduce an asynchronous gathering phase to gather the sequential retinal events, as illustrated in Fig. 2(a)-(d). Then, we present the model hypothesis generation process, to effectively generate model hypotheses, as illustrated in Fig. 2(e)-(f). Finally, we give the details of the model hypothesis selection, to robustly perform the final data association, as illustrated in Fig. 2(g)-(h).

3.2. Grouping Retinal Events

During the recording of event data, each pixel of an event camera asynchronously and independently emits spatio-temporal retinal events, in responding to pixel intensity changes triggered by object motions. The k-th event e_k of the emitted sequential retinal events $\mathcal{E}$ (see Fig. 2(b)) can be represented as follows:

$$e_k \doteq (t_k, u_k, v_k, p_k), \tag{1}$$

where t_k is the timestamp of the event, u_k and v_k are respectively the horizontal and vertical coordinates of the event, and p_k indicates the polarity (i.e., *On* or *Off*). Increasing the pixel intensity will trigger an *On* event, while decreasing the pixel intensity will trigger an *Off* event. Since the changes of pixel intensity usually occur at the edges of objects in a scene due to object and camera motions, the corresponding retinal events are emitted accordingly, as shown in the ATSLTD frame of Fig. 2(c), where the green pixels represent the *On* events and the red pixels represent the *Off* events.

In this work, our goal is to calculate the correct event trajectories to associate the emitted retinal events. Since these event trajectories are approximately 3D lines over short time intervals (or small retinal event displacements), as pointed out by [6], we need to determine the short time intervals over which we gather the sequential retinal events $\mathcal{E}$. Currently, event-based methods can be coarsely categorized into two types, i.e., event-by-event-based and groups-of-events-based, as discussed in [5]. A single event only contains very limited information, thus event-by-event-based methods usually rely on additional appearance information. In contrast, many event-based methods use groups of events as their inputs. However, these groups-of-events-based methods usually set the time interval to a constant value (or use a fixed event number as the threshold), which is not adaptive to different scenes. Moreover, gathering events over a constant time fails to leverage the asynchronous nature of the event data.

To keep the asynchronous nature of event data for benefiting high-level event-based tasks, we propose to leverage the asynchronous event-to-frame strategy in [25] to asynchronously gather the sequential retinal events $\mathcal{E}$ into different sets for data association. According to that strategy, the sequential retinal events are accumulated to form an Adaptive Time-Surface with Linear Time Decay (ATSLTD) frame $\mathcal{F}$. Each of the incoming events will trigger an update at the corresponding pixel coordinates of the ATSLTD frame $\mathcal{F}$, thereby increasing the amount of information contained in the ATSLTD frame $\mathcal{F}$. Then, the amount of information in the ATSLTD frame is measured by the image entropy of $\mathcal{F}$. Here, the image entropy of $\mathcal{F}$ is quantitatively calculated by a Non-Zero Grid Entropy (NZGE) metric. After that, a set of image entropy values is collected from those ATSLTD frames that contain clear and sharp object contours. Here the clear and sharp object contours indicate relatively small event displacements, which keep the event trajectories straight. Finally, a confidence interval $[\alpha, \beta]$ of the collected image entropy values is calculated by using the Student's t-distribution. During the recording, the emitted sequential retinal events $\mathcal{E}$ will continuously update the ATSLTD frame $\mathcal{F}$. If the image entropy value of the ATSLTD frame $\mathcal{F}$ falls into the interval $[\alpha, \beta]$, the historical retinal events are accumulated for further processing.

Suppose that the image entropy value of the accumulated retinal events collected from time t_k falls into the calculated confidence interval $[\alpha, \beta]$ at time $t_k + T$. The corresponding accumulated retinal events are $\mathcal{E}_{t_k, t_k+T}$, and the next time interval starts from time $t_k + T$. By repeating this process, we can asynchronously gather the incoming retinal events into different sets for continuous event trajectory estimation and event-based data association.

3.3. Generating Model Hypotheses

After the grouping process, we have obtained a set of accumulated retinal events $\mathcal{E}_{t_k, t_k+T}$, which is collected during the time interval $[t_k, t_k + T]$. The accumulated retinal events $\mathcal{E}_{k,k+T}$ can be represented in a 3D spatio-temporal space, as shown in Fig. 2(d). In Fig. 2(d), the orthogonal red, green, and blue lines are respectively the vertical, horizontal, and time axises, and each of the white voxels indicates a retinal event. In the 3D spatio-temporal space, each of the accumulated retinal events $\mathcal{E}_{k,k+T}$ is a 3D event voxel, regardless of its polarity. The k-th event $e_k = (t_k, u_k, v_k, p_k)$ of $\mathcal{E}_{k,k+T}$ is represented by the k-th 3D event voxel $\mathbf{e}_k = (u_k, v_k, t_k)$.

As mentioned previously, the expected event trajectories (i.e., the models), which best fit the event data, are 3D lines. Therefore, we propose an effective deterministic model hypothesis generation strategy to generate a set of 3D line hypotheses $\mathcal{L}_{t_k, t_k+T}$ from the accumulated retinal events $\mathcal{E}_{t_k, t_k+T}$ (some examples are shown in the blue lines in Fig. 2(e)). Each generated model hypothesis can be mathematically defined by its start event voxel and its end event voxel. For instance, the k-th hypothesis $\mathbf{l}_k$ is represented by its start event voxel $\mathbf{e}_s^k = (u_s^k, v_s^k, t_s^k)$ and its end event voxel $\mathbf{e}_e^k = (u_e^k, v_e^k, t_e^k)$ as follows:

$$\mathbf{l}_k = \begin{bmatrix} u_s^k + \left(u_e^k - u_s^k\right)\lambda \\ v_s^k + \left(v_e^k - v_s^k\right)\lambda \\ t_s^k + \left(t_e^k - t_s^k\right)\lambda \end{bmatrix}, \forall \lambda \in [0, 1]. \tag{2}$$

During the model hypothesis generation process, we equally divide the accumulated retinal events $\mathcal{E}_{t_k,t_k+T}$ into N_s time slices. Then, each of the generated model hypotheses is sampled using an event voxel pair, which includes an event voxel in the first time slice and an event voxel in the last time slice. As a result, different from traditional model hypothesis generation strategies (which usually randomly sample data to generate model hypotheses), the proposed model hypothesis generation strategy performs deterministic model hypothesis generation, based on guidance by the temporal information in the event data.

As illustrated in Fig. 1, the event trajectories are usually triggered by the camera and object motions in the scene. Moreover, Fig. 1 also reveals the fact that the event trajectories, triggered by the same motion, are nearly parallel with each other, while the event trajectories triggered by different motions have relatively significant angle differences with each other. Therefore, we can refine the generated model hypotheses $\mathcal{L}_{t_k,t_k+T}$ to produce a few representative model hypotheses $\hat{\mathcal{L}}_{t_k,t_k+T}$, as shown in Fig. 2(f), to relieve the computational burden. In Fig. 2(f), the representative model hypotheses triggered by different motions are marked in different colors. During the refining process, we calculate the cosine distances between two hypotheses in $\mathcal{L}_{t_k,t_k+T}$. The cosine distance between two hypotheses $\mathbf{l}_i$ and $\mathbf{l}_j$ is calculated as follows:

$$cosine_dis(\mathbf{l}_i, \mathbf{l}_j) = 1 - \frac{\mathbf{l}_i \cdot \mathbf{l}_j}{\|\mathbf{l}_i\| \|\mathbf{l}_j\|}. \tag{3}$$

If the cosine distances among a set of hypotheses are close to zero, they are considered to be parallel with each other. Then, for each parallel hypothesis set, the model hypothesis that has the largest number of parallel hypotheses is selected as the representative hypothesis for the set, and the other hypotheses in the set are removed. After that, each representative model hypothesis is a unique event trajectory, and it is not parallel with the other representative hypotheses.

