

# E-MLB: Multilevel Benchmark for Event-Based Camera Denoising

Saizhe Ding\*, Jinze Chen\*, Yang Wang†, Yu Kang, *Senior Member, IEEE*, Weiguo Song, Jie Cheng and Yang Cao, *Member, IEEE*,

**Abstract**—Event cameras, such as dynamic vision sensors (DVS), are biologically inspired vision sensors that have advanced over conventional cameras in high dynamic range, low latency and low power consumption, showing great application potential in many fields. Event cameras are more sensitive to junction leakage current and photocurrent as they output differential signals, losing the smoothing function of the integral imaging process in the RGB camera. The logarithmic conversion further amplifies noise, especially in low-contrast conditions. Recently, researchers proposed a series of datasets and evaluation metrics but limitations remain: 1) the existing datasets are small in scale and insufficient in noise diversity, which cannot reflect the authentic working environments of event cameras; and 2) the existing denoising evaluation metrics are mostly referenced evaluation metrics, relying on APS information or manual annotation. To address the above issues, we construct a large-scale event denoising dataset (multilevel benchmark for event denoising, E-MLB) for the first time, which consists of 100 scenes, each with four noise levels, that is 12 times larger than the largest existing denoising dataset. We also propose the first nonreference event denoising metric, the event structural ratio (ESR), which measures the structural intensity of given events. ESR is inspired by the contrast metric, but is independent of the number of events and projection direction. Based on the proposed benchmark and ESR, we evaluate the most representative denoising algorithms, including classic and SOTA, and provide denoising baselines under various scenes and noise levels. The corresponding results and codes are available at <https://github.com/KugaMaxx/cuke-emlb>.

**Index Terms**—Event camera, event denoising, nonreference denoising metric.

## I. INTRODUCTION

EVENT cameras, such as the Dynamic Vision Sensor (DVS), are novel biologically inspired devices [1], [2]. In contrast to traditional frame-based cameras, which capture global scene brightness at a fixed rate, event cameras can asynchronously perceive the environmental brightness change in each pixel and report log-intensity change signals at microsecond resolution [3], [4]. These features show great application potential in many fields, such as optical flow estima-

tion [5]–[7], high-speed video interpolation [8]–[10], feature tracking/detection [11]–[13] and simultaneous localization and mapping (SLAM) [14]–[19].

However, due to its differential imaging mechanism, the event camera is sensitive to various types of noise [4], [20]. In this paper, we are mainly concerned with background activity (BA) noise [21], which is the main type of noise in event cameras. As shown in Fig. 1 (a), with the overall brightness reduction, the noise level in the event camera output will gradually increase. More specifically, the input signal will be disturbed due to the perturbation of incoming light before the receiving photodiodes and junction leakage current of the imaging circuit, as shown in Fig. 1 (b). In conventional cameras, such noise input will be suppressed to a great extent because of the smoothness of the integration function, thus maintaining good imaging quality. However, in the event camera, the noise is much more obvious due to the continuous differential sampler, and the logarithmic operation will further amplify the noise, leading to the production of BA, as shown in Fig. 1 (c).

Several event denoising datasets [22]–[24] and denoising metrics [22], [25], [26] have been proposed to date. Based on these, various event denoising algorithms [22]–[37] have been presented and have achieved remarkable progress. However, existing event denoising datasets and denoising metrics still have the following limitations: 1) the scale of existing datasets is small, and the noise diversity is limited and unable to cover authentic working environments of event cameras. Specifically, the events in existing datasets are mainly captured in similar lighting conditions, resulting in small variances in different event sequences, which cannot cover the real noise distribution in practical environments. 2) The existing denoising evaluation metrics are mostly reference evaluation metrics, relying on active pixel sensors (APS) and inertial measurement unit (IMU) information [22] or manual annotation [25], [26]. However, APS information is not always available, and its quality cannot be guaranteed, especially in low-light environments. In addition, the microsecond event camera can output millions of events per second, and it is impractical to label each event manually.

To better study the influence of noise on event-based visual cues and enable future research on event denoising, we propose a large-scale event denoising dataset and a nonreference event denoising metric. First, we construct a novel large-scale event denoising dataset, which has three advantages over existing datasets: 1) *Various scenes*. The number of sequences in the E-MLB dataset is 12 times larger than existing datasets.

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants 62206262, 62033012 and 52074252.

\* Saizhe Ding and Jinze Chen contributed equally to this paper.

† Yang Wang is the corresponding author of this paper.

Saizhe Ding and Weiguo Song are with State Key Laboratory of Fire Science, University of Science and Technology of China, Hefei, Anhui Province, China. (e-mail: dszh2020@mail.ustc.edu.cn, wgsong@ustc.edu.cn)

Jinze Chen, Yang Wang, Yu Kang, Yang Cao are with School of Informatics, University of Science and Technology of China, Hefei, Anhui Province, China. (e-mail: chjz@mail.ustc.edu.cn, ywang120@ustc.edu.cn, kangduyu@ustc.edu.cn, forrest@ustc.edu.cn)

Jie Cheng is with Huawei Technologies Co., Ltd. (e-mail: chengjie8@huawei.com)

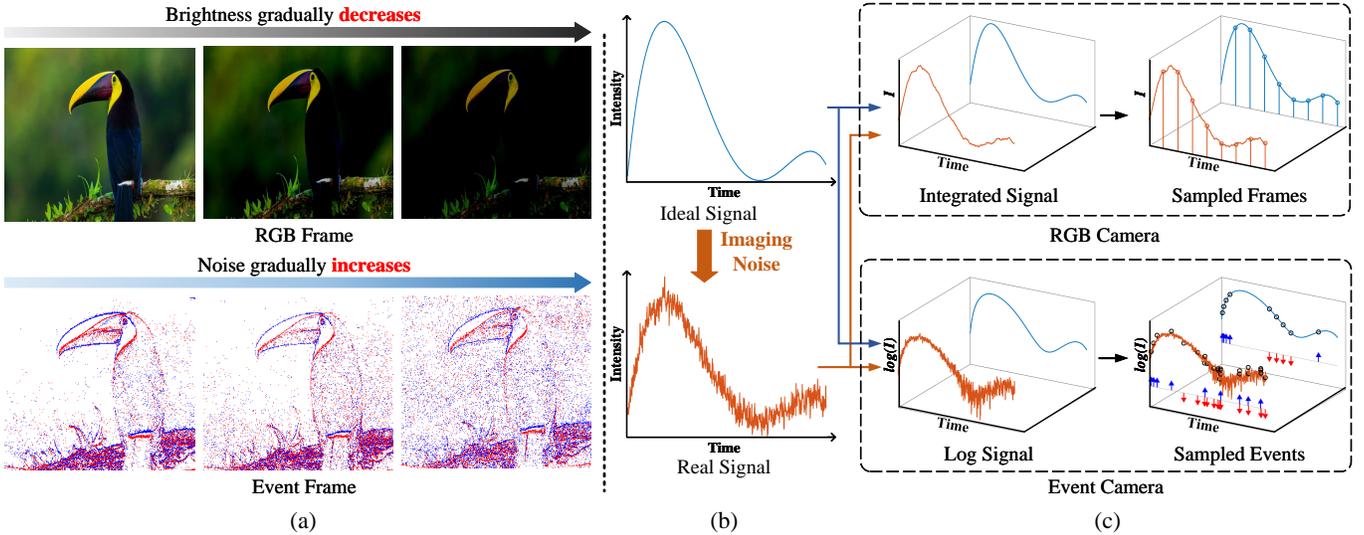


Fig. 1. The difference between event and RGB cameras in signal processing. (a) illustrates that the light intensity is inversely correlated to the noise level of captured events, *i.e.*, with the light intensity gradually decreasing, the noise level increases in the event frame<sup>1</sup>. (b)-(c) explain why the event camera generates so much noise in poor lighting conditions. The main reason is that the continual sampling (or differential sampling) in the event camera cannot smooth noisy signals in the integrated sampling manner of a frame-based camera, which makes the event-based denoising task unique and challenging.