3.4. Selecting Model Instances

After the previous two steps, we have a set of representative model hypotheses $\hat{\mathcal{L}}_{t_k,t_k+T}$. However, in most cases, since the event data is severely contaminated by sensor noise, the majority of the model hypotheses in $\hat{\mathcal{L}}_{t_k,t_k+T}$ are bad model hypotheses. Here, we propose an effective two-stage weighting algorithm (called as TSW) to select the correct model hypotheses triggered by the multi-structural motions in the scene. Meanwhile, the TSW algorithm also removes the bad model hypotheses generated by the sensor noise.

During the first stage of the proposed TSW algorithm, we first calculate the residuals $\mathcal{R}_{t_k,t_k+T}$ between the accumulated retinal events $\mathcal{E}_{t_k,t_k+T}$ and the representative model hypotheses $\hat{\mathcal{L}}_{t_k,t_k+T}$ to select inliers $\mathcal{I}_{t_k,t_k+T}$ from $\mathcal{E}_{t_k,t_k+T}$. The residual $\mathcal{R}^{i,j}_{t_k,t_k+T}$ between the i-th retinal event $\mathbf{e}_i$ in $\mathcal{E}_{t_k,t_k+T}$ and the j-th model hypothesis $\mathbf{l}_j$ in $\hat{\mathcal{L}}_{t_k,t_k+T}$ is calculated by measuring the 3D voxel-to-line distance $d_{i,j}$ between $\mathbf{e}_i$ and $\mathbf{l}_j$, as follows:

$$\mathcal{R}^{i,j}_{t_k,t_k+T} = d_{i,j} = \frac{\left|\left(\mathbf{e}_i - \mathbf{l}^s_j\right) \times \left(\mathbf{e}_i - \mathbf{l}^e_j\right)\right|}{\left|\mathbf{l}^e_j - \mathbf{l}^s_j\right|}, \tag{4}$$

where $\mathbf{l}^s_j = (u^s_j, v^s_j, t^s_j)$ and $\mathbf{l}^e_j = (u^e_j, v^e_j, t^e_j)$ are respectively the start event voxel of $\mathbf{l}_j$ and the end event voxel of $\mathbf{l}_j$. After the calculation, each representative model hypothesis has a vector of residuals, which is calculated with each retinal event in $\mathcal{E}_{t_k,t_k+T}$. To alleviate the problem of scale difference in the spatio-temporal space, the elements in the residual vector are normalized by the norm of the vector.

Then, based on the residual $\mathcal{R}_{t_k,t_k+T}$, for each of the model hypotheses in $\hat{\mathcal{L}}_{t_k,t_k+T}$, the accumulated retinal events $\mathcal{E}_{t_k,t_k+T}$ can be divided into two sets: i.e., inliers $\mathcal{I}_{t_k,t_k+T}$ and outliers $\mathcal{O}_{t_k,t_k+T}$. For a reference model hypothesis in $\hat{\mathcal{L}}_{t_k,t_k+T}$, if the residual (i.e., the 3D voxel-to-line distance) for a retinal event is less than an inlier noise scale τ (which can be estimated by using a robust scale estimator, such

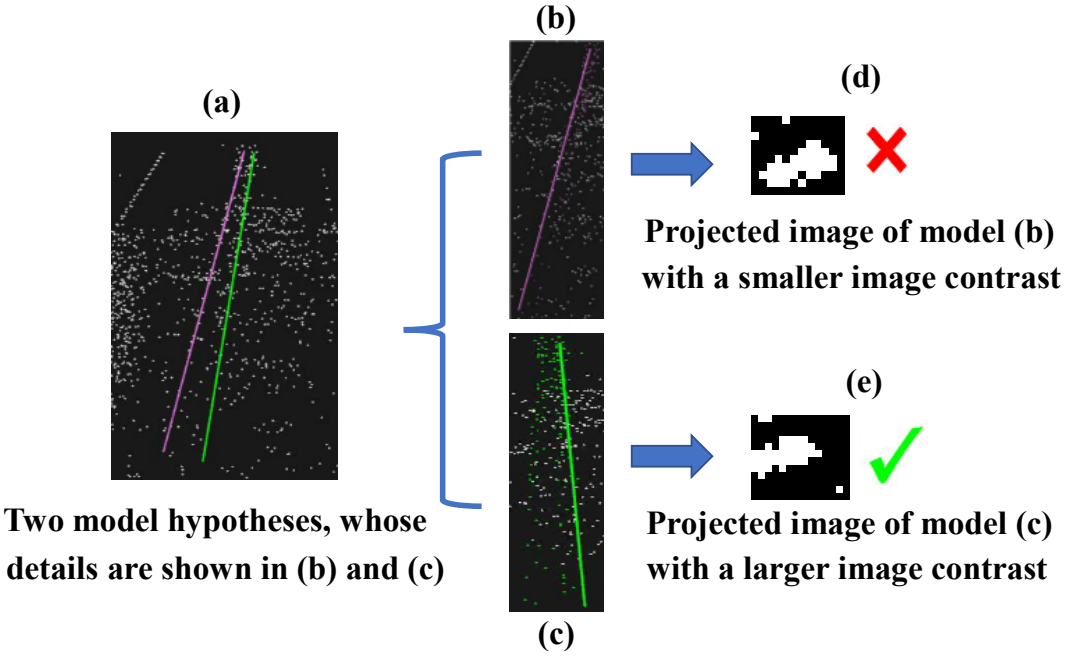


Figure 3. An illustration of the second stage of the proposed TSW algorithm. For the two model hypotheses in (a), compared with the projected image (d) from the sub-optimal model (b), the projected image (e) from the optimal model (c) has a larger image contrast. Thus, the model (c) is more likely to be a correct model hypothesis.

as IKOSE [43]), the corresponding retinal event is considered as an inlier. Otherwise, the corresponding retinal event is considered as an outlier. If the inlier number of the reference model hypothesis is no more than two (i.e., only including the start voxel and the end voxel), this model hypothesis will be considered as noise and removed. After that, the reference model hypothesis will be weighted by measuring the dispersion of its inliers $\mathcal{I}_{t_k,t_k+T}$ on the time axis, which can be written as follows:

$$w = \frac{1}{N_i}\sum_{i=1}^{N_i}(t_i - S_t/2)^2, \tag{5}$$

where N_i is the number of inliers in $\mathcal{I}_{t_k,t_k+T}$, t_i is the timestamp of the i-th inlier in $\mathcal{I}_{t_k,t_k+T}$, and S_t is the length of the time axis. The first weighting stage in Eq. 5 is based on the fact that the inliers of a true model hypothesis are supposed to be normally distributed along the time axis.

After the first weighting stage, the second weighting stage is performed to reduce the influence of the pseudo-outliers (which are the inliers for a model hypothesis, but they are the outliers for the other model hypotheses). The second stage of the proposed TSW algorithm is based on the fact that the warped event image, projected by the true model hypothesis, has a larger image contrast compared with the false model hypotheses generated by the pseudo-outliers. An illustration of the second stage of the TSW algorithm is shown in Fig. 3. In the second stage, for the reference model hypothesis, we project its 3D spatio-temporal inliers in $\mathcal{I}_{t_k,t_k+T}$ to the reference 2D image plane according to the model hypothesis. After that, we crop the projected image to get a warped event image $\mathcal{P}_{t_k,t_k+T}$ using a bounding box, which is the minimum enclosing rectangle of the projected inliers in the 2D image plane. Moreover, we normalize the warped event image $\mathcal{P}_{t_k,t_k+T}$ and then calculate the image contrast of the normalized event image $\hat{\mathcal{P}}_{t_k,t_k+T}$ to adjust the weight w of the reference model hypothesis, as follows:

$$\hat{w} = w * (1 - \frac{1}{N_{\hat{\mathcal{P}}}}\sum_{i,j}(\hat{\mathcal{P}}^{i,j}_{t_k,t_k+T} - \mu_{\hat{\mathcal{P}}})^2), \tag{6}$$

where $N_{\hat{\mathcal{P}}}$ is the number of pixels of $\hat{\mathcal{P}}_{t_k,t_k+T}$, $\hat{\mathcal{P}}^{i,j}_{t_k,t_k+T}$ is the number of the projected inliers in the i-th row and the j-th column of $\hat{\mathcal{P}}_{t_k,t_k+T}$, and $\mu_{\hat{\mathcal{P}}}$ is the mean of $\hat{\mathcal{P}}_{t_k,t_k+T}$. After the two-stage weighting process, we have N_m remaining model hypotheses together with their weights $\hat{\mathcal{W}} = \{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{N_m}\}$.

Algorithm 1 The Proposed Event Data Association.

Input: The events $\mathcal{E} = \{e_1, e_2, \cdots, e_n\}$ collected from the start time t_k.

1: Initialize an ATSLTD frame $\mathcal{F}$ to a two-channel all-zero image with the resolution of the corresponding event camera;
2: Update $\mathcal{F}$ with the emitted events in $\mathcal{E}$ and accumulate the emitted events using the asynchronous event-to-frame strategy in Sec. 3.2;
3: Calculate the image entropy of $\mathcal{F}$ using the quantitative NZGE measure in the asynchronous event-to-frame strategy;
4: Generate the confidence interval $[\alpha, \beta]$ for the calculated image entropy based on the Student's t-distribution or adopt pre-calculated intervals;
5: Group the current accumulated events between t_k and $t_k + T$ into a set of events $\mathcal{E}_{t_k,t_k+T}$ when the image entropy of $\mathcal{F}$ reach $[\alpha, \beta]$ at time $t_k + T$;
6: Generate event trajectory model hypotheses $\mathcal{L}_{t_k,t_k+T}$ from the event set $\mathcal{E}_{t_k,t_k+T}$ using Eq. (2);
7: Select the representative model hypotheses $\hat{\mathcal{L}}_{t_k,t_k+T}$ from the generated event trajectory model hypotheses $\mathcal{L}_{t_k,t_k+T}$ using Eq. (3);
8: Weigh the representative model hypotheses $\hat{\mathcal{L}}_{t_k,t_k+T}$ based on the residuals $\mathcal{R}_{t_k,t_k+T}$ between $\mathcal{E}_{t_k,t_k+T}$ and $\hat{\mathcal{L}}_{t_k,t_k+T}$ using Eq. (4);
9: Calculate an inlier noise scale τ for $\mathcal{E}_{t_k,t_k+T}$ using a robust scale estimator or a user-specified value;
10: Select inliers $\mathcal{I}_{t_k,t_k+T}$ from $\mathcal{E}_{t_k,t_k+T}$ according to the inlier noise scale τ;
11: Weigh the representative model hypotheses $\hat{\mathcal{L}}_{t_k,t_k+T}$ by measuring the dispersion of the inliers $\mathcal{I}_{t_k,t_k+T}$ on the time axis using Eq. (5);
12: Weigh the representative model hypotheses $\hat{\mathcal{L}}_{t_k,t_k+T}$ based on the weights in the previous step and the contrast of warped event images using Eq. (6);
13: Estimate the number of true model hypotheses N_s in the scene using the elbow weight of $\hat{\mathcal{L}}_{t_k,t_k+T}$, and select true model hypotheses according to N_s;

Output: The event trajectories associated by the true model instances.

The number of the true model instances N_s in a scene (i.e., the number of the true motions) is usually unknown. Therefore, we propose an adaptive model selection strategy to automatically estimate it. The proposed strategy is based on the observation that the dispersion of weights between a true and a false model hypotheses is usually large, while the dispersion between a pair of the true model hypotheses is relatively small. Therefore, the elbow model weight that has the largest difference of weights among its neighbors, is critical for the estimation. In the proposed strategy, we sort all the remaining model hypotheses in an ascending order according to their weights $\hat{\mathcal{W}}$. Then, we calculate the difference of weights between every two adjacent sorted model hypotheses. The number of the true model structures N_s is set to the first k, where the k-th difference value d_k achieves the elbow value (i.e., the biggest difference value) among its four neighbors (i.e., d_{k-2}, d_{k-1}, d_{k+1}, and d_{k+2}).

Finally, the model hypotheses having the N_s smallest weights are selected as the estimated model instances, as shown in Fig. 2(g), where the green and red lines are the estimated model instances. For each estimated model instance, we calculate its parallel model hypotheses in $\mathcal{L}_{t_k,t_k+T}$, to perform the final data association. As a result, the inliers are associated with each other by their model structures, as shown in Fig. 2(h). Overall, the step-by-step description of the proposed EDA is given in Algorithm 1.

4. Experiments

As a fundamental task in computer vision, object tracking is still affected by high speed and high dynamic range problems. Therefore, in this work, we evaluate the performance of the proposed EDA on the task of object tracking.

4.1. Parameter Settings

For the parameters used in the proposed EDA, we adopt the same confidence interval $[\alpha, \beta]$ as that of ETD [25] for fair competition. The number of the time slices N_s is empirically set to 10 to

achieve the trade-off between accuracy and efficiency. The inlier noise scale τ is fixed to 0.01 for all the experiments.

4.2. Evaluation Protocol

For the object tracking task, since the event data is associated between two adjacent frames, we employ an evaluation protocol of frame-wise tracking to evaluate the performance of the proposed EDA approach. For the frame-wise tracking, the evaluation of these competing methods is based on object pairs, each of which includes two object regions on two adjacent frames corresponding to the same object. During the evaluation, we treat each of the object pairs as a tracking instance in the corresponding frame.