2) *Varying light conditions.* To better cover the actual lighting conditions in DVS working environments, we collected a large number of event sequences at different times (from day to night). We placed neutral density (ND) filters with fractional transmittances of 1/4, 1/16 and 1/64 in front of the event camera to simulate different light intensities. Thus, four event sequences with different noise levels were obtained for any given scene. 3) *Multiple motion types.* Our dataset contains events generated by objects with different motion types, including translation, rotation, and a combination of both 2D and 3D with perspective changes.

Second, we propose a novel nonreference event denoising metric, termed the event structural ratio (ESR), to reduce the dependence of evaluation metrics on APS information and manual annotation. It has the following advantages: 1) *Effective ranked noise level.* The principle of ESR is to judge the noise level of an event stream. Since each denoising method leads to different noise levels, we can use ESR to evaluate these denoising events and, as a result, distinguish the denoising effect indirectly; 2) *Reflect the intrinsic property of events.* The calculated ESR is not dependent on either the number of events or the projection directions. Therefore, it is an intrinsic property of the events alone. 3) *Easy to calculate.* The only information needed to calculate ESR is the event data, and only basic arithmetic operations are needed.

In summary, the main contributions of this paper are three-fold:

- We construct a large-scale event denoising dataset Multi-Level Benchmark (E-MLB) for the first time, which is 12 times larger than the largest existing dataset. Our proposed dataset far exceeds existing datasets in rich real-world scenes and multiple noise levels.

- We propose the first nonreference event denoising metric, the event structural ratio (ESR), which measures the structural intensity of events without additional information sources such as the APS frame and IMU data. The proposed ESR is easy to calculate and faithfully indicates the noise level of event data under various scenes and lighting conditions.
- We conduct extensive experiments with 11 state-of-the-art denoising methods on the E-MLB dataset and give the ESR score of each algorithm. We hope that the comparative analysis will contribute to future event denoising research.

The remainder of the paper proceeds as follows. In Section II, we introduce relevant works on event denoising datasets, metrics and algorithms. In Section III, we describe the collection details of our E-MLB dataset. Then, we illustrate our proposed event denoising metrics and provide a detailed rigorous mathematical proof in Section IV. In Section V and Section VI, we report the experimental results and give a conclusion, respectively.

## II. RELATED WORKS

### A. Event Denoising Datasets

Some denoising datasets have been presented recently to suppress the impact of noise on event cameras. DVSNOISE20 [22] provides 48 event sequences on 16 stationary scenes, which were captured by a *DAVIS 346* mounted in a gimbal restricted to rotation-only movement. It also provides ground-truth labels representing event generation probability by combining the APS and IMU information. ENFS [24] contains 100 sequences. *DAVIS 346* camera was mounted on top of a table and shot a monitor playing the need-for-speed (NFS) [38] dataset. RGB *DAVIS* [23] provides 20 real event sequences from a *DAVIS 240* camera, including indoor

<sup>1</sup>The event frame is obtained by accumulating events for each pixel, where red represents positive events and blue represents negative events.

and outdoor scenes, as well as high-resolution frames from a conventional RGB camera. Although these datasets provide a large quantity of realistic noisy data, they were collected under limited lighting conditions; some of them (*e.g.* DVSNOISE20 and ENFS) contain only restricted motion, which cannot cover authentic camera working scenarios.

To solve the lack of ground truth labels, DND21 [39] collected realistic pure noise and pure signal sequences and then synthesized hybrid noisy sequences. Additionally, some simulators, such as ESIM [40] and V2E [41], can be used to generate synthetic DVS events from provided image or video datasets and control noise generation. However, due to the complexity of the actual noise distribution, the above methods cannot reflect real situations.

### B. Event-based Denoising Metrics

Percentage of signal/noise remaining (PSR/PNR) [25] treats the events that fall in the manually generated bounding box as signals, calculating the percentage of remaining events inside (or outside) the bounding box. Noise in real (NIR) [26] and relative plausibility measure of denoising (RPM) [22] annotate the probability of each event. The former convolves the event stream with a Gaussian kernel, and the latter combines APS and IMU to calculate the probability of event occurrence in each space-time coordinate. In addition, there are some metrics designed on synthetic datasets. Event denoising precision (EDP) [42] can briefly report the ratio of the total number between the denoised event stream and the original event stream. [39] plots receiver operating characteristic (ROC) curves to compare different event data.

Although the aforementioned metrics can evaluate denoising algorithm performance, some methods rely heavily on synthetic data and this generalization to real event data is still unclear. Others need ground truth data by either manually labeling or introducing additional information sources, which may become invalid in a practical environment where labels are not always available.

### C. Event-based Denoising Algorithms

Statistical methods were the earliest classical approaches for event-based denoising. In [28], outliers are filtered by calculating the density for each event in their local spatial-temporal neighborhood and setting the threshold to judge low-density events. Then, based on this theory, approaches such as [29], [32], [39], [42], [43] reduce the operating complexity by setting different event storage strategies. Other works, such as [25], [26], [44], introduce additional process stages to eliminate dead pixels or sharpen edges. However, these density statistics methods are difficult to apply across a wide variety of noise and require manually finetuning parameters to deal with different scenarios.

Other algorithm filters conduct event denoising in the context of surface fitting. EV-Gait [36] performs local plane optical flow estimation and filters noisy events to smooth the optical flow surface. Afterward, the guided event filter (GEF) [23] combines the gradient of active pixel sensor (APS) frames. In contrast, time surface (TS) [30], [34], [35]

transforms events from unit impulses into a representation that is monotonically decreasing with time, which solves the sparsity problem in the local plan fitting process. These fitting methods are well suited for a single moving object but perform poorly in low-light conditions or complex scenarios.

Learning-based methods have been widely used in event-based denoising most recently. For example, a K-SVD method [31] was proposed to extract the sparse features from several noise-free event frames. In [39], a multilayer perceptron denoising filter (MLPF) was used to calculate the probability of noise event-by-event. In addition, some convolutional neural network (CNN) methods [22], [24], [45], [46] have also been proposed recently. EDnCNN [22] trains a binary classification network using the probability tag of each event, which is estimated by combining APS and IMU information. EDnCNN can classify individual events as signals or noise well but is a time-consuming network. EventZoom [24] is a high-efficiency U-Net that achieves event denoising in a noise-to-noise fashion.