The test object tracking sequences include the *shapes_translation*, *shapes_6dof*, *poster_6dof*, and *slider_depth* sequences from the popular Event Camera Dataset (ECD) [44] and the *fast_drone*, *light_variations*, *what_is_background*, and *occlusions* sequences from the challenging Extreme Event Dataset (EED) [24]. In the four sequences from the ECD dataset, the first three sequences have an increasing motion speed. The fourth sequence has a constant motion speed. For the four sequences, the corresponding object textures vary from simple Black&White shapes to complicated artifacts. By using these four sequences, we are mainly concerned with the performance of all competing methods for a variety of object motions, especially for fast motion, and for different object shapes and textures. For the four sequences from the EED dataset, the first three sequences record a fast moving drone under low illumination environments. The fourth sequence records a moving ball with a net as the foreground. By using these four sequences, we evaluate the performance of the competing trackers in low illumination conditions and occlusion situations.

For the evaluation metrics, we follow the frame-wise tracking protocol in [26] and use the Average Overlap Rate (AOR) and the Average Robustness (AR) to respectively measure the precision and the robustness of all the competing object tracking methods, as follows:

$$AOR = \frac{1}{N_{rep}} \frac{1}{N_{pair}} \sum_{i=1}^{N_{rep}} \sum_{j=1}^{N_{pair}} \frac{O_{i,j}^{E} \cap O_{i,j}^{G}}{O_{i,j}^{E} \cup O_{i,j}^{G}}, \tag{7}$$

$$AR = \frac{1}{N_{rep}} \frac{1}{N_{pair}} \sum_{i=1}^{N_{rep}} \sum_{j=1}^{N_{pair}} success_{i,j}, \tag{8}$$

where $O_{i,j}^{E}$ is the estimated bounding box in the i-th round of the evaluation for the j-th object pair. $O_{i,j}^{G}$ is the corresponding ground truth. N_{rep} is the number of times that we repeat the evaluation. N_{pair} is the number of object pairs in the current sequence. We set the value of N_{rep} to 5 for all the following experiments. $success_{i,j}$ indicates that whether the tracking in the i-th round for the j-th pair is successful or not (0 means failure and 1 means success). If the AOR score obtained by a tracker for one object pair is under 0.5, we consider it as a tracking failure case.

4.3. Event-Based Object Tracking

In this work, we compare the proposed EDA with eight popular tracking methods, including SiamFC [45], ECO [46], SiamRPN++ [47], ATOM [48], E-MS [21], ETD [25], RMRNet [26], and an event-based variants of ECO (called as ECO-E). Among these trackers, SiamFC, ECO, SiamRPN++, and ATOM are the popular conventional tracking methods. E-MS, ETD, and RMRNet are the state-of-the-art event-based tracking methods. ECO-E is an event-based variant of ECO, which uses TSLTD frames as its inputs. Thus, ECO-E is adopted to evaluate the performance of ECO on the event data. Moreover, E-MS is extended to support the bounding box-based object tracking.

The quantitative results are given in Table 1, Table 2, Table 3, and Table 4. We also provide some qualitative results obtained by SiamFC, ECO, SiamRPN++, ATOM, E-MS, ETD, RMRNet, and our EDA in Fig. 4. From the tables and the figure, we can see that our EDA achieves the best performance on most of the sequences except for the *poster_6dof* and *occlusions* sequences, on which our EDA has

Table 1. The AOR results obtained by the eight competitors and our EDA on the ECD dataset. The best results are highlighted in **bold**.

Method	*shapes_translation*	*shapes_6dof*	*poster_6dof*	*slider_depth*
SiamFC [45]	0.812	0.835	0.830	0.909
ECO [46]	0.823	0.847	0.846	0.947
SiamRPN++ [47]	0.790	0.779	0.753	0.907
ATOM [48]	0.815	0.803	0.835	0.897
ECO-E [46]	0.821	0.834	0.783	0.771
E-MS [21]	0.675	0.612	0.417	0.447
ETD [25]	0.843	0.852	0.824	0.903
RMRNet [26]	0.836	**0.866**	0.859	0.915
EDA(Ours)	**0.896**	**0.866**	**0.872**	**0.969**

Table 2. The AR results obtained by the eight competitors and our EDA on the ECD dataset. The best results are highlighted in **bold**.

Method	*shapes_translation*	*shapes_6dof*	*poster_6dof*	*slider_depth*
SiamFC [45]	0.940	0.968	0.956	**1.000**
ECO [46]	0.943	0.969	0.960	**1.000**
SiamRPN++ [47]	0.942	0.972	0.899	**1.000**
ATOM [48]	0.945	0.974	0.961	**1.000**
ECO-E [46]	0.941	0.960	0.878	0.993
E-MS [21]	0.768	0.668	0.373	0.350
ETD [25]	**0.998**	**0.998**	**0.995**	**1.000**
RMRNet [26]	0.951	0.980	0.962	**1.000**
EDA(Ours)	**0.998**	**0.998**	0.983	**1.000**

Table 3. The AOR results obtained by the eight competitors and our EDA on the EED dataset. The best results are highlighted in **bold**.

Method	*fast_drone*	*light_variations*	*what_is_background*	*occlusions*
SiamFC [45]	0.766	0.772	0.712	0.148
ECO [46]	0.830	0.782	0.675	0.209
SiamRPN++ [47]	0.717	0.497	0.653	0.096
ATOM [48]	0.763	0.652	0.725	0.387
ECO-E [46]	0.728	0.685	0.099	0.308
E-MS [21]	0.313	0.325	0.362	0.356
ETD [25]	0.817	0.859	0.653	0.608
RMRNet [26]	0.892	0.802	0.202	**0.716**
EDA(Ours)	**0.913**	**0.894**	**0.752**	0.702

obtained the second best AR and AOR, respectively. This is because that, these two sequences (i.e., *poster_6dof* and *occlusions*) include considerable object pairs that are almost out-of-view or highly occluded. These object pairs slightly affect the robustness of EDA compared with the edge-based ETD tracker and the motion estimation-based RMRNet tracker. The superior performance achieved by EDA is largely attributed to the proposed two-stage weighting algorithm, which accurately selects true event trajectories without being affected by the sensor noise.

As for the competitors, SiamFC, ECO, SiamRPN++, and ATOM also achieve competitive results on some sequences of the ECD dataset. However, they show inferior tracking performance on the sequences with fast motion and low illumination conditions (such as the *light_variations* and *occlusions* sequences). They usually lose the tracked objects, due to the influence of motion blur and HDR conditions. In contrast, the three event-based competitors (i.e., ECO-E, E-MS, and ETD)

Table 4. The AR results obtained by the eight competitors and our EDA on the EED dataset. The best results are highlighted in **bold**.

Method	*fast_drone*	*light_variations*	*what_is_background*	*occlusions*
SiamFC [45]	**1.000**	0.947	0.833	0.000
ECO [46]	**1.000**	0.934	0.750	0.333
SiamRPN++ [47]	0.941	0.500	0.833	0.167
ATOM [48]	**1.000**	0.921	**0.917**	0.500
ECO-E [46]	0.882	0.803	0.000	0.333
E-MS [21]	0.307	0.321	0.360	0.353
ETD [25]	0.941	0.934	0.807	0.647
RMRNet [26]	**1.000**	0.947	0.083	**0.833**
EDA(Ours)	**1.000**	**0.966**	**0.917**	**0.833**

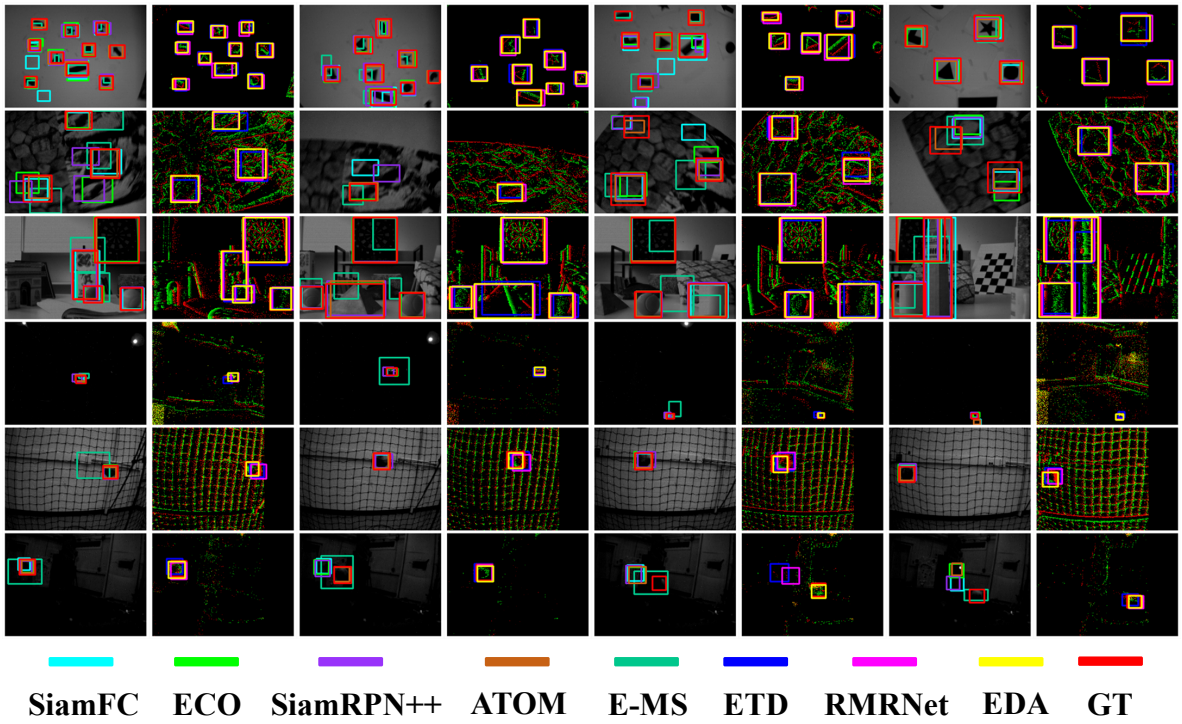


Figure 4. Quantitative tracking results obtained by SiamFC, ECO, SiamRPN++, ATOM, E-MS, ETD, RMRNet, and the proposed EDA. Each row represents a sequence. From top to bottom, the corresponding sequences are *shape_6dof*, *poster_6dof*, *slider_depth*, *light_variations*, *what_is_background*, and *occlusions*, respectively. From left to right, the 2nd, 4th, 6th, and 8th columns show the results obtained by ETD, RMRNet, and EDA with the ground truth (abbreviated as GT). The 1st, 3rd, 5th, and 7th columns show the results obtained by the other competing methods with the GT. Best viewed in color.

can effectively handle fast motions and HDR scenes, by leveraging the good properties of the event data. Nevertheless, E-MS is less effective in handling the challenges of complicated object textures and cluttered backgrounds (e.g., on the *poster_6dof* and *slider_depth* sequences), and it is sensitive to sensor noise. Compared with E-MS, the ETD and RMRNet trackers achieve very competitive results on most of the sequences. However, due to the unstable performance during the processes of the edge-based object proposal generation and motion estimation, ETD and RMRNet achieve inferior precision performance compared with the proposed EDA. Moreover, compared with ECO, ECO-E has achieved much inferior performance, which indicates that directly applying conventional tracking methods to event data is less effective.

We also provide the event trajectory results obtained by the proposed EDA on some representative sequences from the ECD and EED datasets. Furthermore, in order to evaluate the performance of EDA

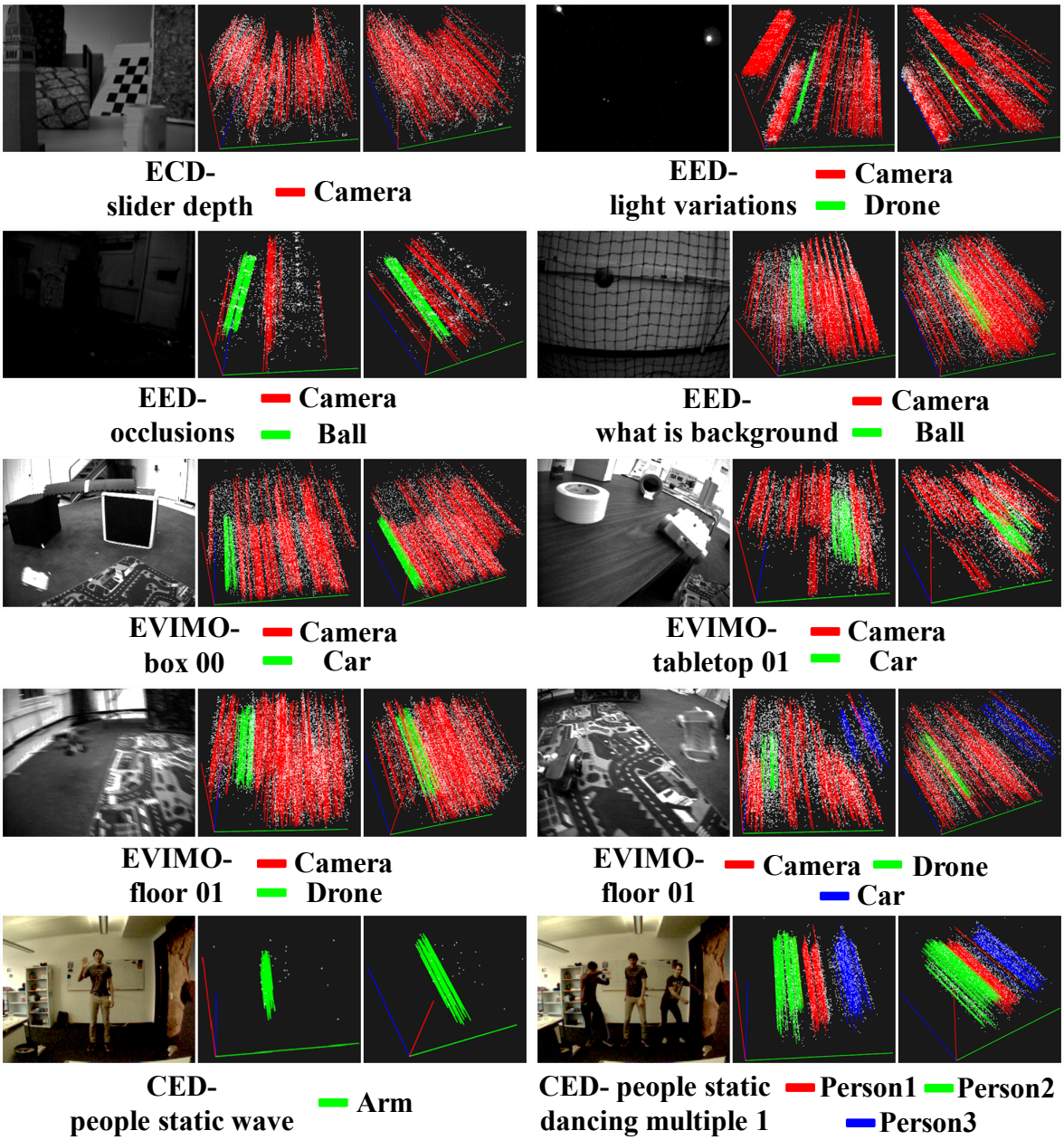
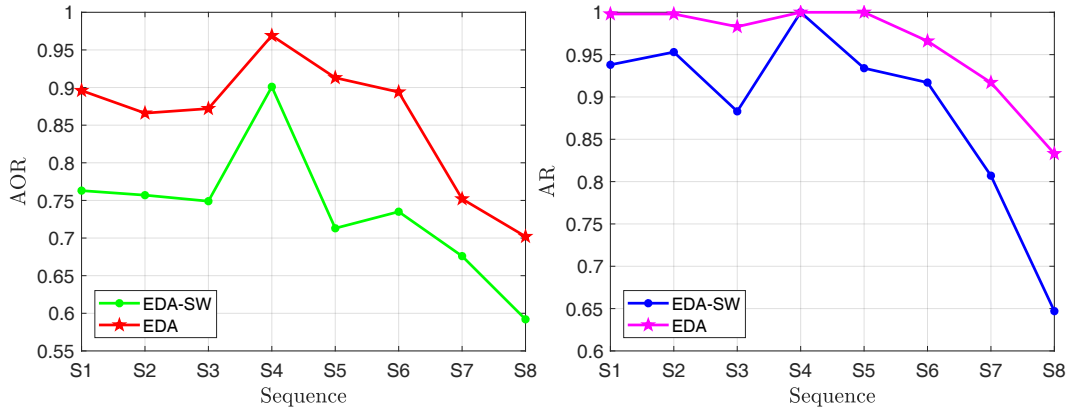