## III. E-MLB DATASET

In this section, we introduce the collection details of our E-MLB dataset. We first introduce our capture device. Then, the shooting details and photographic accessories used are presented. Finally, the comparison of E-MLB with the existing datasets is given.

**Event Sensor:** The type of event camera we used was a *DAVIS346*, which can simultaneously output a spatially aligned event stream (120 dB) and intensity images (56 dB) with a resolution of  $346 \times 260$ . In addition, to simulate different lighting conditions, we placed three neutral density filters (ND filters) with different transmittance in front of the lens, as shown in Fig. 2 (a).

**Collection Details:** Benefiting from the high dynamic property, the DVS is widely used in extreme light conditions, such as low-light and overexposed conditions [47], [48]. However, the noise level output by DVS gradually increases as the light intensity increases/decreases, as shown in Fig. 1. To better analyze the relationship between noise level and light intensity, we place the ND filter to simulate the different light conditions, as shown in Fig. 2 (a). For each scene, we first capture the original scene in the natural light condition. Then, we add the ND filter in front of the DVS and repeat the capture process. In this paper, we use three kinds of ND filters with different transmittance (1/4, 1/16, and 1/64), which are denoted as ND4, ND16, and ND64, respectively. The captured samples are shown in Fig. 2 (b). For each light condition, we repeatedly shoot the scene 3 times. The simulated light intensity and diversity are highly dependent on the original natural light intensity. Thus, to further increase the light diversity, we change the capture time from day to night to guarantee the diversity of natural lighting conditions.

In addition to changing the light intensity, we also change the shooting scene to guarantee the diversity of the content of the captured event sequence. In this paper, we select 100 scenes, including both indoor and outdoor scenes and diverse motion types (translation, rotation, and combination of both

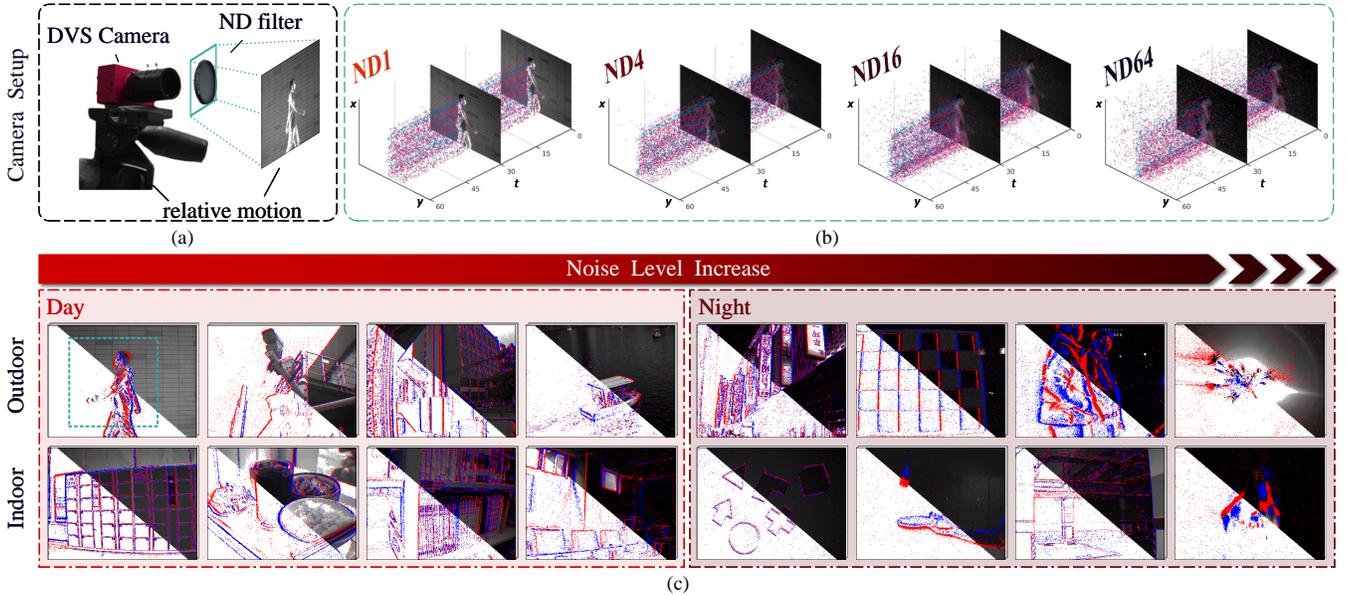


Fig. 2. (a) The event camera and ND filters were used for capturing event sequences. (b) The captured event stream with different ND filters<sup>2</sup>. The noise level gradually increases with the amount of light entering the lens reduction. (c) Examples of event sequences in the E-MLB from daytime to night. In each square, the lower-left is the converted event frame, and the upper-right is the hybrid image, including the event and APS frame.

TABLE I  
THE COMPARISON OF OUR PROPOSED E-MLB WITH EXISTING EVENT DENOISING DATASET.

Datasets	Camera	Resolution	APS	IMU*	Scenes	Sequences	Capture/s	DoF	Noise Level
DVSN0ISE20 [22]	DAVIS 346	346 × 260	Gray	✓	16	48	807	Cam.	-
RGBDAVIS [23]	DAVIS 240	190 × 180	RGB	-	20	20	122	All.	-
ENFS [24]	DAVIS 346	224 × 125	-	-	1	100	4238	Obj.	-
DND-21 [39]	DAVIS 346	346 × 260	-	-	-	8	-	All.	-
E-MLB	DAVIS 346	346 × 260	Gray	✓	<b>100</b>	<b>1200</b>	<b>7300</b>	All.	<b>4</b>

\* Inertial Measurement Unit

in 2D and 3D with perspective change). In addition, we provide the corresponding APS frame and IMU data for each captured event sequence. It should be noted that the APS quality will decrease as light intensity decreases. Considering that the event camera has a superior high dynamic range, we also include some special sequences that create more challenges for denoisers, such as extremely low light scenes (with high background activity and blurred edges), special weather conditions (rainy and snowy days), and high-speed objects. Some example sequences can be found in Fig. 2 (c). A comparison of our E-MLB with the existing event denoising dataset is reported in Tab. I.

#### IV. EVENT STRUCTURAL RATIO

In Section IV-A, we review the working principle of event cameras and the event contrast measurement, in which event contrast is the main inspiration of our denoising metric. In Section IV-B, we introduce our proposed event structure ratio, and

<sup>2</sup>For consistency, ND1 is used to represent the data captured without any ND filters

the relevant derivation and proof can be found in Section IV-C. Evaluations on ESR are conducted in Section IV-D, including both synthetic and real experiments, which demonstrate that our proposed ESR is a good denoising indicator.

##### A. Preliminaries

**Working Principle:** In event cameras, each pixel works asynchronously and will trigger an event  $e_k := (x_k, t_k, p_k)$  when its logarithmic brightness change reaches the predefined contrast threshold  $c$ , which can be defined as:

$$\Delta L \doteq L(x_k, t_k) - L(x_k, t_k - \Delta t_k) = c \cdot p_k \quad (1)$$

where  $x_k := (x_k, y_k)$  is the pixel position of the  $k$ -th event.  $t_k$  is the timestamp, and  $\Delta t_k$  is the time interval since pixel  $(x_k, y_k)$  last reaches the threshold.  $p_k \in \{-1, +1\}$  is the polarity, representing the decrease and increase in brightness, respectively.  $L(x_k, t_k) := \log I(x_k, t_k)$  denotes the log intensity.

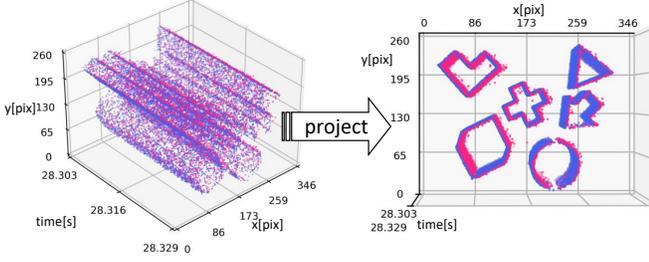


Fig. 3. Since the event camera is responsive to edges, we will obtain an image of the objects' edges after projecting the events along the trajectories to the 2D dimension, which helps us to analyze its statistical characteristics more easily.

The difference between log intensity in a duration of  $t$  can be obtained by integrating the sequences of events [49]:

$$L(\mathbf{x}, t) - L(\mathbf{x}, 0) \doteq c \cdot \int_0^t \sum_k e_k(\mathbf{x}, \tau) d\tau \quad (2)$$

where  $e_k(\mathbf{x}, t)$  can be described by using Dirac function  $\delta(\cdot)$ :

$$e_k(\mathbf{x}, t) = p_k \cdot \delta(\mathbf{x} - \mathbf{x}_k, t - t_k) \quad (3)$$

**Event Contrast:** Since event cameras are highly responsive to the moving edges of an object [50], a set of events will occur on the edge trajectories as long as relative movement occurs between the camera and objects. In contrast, given a set of events  $\{e_k\}_N$ , we can project (warp) these events to a reference time  $t_{ref}$  along these trajectories by a warping function  $W$ :

$$e_k := (\mathbf{x}_k, t_k, p_k) \xrightarrow{W} e'_k := (\mathbf{x}'_k, t_{ref}, p_k) \quad (4)$$

After projecting, we obtain an accumulated 2D histogram, also known as an image of warped events (IWE) [51]:

$$\text{IWE}(\mathbf{x}) = \sum_{k=1}^N b_k \delta(\mathbf{x} - \mathbf{x}'_k) \quad (5)$$

where  $b_k$  is the weight of the summation of  $e_k$ . Here, we set  $b_k = 1$  to facilitate the subsequent derivations. Usually, the warping function can be modeled as linear motion (optic flow), rotational motion, 4-DOF motion and so on [52]. If we correctly model the warping function and estimate accurate parameters, the IWE will form an edge-like image. Taking Fig. 3 as an example, for some simple shapes performing translation motion relative to the camera, we can project events along the translation direction to obtain a clear and sharp edge-like IWE.

Because edge strength is directly related to image contrast [51], we can use IWE to measure scene contrast. Here, we use an image-based contrast metric named the total sum of squares (TSS):

$$\text{TSS} = \sum_{\mathbf{x}} \text{IWE}^2(\mathbf{x}), \quad (6)$$

where the summation is carried over all the pixels. The area of spatial support  $L$  (the total number of pixels that output events) can be defined as:

$$L := \sum_{\text{IWE}(\mathbf{x}) > 0} 1, \quad (7)$$

TSS and  $L$  are inversely correlated most of the time. Given a number of events, the more aggregated the events are in IWE, the less spatial support  $L$  the event image has. In other words, the event contrast will decrease when the data are influenced by noise, which we believe is an important clue to judging the impact of noise.

However, TSS and other contrast metrics are highly dependent on the number of events, and they cannot be directly used as event denoising metrics. Taking *TSS* as an example, it will always assign the highest score for the denoising method that outputs the maximum number of events. In practice, we cannot guarantee that the different denoising methods keep the same number of events.

### B. Definition of ESR

To address the above issues, we extract an invariant from the TSS, which is called the normalized TSS (NTSS):

$$\text{NTSS} := \sum_{i=1}^K \frac{n_i(n_i - 1)}{N(N - 1)} \quad (8)$$

where  $K$  is the total number of pixels in the IWE.  $N$  is the total number of events, and  $n_i$  is the sum of all events that occur on pixel  $(x_i, y_i)$ . NTSS is used to represent the relative contrast of the scene regardless of the number of events. Nevertheless, due to the intrinsic deficiency of the contrast metric, the NTSS tends to assign a higher score to the method that performs overdenoising. An extreme case is that if only one event remains, the calculated NTSS will reach the upper bound and fail to faithfully represent the noise situation. Therefore, we add a penalty coefficient before NTSS, which is defined as:

$$L_N := K - \sum_{i=1}^K \left(1 - \frac{M}{N}\right)^{n_i} \quad (9)$$

where  $L_N$  is the number of nonzero pixels (or the area of spatial support) in the IWE.  $M$  is the reference number of events used for interpolation, which is fixed during the entire evaluation process. In this way, the normalized contrast of any  $N$  events can be interpolated to that of fixed  $M$  events. Based on the invariant representation of scene contrast NTSS and penalty coefficient  $L_N$ , we can finally define the proposed ESR as:

$$\text{ESR} := \sqrt{\text{NTSS} \cdot L_N}, \quad (10)$$

### C. Proof of NTSS and $L_N$

For small duration  $\Delta t$ , the probability of a given number of events follows the Poisson distribution [32]:

$$P(N_{\mathbf{x}}(t) = m) = e^{-\lambda_{\mathbf{x}} t} \frac{(\lambda_{\mathbf{x}} t)^m}{m!} \quad (11)$$

where  $P(N_{\mathbf{x}}(t) = m)$  is the probability of  $m$  events occurring. Event rate  $\lambda_{\mathbf{x}}$  is the rate of triggering events at pixel per unit time [49], which can be derived from Eq. (2) as:

$$\lambda_{\mathbf{x}} := \frac{1}{\Delta t} \cdot \frac{L(\mathbf{x}, t) - L(\mathbf{x}, 0)}{c} = \frac{\int_0^t \sum_k e_k(\mathbf{x}, \tau) d\tau}{\Delta t} \quad (12)$$

then we can obtain the uniform event rate as:

$$p_x = \frac{\lambda_x}{\sum_x \lambda_x}, \quad (13)$$

where  $p_x$  represents the relative portion of event rate  $\lambda_{x,y}$  and sums to 1. Given the sum of the number of events, the joint probability distribution of the number of events in different pixels follows a multinomial distribution, provided the number of events in each pixel follows the Poisson distribution and all the pixels are independent. Therefore, events can be viewed as drawn from a multinomial distribution provided that the number of events is fixed. Let the total number of pixels of the event image be  $K$ , and use flattened index  $i \in \{1, 2, \dots, K\} = (x_i, y_i)$  to represent different pixels for simplicity of notation; then we have:

$$(n_1, \dots, n_K) | Y = N \sim \text{Multinomial}(N, (p_1, \dots, p_K)), \quad (14)$$

$$n_i | Y = N \sim \text{Binomial}(N, p_i), \quad (15)$$

$$Y = \sum_{i=1}^K n_i, \quad (16)$$

$$n_i = N_{\mathbf{x}_i}(t). \quad (17)$$

After deriving the uniform event rate  $p_x$ , we can further derive the NTSS and  $L_N$ . The expectation of TSS is:

$$\mathbb{E}[\text{TSS} | Y = N] = \sum_{i=1}^K \mathbb{E}[n_i^2 | Y = N]. \quad (18)$$