Figure 5. Event trajectory estimation results obtained by the proposed EDA approach. The event trajectories marked by different colors are triggered by different motions. The names and the corresponding motions of the video sequences are listed in the bottom of the corresponding event trajectory results, respectively. For each sequence, we show the intensity image of the scene (only for reference, not used in the proposed EDA) and two views of its event trajectory results. Best viewed in color.

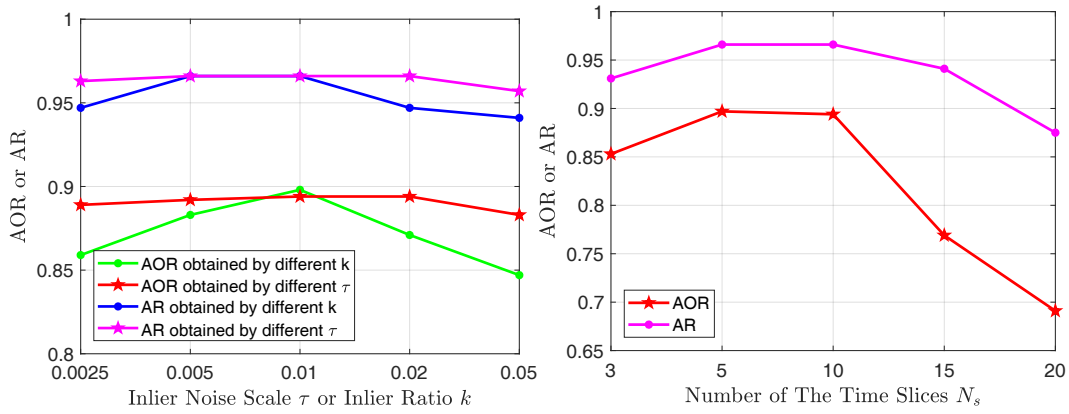
under more tracking scenarios with challenges, we test it on some representative sequences from the Color Event Dataset (CED) [49] and the recently released EVIMO dataset [50]. Here, it should be noted that there is no ground truth that is publicly available for the CED and EVIMO datasets for the task of object tracking. Therefore, we only provide the event trajectory results obtained by EDA for qualitative evaluation. The qualitative results are shown in Fig. 5.

From the results in Fig. 5, we can see that the proposed EDA has accurately estimated most of the true event trajectories triggered by the camera and object motions through robust model fitting. In particular, it is worth noting that, with the help of the proposed adaptive model selection strategy, the numbers of the true models for all the video sequences in Fig. 5 are correctly estimated by the proposed EDA. Moreover, EDA works well when dealing with various challenging situations including HDR



(**a**) (**b**)

Figure 6. The AOR and AR results obtained by the proposed EDA and its variant EDA-SW on the eight sequences in the ECD and EED datasets. The corresponding eight sequences S1 to S8 are the sequences from *shapes_translation* to *occlusions*.



(**a**) (**b**)

Figure 7. Parameter analysis on the inlier noise scale τ, the inlier ratio k in IKOSE, and the number of the time slices N_s. The AOR and AR results obtained by the proposed EDA with different values of the inlier noise scale τ and different values of the inlier ratio k are shown in the left image. The AOR and AR results obtained by the proposed EDA with different numbers of the time slices N_s are shown in the right image.

scenes (i.e., the EED-light_variations and EED-occlusions sequences), fast motions (i.e., all the three sequences from the EED dataset and the EVIMO-floor_01 sequence), and multi-structural motions (i.e., the EVIMO-floor_01 and CED-people_static_dancing_multiple_1 sequences).

5. Discussion

5.1. Ablation Study

To evaluate the proposed EDA on visual tracking, we need to calculate the corresponding object bounding box based on the event trajectories associated by EDA. According to the frame-wise tracking protocol, for each tracking instance, we have the ground truth bounding box of the tracked object at the current frame. EDA needs to estimate the corresponding object bounding box of the tracked object at the next frame. To achieve this goal, we project those retinal events within the current bounding box to the reference image plane at the timestamp of the next frame using the estimated event trajectories calculated by EDA. Then, we use the minimum enclosing rectangle of the projected events in the reference plane as the estimated bounding box at the current frame.

It is worth pointing out that the proposed EDA is the first that introduces robust model fitting to solve the event data association problem for object tracking. Since EDA does not rely on any existing

event data association methods or object tracking methods, there is no baseline method for it. In EDA, only the second stage of the TSW algorithm can be removed for ablation study, and we term the new variant of EDA as EDA-SW. The other components in the proposed EDA are necessary parts, and they do not have appropriate alternatives for an ablation study.

The AOR and AR results obtained by the proposed EDA and its variant EDA-SW on the eight sequences in ECD and EED are shown in Fig. 6. EDA-SW respectively achieves AOR/AR scores of 0.763/0.938, 0.757/0.953, 0.749/0.883, 0.901/1.000, 0.713/0.934, 0.735/0.917, 0.676/0.807, and 0.592/0.647 on the eight sequences (i.e., from *shapes_translation* to *occlusions*). In contrast, EDA respectively obtains 0.896/0.998, 0.866/0.998, 0.872/0.983, 0.969/1.000, 0.913/1.000, 0.894/0.966, 0.752/0.917, and 0.702/0.833 on the eight sequences. From the results, we can see that the second stage of the TSW algorithm contributes significantly on improving the performance. By using the second stage of the TSW algorithm, the majority of pseudo-outliers are removed. This effectively reduces the ambiguity in the model selection process.