Because the distribution of  $N_i(t)$  conditioned on  $M(t)$  is binomial, we introduce

$$f(x, p) = (px + (1-p))^N, \quad (19)$$

then from Eq. (15), we have:

$$\mathbb{E}[n_i^2 | Y = N] = \sum_{k=1}^N k^2 \binom{N}{k} p^k (1-p)^{N-k} \quad (20)$$

$$= \left(x \frac{\partial}{\partial x}\right)^2 \circ f(x, p_i) |_{x=1} = Np_i + N(N-1)p_i^2. \quad (21)$$

so for TSS, there is:

$$\mathbb{E}[\text{TSS} | Y = N] = \sum_{i=1}^K Np_i + N(N-1)p_i^2 \quad (22)$$

$$= N + N(N-1) \sum_{i=1}^K p_i^2. \quad (23)$$

$\sum_{i=1}^K p_i^2$  is an inherent property of the scene and is invariant with respect to the number of events  $N$ . In effect, it can be estimated by:

$$\sum_{i=1}^K p_i^2 \approx \sum_{i=1}^K \frac{n_i(n_i-1)}{N(N-1)}. \quad (24)$$

$\sum_{i=1}^K p_i^2$  can be viewed as the normalized TSS, and its estimation is denoted as NTSS:

$$\text{NTSS} := \sum_{i=1}^K \frac{n_i(n_i-1)}{N(N-1)}. \quad (25)$$

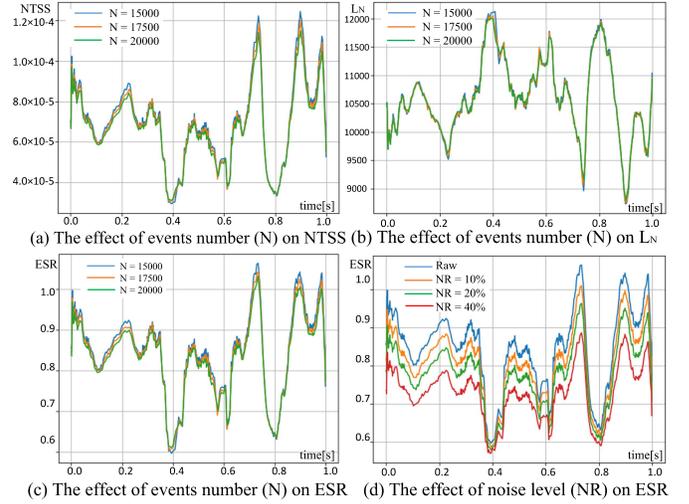


Fig. 4. The effect of event number and noise level on ESR. The NTSS and  $L_N$  are robust to the number of events, as in (a) and (b), which results in the obtained ESR also being robust to the number of events, as in (c). (d) shows that the proposed ESR is inversely correlated to the noise level, and a higher noise level corresponds to a lower ESR.

The expectation of  $L$  is:

$$\mathbb{E}[L | Y = N] = \mathbb{E}\left[\sum_{i=1}^K 1_{n_i > 0} | Y = N\right] = \sum_{i=1}^K P(n_i > 0 | Y = N) \quad (26)$$

$$= K - \sum_{i=1}^K P(n_i = 0 | Y = N) \approx K - \sum_{i=1}^K e^{-Np_i}. \quad (27)$$

There is no simple scene invariant from the expression because  $N$  and  $p_i$  are tightly coupled; however, by introducing a new random variable  $\alpha^{n_i}$ , we can interpolate the resultant  $L$  to any given number of  $M$  as if it were calculated by exactly  $M$  events. The expectation of this new random variable is:

$$\mathbb{E}[\alpha^{n_i} | Y = N] = \sum_{k=1}^N \alpha^k p_i^k (1-p_i)^{N-k} \binom{N}{k} \quad (28)$$

$$= (1 + (\alpha-1)p_i)^N \approx e^{(\alpha-1)Np_i}. \quad (29)$$

Thus, by setting  $(\alpha-1)N = -M$ , or equivalently  $\alpha = 1 - \frac{M}{N}$ , we can interpolate any  $L$  when  $Y = M$  from  $N$  events, defined as:

$$L_N := K - \sum_{i=1}^K \left(1 - \frac{M}{N}\right)^{n_i}. \quad (30)$$

#### D. Experimental Verification

To verify that the proposed NTSS,  $L_N$  and ESR are independent of the number of events, we conduct experiments on real-world event sequences. We conduct three experiments with  $N = 15,000, 17,500, \text{ and } 20,000$ .  $M$  is set to 15,000 in all three experiments. Events in the whole sequence are split into packets of events with equal sizes of  $N$ . Then, we compute the NTSS,  $L_N$ , and ESR values for each event packet with predefined parameters and draw the NTSS,  $L_N$ , and ESR

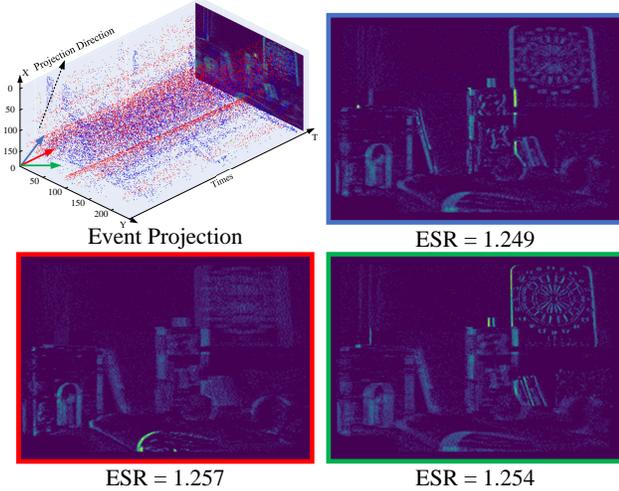


Fig. 5. The effect of projection directions on ESR. Although events are projected along different directions, the ESR values are relatively close.

curves of the entire sequence. As shown in Fig. 4 (a-b), when the number  $N$  changes from 15,000 to 20,000, the NTSS and  $L_N$  curves are very close, which verifies their independence from the event number  $N$ . As a result, the ESR curve is also independent of  $N$ , as shown in Fig. 4 (c). Then, we test the relationship between the ESR and noise level. We manually add random noise (the noise ratio is set to 10%, 20%, and 40%) to the original sequence and calculate the corresponding ESR curve. As shown in Fig. 4 (d), the noisy ESR curves have the same shape as the original curve, and the noisier the ESR curve is, the lower the ESR value, which validates that it can indicate the noise level and can be used as an event-based denoising evaluation metric.