5.2. Parameter Analysis

In addition, we perform a parameter analysis on the challenging *light_ variations* sequence. Firstly, we study the influence of different inlier noise scale τ. The proposed EDA respectively achieves AOR/AR scores of 0.889/0.963, 0.892/0.966, 0.894/0.966, 0.894/0.966, and 0.883/0.957 when τ is set to 0.0025, 0.005, 0.01, 0.02, and 0.05. From the results, we can see that the performance of EDA is not very sensitive to the value of τ, and a small value of τ (around 0.01) can work well even in challenging scenes. Then, we replace τ with the IKOSE estimator [43]. Since IKOSE needs the inlier ratio k, we study the influence of different k. EDA respectively obtains AOR/AR scores of 0.859/0.947, 0.883/0.966, 0.898/0.966, 0.871/0.947, and 0.847/0.941 when k is set to 0.0025, 0.005, 0.01, 0.02, and 0.05. The comparison between the EDA with a fixed τ and the EDA with IKOSE are shown in the left image of Fig. 7. As shown in the figure, the EDA with IKOSE only achieves a small improvement when k is set to 0.01, and its AOR/AR scores drops considerably when k is set to the other values compared with the EDA with a fixed τ. Therefore, the proposed EDA adopts the fixed noise scale τ to avoid the trivial parameter tuning. Finally, we study the influence of different numbers of the time slices N_s on the performance of EDA. EDA respectively achieves AOR/AR scores of 0.853/0.931, 0.897/0.966, 0.894/0.966, 0.769/0.941, and 0.691/0.875 when N_s is set to 3, 5, 10, 15, and 20, as shown in the right image of Fig. 7. The proposed EDA with N_s=5 only achieves 0.003 improvements on AOR compared with the EDA with N_s=10 at the cost of a 10x slower speed. Therefore, it is a good trade-off between accuracy and efficiency to set N_s to 10.

5.3. Time Cost

The proposed EDA is evaluated on a PC with an Intel Core i7 CPU and an NVIDIA GTX 1080 GPU. For conventional tracking methods, they are synchronized with the global camera shutter, and thus their speeds are evaluated by a synchronous criterion (e.g., 25 frames per second and above can be considered as real-time). Since EDA works asynchronously, the synchronous criterion is not suitable for it. Instead, in event-based studies, the efficiency of event-based methods is commonly evaluated in terms of events per second (EPS) rather than the conventional frames per second (FPS). EDA runs at the average speed of 56.33K/31.72K EPS on the test sequences with/without the GPU support. Since the speed of event generation (about 30K EPS in relatively clutter-free scenes) is usually slower than the running speed of EDA (i.e., 56.33K/31.72K EPS), the proposed EDA is well suited for most common tasks.

6. Conclusions

In this paper, we propose a novel unifying event data association approach (EDA) to effectively and explicitly handle the essential event data association problem. The proposed EDA can asynchronously gather the event data over time for processing. Then, we present a deterministic model hypothesis generation strategy to effectively generate model hypotheses in the spatio-temporal domain

from the gathered events. After that, we provide a two-stage weighting algorithm, which robustly weighs and selects the true models from the generated model hypotheses through multi-structural model fitting. Furthermore, we propose an adaptive model selection strategy to automatically determine the number of the true models. Finally, the selected true models are exploited to perform the final data association for remote object tracking. Extensive experiments on several challenging datasets demonstrate the effectiveness and superiority of the proposed EDA.

Author Contributions: Conceptualization, H.C. and Y.W.; methodology, H.C.; software, H.C. and Y.W.; validation, Y.W. and Y.P.; formal analysis, H.C.; investigation, H.C.; resources, H.C.; data curation, Y.W.; writing-original draft preparation, H.C.; writing—review and editing, Y.W. and Y.P.; visualization, Y.P.; supervision, Y.W. and Y.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the China Postdoctoral Science Foundation (Grant Nos. GZC20233362, 2024MD754043, and 2024MD754039), Chongqing Postdoctoral Innovative Talents Support Program (Grant No. CQBX202316), Science and Technology Research Program of Chongqing Municipal Education Commission (Grant No. KJQN202400648), and National Natural Science Foundation of China (Grant No. 62301103).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to thank all the reviewers for their valuable contributions to our work.

Conflicts of Interest: The authors declare no conflict of interest.

s

y, B.P.; Baccus, S.A.; Meister, M. Segregation of object and background motion in the retina. *Nature* **2003**, *423*, 401–408.
iner, P.; Posch, C.; Delbruck, T. A 128×128 120 dB 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of te Circuits* **2008**, *43*, 566–576.
C.; Berner, R.; Yang, M.; Liu, S.C.; Delbruck, T. A 240× 180 130 dB 3 μs latency global shutter spatiotemporal vision sensor. *IEEE f Solid-State Circuits* **2014**, *49*, 2333–2341.
C.; Matolin, D.; Wohlgenannt, R. A QVGA 143 dB dynamic range frame-free PWM image sensor with lossless pixel-level video ssion and time-domain CDS. *IEEE Journal of Solid-State Circuits* **2011**, *46*, 259–275.
G.; Delbrück, T.; Orchard, G.; Bartolozzi, C.; Taba, B.; Censi, A.; Leutenegger, S.; Davison, A.J.; Conradt, J.; Daniilidis, K.; et al. ased vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2020**, *44*, 154–180.
G.; Rebecq, H.; Scaramuzza, D. A unifying contrast maximization framework for event cameras, with applications to motion, depth, cal flow estimation. In Proceedings of the CVPR, 2018, pp. 3867–3876.
Z.; Atanasov, N.; Daniilidis, K. Event-based feature tracking with probabilistic data association. In Proceedings of the ICRA, 2017, pp. 70.
G.; Gehrig, M.; Scaramuzza, D. Focus is all you need: Loss functions for event-based vision. In Proceedings of the CVPR, 2019, pp. 2289.
Sun, H.; Wang, S. SAT: Spectrum-Adaptive Transformer with Spatial Awareness for UAV Target Tracking. *Remote Sensing* **2024**, *17*, 52.
; Yang, Z.; Feng, W. SiamRhic: Improved Cross-Correlation and Ranking Head-Based Siamese Network for Object Tracking in Remote Videos. *Remote Sensing* **2024**, *16*, 4549.
Y.; Huang, H.; Niu, M.; Miah, M.S.; Wang, H.; Gao, T. UAV Complex-Scene Single-Target Tracking Based on Improved Re-Detection lgorithm. *Remote Sensing* **2024**, *16*, 1768.
; Yin, G. Attention-Enhanced One-Shot Attack against Single Object Tracking for Unmanned Aerial Vehicle Remote Sensing Images. ensing **2023**, *15*, 4514.
iao, G.; Liu, J. Multimodal Features Alignment for Vision–Language Object Tracking. *Remote Sensing* **2024**, *16*, 1168.
.; Chen, H.; Wu, Q.; Xia, C.; Li, J. Joint spatio-temporal similarity and discrimination learning for visual tracking. *IEEE Transactions on and Systems for Video Technology* **2024**.
H.; Jin, L.; He, Y.; Huo, Z.; Wang, G.; Sun, X. Detector–tracker integration framework for autonomous vehicles pedestrian tracking. ensing **2023**, *15*, 2088.
Cui, Y.; Ren, Z.; Li, Z.; Wang, S.; Jiao, L. Siamese Multi-Scale Adaptive Search Network for Remote Sensing Single-Object Tracking. ensing **2023**, *15*, 4359.

.; Liu, Y.; Yan, Y.; Zhang, L.; Wang, H. Robust visual tracking via spatio-temporal adaptive and channel selective correlation filters. *Recognition* **2021**, *112*, 107738.

Chen, L.; Jiang, Y.; Xi, B.; Xie, W.; Li, Y. Feature-level fusion network for hyperspectral object tracking via mixed multi-head ntion learning. *Remote Sensing* **2025**, *17*, 997.

Li, Z.; An, W.; Wang, Y.; He, X.; Luo, Y.; Ling, Q.; Li, M.; Lin, Z. SPTrack: Spectral Similarity Prompt Learning for Hyperspectral racking. *Remote Sensing* **2024**, *16*, 2975.

Zhang, J.; Lin, Z.; Li, C.; Huo, B.; Zhang, Y. SiamCAF: Complementary attention fusion-based Siamese network for RGBT tracking. *ensing* **2023**, *15*, 3252.

, F.; Fermuller, C.; Ros, E. Real-time clustering and multi-target tracking using event-based sensors. In Proceedings of the IROS, 2018, –5769.

s-Mesa, L.A.; Serrano-Gotarredona, T.; Ieng, S.H.; Benosman, R.; Linares-Barranco, B. Event-driven stereo visual tracking algorithm object occlusion. *IEEE Transactions on Neural Networks and Learning Systems* **2017**, *29*, 4223–4237.

A.; Bartolozzi, C. Robust visual tracking with a freely-moving event camera. In Proceedings of the IROS, 2017, pp. 3769–3776.

in, A.; Fermüller, C.; Parameshwara, C.; Aloimonos, Y. Event-based moving object detection and tracking. In Proceedings of the IROS, . 1–9.

.; Wu, Q.; Liang, Y.; Gao, X.; Wang, H. Asynchronous tracking-by-detection on adaptive time surfaces for event-based object tracking. edings of the ACM MM, 2019, pp. 473–481.

.; Suter, D.; Wu, Q.; Wang, H. End-to-end learning of object motion estimation from retinal events for event-based object tracking. In ings of the AAAI, 2020, pp. 10534–10541.

Z.; Yuan, L.; Chaney, K.; Daniilidis, K. Unsupervised event-based learning of optical flow, depth, and egomotion. In Proceedings of R, June 2019, pp. 989–997.

G.; Lund, J.E.; Mueggler, E.; Rebecq, H.; Delbruck, T.; Scaramuzza, D. Event-based, 6-DOF camera tracking from photometric depth EEE Transactions on Pattern Analysis and Machine Intelligence **2017**, *40*, 2402–2412.

en, T.; Gallego, G.; Drummond, T.; Kleeman, L.; Scaramuzza, D. Event-based motion segmentation by motion compensation. In ings of the ICCV, 2019, pp. 7244–7253.

.; Ieng, S.H.; Benosman, R. Asynchronous event-based corner detection and matching. *Neural Networks* **2015**, *66*, 91–106.

X.; Orchard, G.; Galluppi, F.; Shi, B.E.; Benosman, R.B. Hots: A hierarchy of event-based time-surfaces for pattern recognition. *IEEE ons on Pattern Analysis and Machine Intelligence* **2017**, *39*, 1346–1359.

M.A.; Bolles, R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated phy. *Communications of the ACM* **1981**, *24*, 381–395.

.T.; Chin, T.J.; Schindler, K.; Suter, D. Interacting geometric priors for robust multimodel fitting. *IEEE Transactions on Image Processing* 4601–4610.

D.; Matas, J. Graph-cut RANSAC. In Proceedings of the CVPR, 2018, pp. 6733–6741.

.; Fusiello, A. T-linkage: A continuous relaxation of j-linkage for multi-model fitting. In Proceedings of the CVPR, 2014, pp. 3954–3961.

L.; Fusiello, A. Fitting multiple heterogeneous models by multi-class cascaded t-linkage. In Proceedings of the CVPR, 2019, pp. 68.

; Ma, J.; Wang, S.; Chen, C. Deterministic model fitting by local-neighbor preservation and global-residual optimization. *IEEE ons on Image Processing* **2020**, *29*, 8988–9001.

; Wang, H.; Lai, T.; Suter, D. Hypergraph modelling for geometric model fitting. *Pattern Recognition* **2016**, *60*, 748–760.

.; Xiao, G.; Yan, Y.; Suter, D. Searching for representative modes on hypergraphs for robust geometric model fitting. *IEEE Transactions rn Analysis and Machine Intelligence* **2019**, *41*, 697–711.

; Wang, H.; Yan, Y.; Suter, D. Superpixel-guided two-view deterministic geometric model fitting. *International Journal of Computer* **019**, *127*, 323–339.

hao, J.; Jiang, J.; Zhou, H.; Guo, X. Locality preserving matching. *International Journal of Computer Vision* **2019**, *127*, 512–531.

ang, X.; Fan, A.; Jiang, J.; Yan, J. Image matching from handcrafted to deep features: A survey. *International Journal of Computer Vision* *9*, 23–79.

H.; Chin, T.J.; Suter, D. Simultaneously fitting and segmenting multiple-structure data with outliers. *IEEE Transactions on Pattern and Machine Intelligence* **2011**, *34*, 1177–1192.

er, E.; Rebecq, H.; Gallego, G.; Delbruck, T.; Scaramuzza, D. The event-camera dataset and simulator: Event-based data for pose on, visual odometry, and SLAM. *International Journal of Robotics Research* **2017**, *36*, 142–149.

to, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of V, 2016, pp. 850–865.

n, M.; Bhat, G.; Khan, F.S.; Felsberg, M.; et al. ECO: Efficient convolution operators for tracking. In Proceedings of the CVPR, 2017, pp. 46.

u, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. SiamRPN++: Evolution of siamese visual tracking with very deep networks. In Proceedings VPR, 2019, pp. 4282–4291.
n, M.; Bhat, G.; Khan, F.S.; Felsberg, M. ATOM: Accurate tracking by overlap maximization. In Proceedings of the CVPR, 2019, pp. 69.
nck, C.; Rebecq, H.; Stoffregen, T.; Barnes, N.; Mahony, R.; Scaramuzza, D. CED: Color event camera dataset. In Proceedings of the , 2019, pp. 1684–1693.
in, A.; Ye, C.; Fermuller, C.; Aloimonos, Y.; Delbruck, T. EV-IMO: Motion segmentation dataset and learning pipeline for event . In Proceedings of the IROS, 2019, pp. 6105–6112.

ublisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property n any ideas, methods, instructions or products referred to in the content.