The proposed ESR is calculated in the event frame to simplify the calculation, whereas the existing algorithms adopt different projection methods during the process. For example, EventZoom uses projections along the time axis, and GEF uses projections along the motion axis, which leads to a change in the event distribution after denoising. Therefore, we need to test the influence of different projection methods on the ESR value to verify its robustness on different algorithms. As shown in Fig. 5, we calculate the ESR value of the same event packet in different projection directions; the resultant ESR values are quite close, so the proposed ESR is also invariant to the change in projection direction. In conclusion, although the calculation is performed in the frame space, the resultant ESR is independent of the number of events and the projection method only represents the event quality, which is thus, an intrinsic property of events.

## V. EXPERIMENTAL RESULTS

In this section, we first provide the mean ESR (**MESR**) score of each representative denoiser in both our E-MLB and other existing datasets and present some typical visualization results. Then, a comparison of ESR with another denoising metric is given, which proves the superiority of ESR.

### A. Event Denoising Baselines

We select the 11 most representative event denoising methods for comparison:

- **BAF** [28], **KNoise** [32] & **DWF** [39] follow the same denoising theory. The background activity filter (BAF) counts the density of each incoming event in its eight neighborhood pixels within a time interval and filters out noise events according to a predetermined threshold. KNoise improves on this basis by allocating two blocks of memory to store the latest events of rows and columns, which gains the advantage of  $O(N)$  space complexity. The double window filter (DWF) further reduces the memory footprint by using a first-in-first-out (FIFO) queue, which stores only a few recent events and determines whether to insert a new event into this queue by comparing it with in-queue events.
- **TS** [30] & **IETS** [34] convert a sparse event stream into a dense representation. First, the time surface (TS) converts the Dirac function of time into a logarithmic decay representation, in which case the effective events form a regular manifold called the time surface. Then, it eliminates events that destroy the smoothness of the surface. The inceptive event time surface (IETS) introduces predefined time thresholds to eliminate redundant events within the same edge.
- **EvFlow** [36] calculates the gradient by local plane fitting to attain optical flow and then achieves event denoising by filtering all the events with abnormal flow values.
- **YNoise** [26] calculates the density of each incoming event in its spatiotemporal domain and then achieves event denoising by passing events with high density.
- **MLPF** [39] is a kind of multilayer perceptron (MLP) method with a single hidden layer, which is trained by adding simulated noise events in the noise-free sequences.
- **EDnCNN** [22] is a convolutional neural network. The probability of an event can be calculated by fusing APS and IMU data, which are used as the labels for each training event.
- **GEF** [23] provides two denoising modes. In the frame-guide mode, the guided event filter (GEF) extracts mutual structures between the event frame (project along optical flow) and the gradient of the APS image (by Sobel operator), then deletes unreasonable events and reallocates back to spatiotemporal space. When the APS quality is not high, GEF changes to self-guide mode, aligning two adjacent event frames and erasing inconsistent events.
- **EventZoom** [24] follows a noise-to-noise fashion that utilizes paired noisy event sequences to train a U-net and performs event reconstruction guidance using good quality videos on the network branch.

**Experimental Details:** All sequences in the E-MLB dataset are tested with the above denoising algorithms. To ensure a fair comparison, we manually fine-tune the adjustable parameters of all methods in each sequence. It should be noted that EDnCNN trained on our dataset performs inferior to the pretrained EDnCNN. The reason is that EDnCNN is highly dependent on its exclusive event noise probability labels, which

TABLE II  
THE MEAN ESR (MESR) RESULTS OF DIFFERENT DENOISING METHODS ON BOTH E-MLB DATASET AND PUBLIC AVAILABLE EVENT DENOISING DATASETS. WE MARK THE **BEST** AND SECOND BEST.

	E-MLB (Daytime)				E-MLB (Night)				RGB DAVIS		DVS NOISE20	ENFS	DND21
	ND1	ND4	ND16	ND64	ND1	ND4	ND16	ND64	Indoor	Outdoor	-	-	-
Raw	0.821	0.824	0.815	0.786	0.890	0.824	0.786	0.768	0.905	0.776	0.524	0.843	0.869
BAF [28]	0.861	0.869	0.876	0.890	0.946	0.973	0.992	0.942	0.943	0.891	0.600	1.119	0.920
KNoise [32]	0.846	0.837	0.830	0.807	0.954	0.956	0.871	0.817	0.934	0.895	0.550	0.945	0.887
DWF [39]	0.878	0.876	0.866	0.865	0.923	0.962	0.988	0.932	0.923	0.890	0.458	1.108	0.905
EvFlow [36]	0.848	0.878	0.868	0.833	0.969	0.983	0.889	0.797	0.829	1.061	0.667	1.131	<u>1.006</u>
YNoise [26]	0.866	0.863	0.857	0.821	1.009	0.943	0.875	0.792	0.825	1.077	0.654	1.178	0.966
TS [30]	0.877	0.887	0.870	0.837	<u>1.033</u>	0.944	0.886	0.797	0.837	<u>1.120</u>	0.745	<u>1.241</u>	0.985
IETS [34]	0.772	0.785	0.777	0.753	0.950	0.823	0.804	0.711	0.762	0.988	0.733	0.982	0.900
GEF [23]	<b>1.051</b>	<u>0.938</u>	0.935	0.927	1.027	0.955	0.946	0.935	<b>1.031</b>	0.986	<u>1.010</u>	1.152	0.932
MLPF [39]	0.851	0.855	0.846	0.840	0.926	0.928	0.910	0.906	<u>0.983</u>	0.932	<b>1.041</b>	1.132	0.944
EDnCNN [22]	0.887	0.908	<u>0.903</u>	<u>0.912</u>	1.001	<b>1.024</b>	<b>1.079</b>	<b>1.086</b>	0.982	1.014	0.862	1.232	0.977
EventZoom [24]	<u>0.996</u>	<b>0.988</b>	<b>0.996</b>	<b>0.970</b>	<b>1.055</b>	<u>1.007</u>	<u>1.010</u>	<u>0.988</u>	0.930	<b>1.135</b>	0.899	<b>1.417</b>	<b>1.059</b>

are restricted to stationary scenes with rotation-only camera motion; otherwise, it will be trained with incorrect training data, and our dataset does not strictly meet this requirement. Therefore, we choose a pretrained network on their DVS-NOISE20 dataset and then fine-tune it on our rotation-only sequences. In terms of GEF, we set the frame-guide model in ND1 sequences while changing to self-guided in ND4, ND16 and ND64 sequences. As mentioned in Section 3, in ND1 sequences frame-guide performs better because of the high-quality frames. However, in ND4 to ND64 frames, the quality falls and the self-guided mode can provide more reasonable denoising results. Considering that we do not provide similar paired noise sequences as in the ENFS dataset, we only trained EventZoom on its ENFS dataset sequences.

To calculate MESR, we slice the event sequence  $\mathcal{E} := \{e_k\}$  consecutively along the time, which can make the set of nonoverlapping event groups  $\{\{e_k\}^1, \{e_k\}^2, \dots, \{e_k\}^G \subseteq \mathcal{E}\}$ , where  $G$  is the number of event groups. Each group is a subset that belongs to the original sequence. In the experiment, we specified that each event group contains 30,000 events; therefore, we chose  $M = 20,000$  and  $N = 30,000$  for all sequences for a fair comparison.

### B. Experimental Results

**Quantitative Evaluation:** The mean ESR (MESR) results are reported in Table II. As shown in the first row, the MESR score of the E-MLB dataset decreases as the noise level increases (from ND1 to ND64), which again verifies the inverse correlation between the ESR value and noise level. The only exception is that the MESR score of ND1 is slightly lower than that of ND4 in the daytime sequences. We also provide MESR scores in some other event-based denoising datasets, i.e., RGB DAVIS, DVSNOISE20, ENFS and DND21. Their ESR results are similar to those sequences in our daytime E-MLB dataset because they were all captured under normal light conditions.

For the different denoising methods, it is clear that almost all the denoised sequences report better ESR scores compared to the raw sequences, especially in the higher score improvement in the night sequences. Overall, we can determine that BAF,

KNoise and DWF receive approximate ESR scoring results as they follow a similar denoising principle. Considering that IETS eliminates a large number of effective signals, it reports the poorest score. GEF outperforms other denoising methods when the APS quality is good, namely, in ND1 sequences of E-MLB and other datasets that provide related frames. Nevertheless, the denoising score drops to the second tier when GEF enters the self-guided mode. EventZoom reports the highest MESR score in almost all normal light sequences, e.g., E-MLB in the daytime, while EDnCNN presents the best performance when the noise level is higher, as shown in the ND4 to ND64 columns at night E-MLB.

It is also worth noting that our ESR still works effectively for algorithms that can generate new possible events (such as EventZoom and GEF). However, the other existing event-based denoising metrics almost fail to evaluate such self-generated events from denoisers, providing lower scores despite good denoising performance.

**Qualitative Evaluation:** First, we visualize the denoising results of different algorithms in some challenging ND1 sequences to determine the performance boundary of each denoiser, as shown in Fig. 6 (Daytime) and Fig. 7 (Night).

Generally, BAF remains noisy after denoising because it only performs simple density statistics on the event stream but can preserve the edges from being damaged. Although KNoise and DWF follow the same denoising principles as BAF, they perform inferiorly in some complex structural scenes such as Fig. 6 (a). This is because they limit the memory space, resulting in a large number of valid events being filtered out rapidly owing to memory limitations and the high noise density. EvFlow performs well when the scene motion type is limited to a single object motion such as Fig. 6 (a). To some extent, YNoise and TS perform similarly, but they have distinguished denoising strategies. In detail, TS removes as much noise as possible by local plane fitting, which may damage the texture and details. In contrast, YNoise is a kind of kernel density estimation method that can preserve more structural information. However, YNoise may become invalid in some high-intensity mono-polar noise sequences such as in Fig. 6 (a); additionally, YNoise actually costs much more human labor on adjustment. As a denoising method for fast

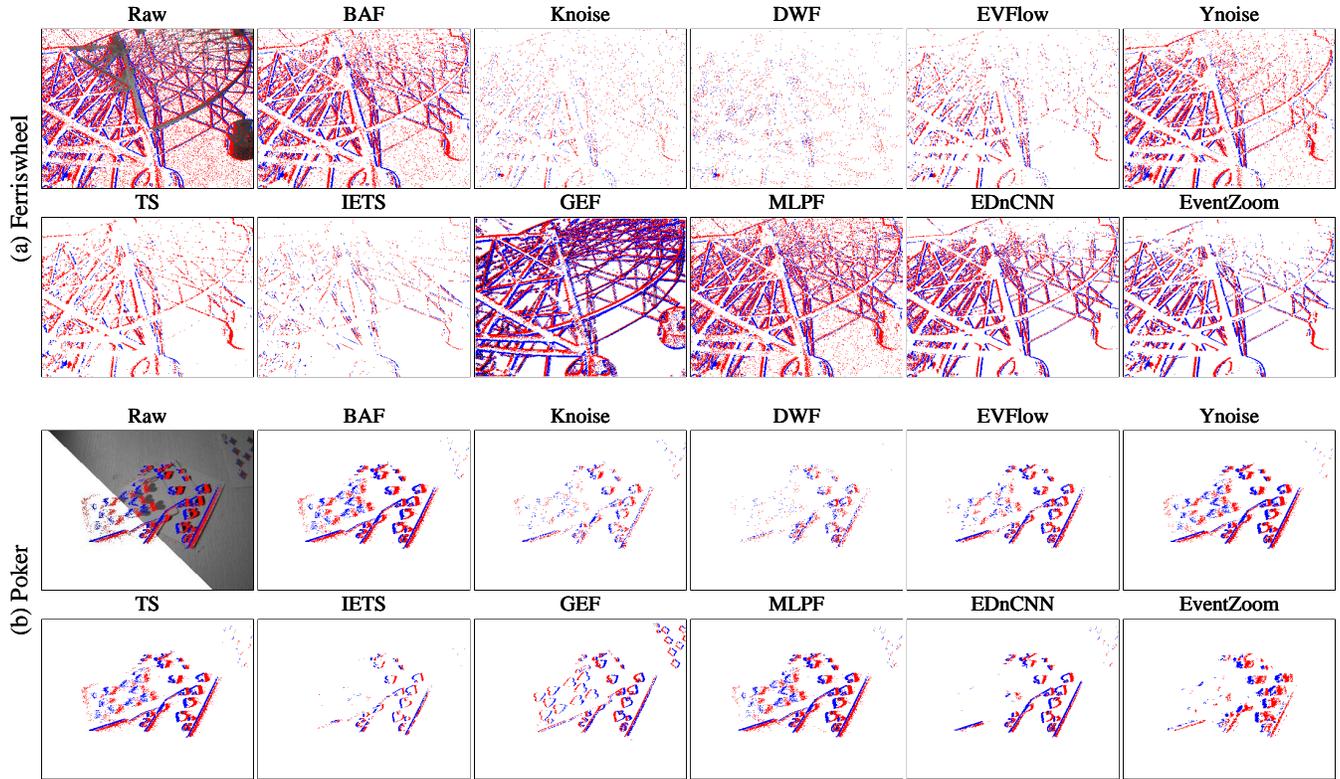


Fig. 6. The visual comparison from different denoising algorithms in some representative daytime sequences, including (a) a static object shoot against strong sunlight, in which case a lot of single polarity noise will be generated, and (b) multiple fast-moving objects in a noise-free environment, which is challenging for speed-sensitive denoisers.

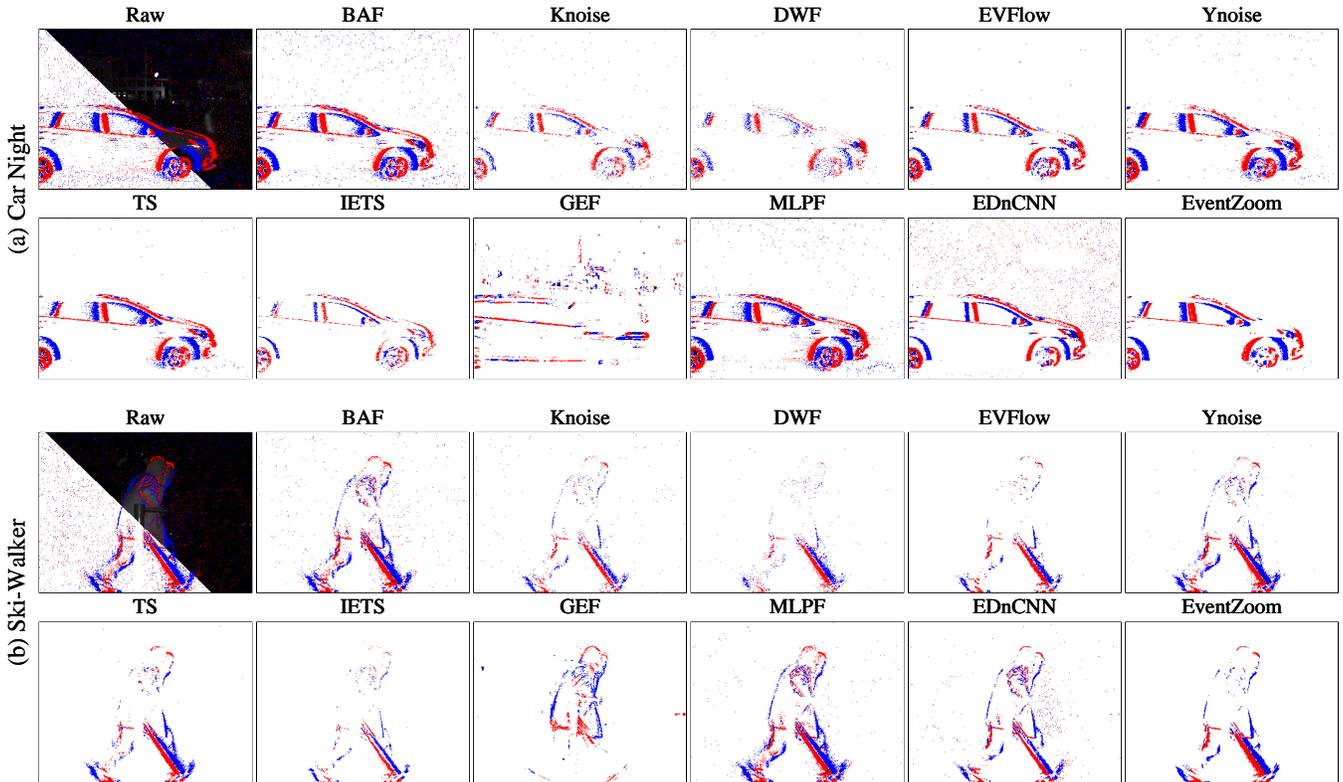


Fig. 7. The visual comparison from different denoising algorithms in some representative night sequences, including (a) a vehicle under a street light and (b) nonrigid body motion. Note that in night sequences, we have no choice but to increase exposure times as much as possible to acquire visible frames, which creates some inevitable problems such as smear or blur.



Fig. 8. Visual comparison of different denoising algorithms on multiple noise levels of the E-MLB dataset. (a)-(c) contain a cyclist who maintains the same movement at an almost consistent distance from the event camera; as the noise level increases, the edges become more blurred, and details disappear.

corner detection, IETS destroys the distribution of real events. Although it is highly suppressed in background activities, the edge of the target is no longer obvious. Benefiting from the addition of APS information, the GEF output contains sharp edges and rich texture details, as shown in Fig. 6 (a) and (b). However, when the quality of the APS image is poor, the quality of output events also decreases drastically. For example, in Fig. 7 (a), we can see that GEF cannot generate a reasonable event distribution because motion blur occurs.

For neural networks, since MLPF has a simple structure (only 2 hidden layers), it can be difficult to extract global

information, resulting in poor performance in complicated scenes such as Fig. 6 (a). However, MLPF has the lowest computational and parameter costs compared with other networks. Although EDnCNN can preserve edges well, it loses some texture information of the scene. In addition, we can see EDnCNN's weakness in dealing with object motion in Fig. 7 (a) or extreme noise environment in Fig. 7 (b). Comparatively, EventZoom has more robust performances in various sequences; however, it can cause time and pixel jittering in each event, such as in Fig. 6 (a).

Second, we present the denoising results in the same scene

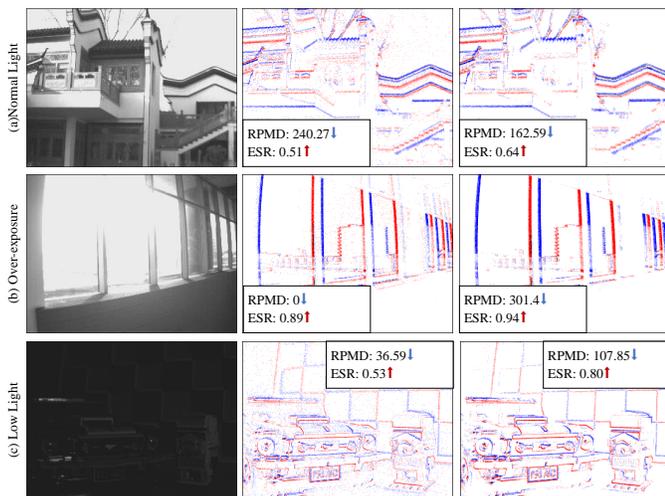


Fig. 9. The comparison of ESR with RPMD. (a) shows a normal light sequence, and both methods give reliable scores. (b)-(c) provide an overexposed and a low light sequence correspondingly, which leads to the unexpected results of RPMD, but the proposed ESR still works.

with different lighting conditions in Fig. 8. As seen in Fig. 8, the performance of all methods decreases as the noise level increases, and most fail in ND64 sequences. For BAF, NN and KNoise, their denoising sequences are contaminated as the noise level increases, but they have the least computational consumption. As GEF switches to self-guide mode due to the poor quality of the APS frames, it only performs well under moderate noise levels (ND4 and ND16). When the noise level continues to increase (ND64), GEF loses many real events. TS, IETS and YNoise outperform the other methods at medium noise levels and below in Fig. 8 (a)-(b), but IETS loses performance at extreme noise conditions in Fig. 8 (c). With regard to EDnCNN and EventZoom, each has its own merits: EDnCNN performs well in texture preservation, while EventZoom can retain more edge information. However, both of them may have undesirable performance in some high-noise scenes, specifically compared with TS and YNoise in Fig. 8 (c).

**Comparison between ESR and RPMD:** The proposed ESR in this paper is the first nonreference event denoising metric, which solves the difficulties in obtaining real event labels. In Fig. 9, we provide a comparison with another common public reference metric, RPMD. Since other methods are not suitable for evaluation on our E-MLB dataset (PSR/PNR and NIR require manual labeling, EDP and ROC require noise-free reference), we do not provide them here.

We visualize the MESR and RPMD scores on 3 representative scenes under normal light, overexposed, and low light conditions in Fig. 9. As seen, the denoised event frames look better for all the sequences by human perception. However, RPMD fails to give a better score under overexposed and low light sequences. This is because the correct calculation of RPMD requires high-quality and properly aligned APS and IMU data, which is not always met when the event camera is used in the real world. In comparison, the proposed ESR is not dependent on additional information sources and faithfully

represents the noise level under all circumstances. Overall, our metric could ignore the restriction to lighting conditions and give more reasonable scores.

## VI. DISCUSSION

In this paper, we propose a large-scale event denoising dataset E-MLB and a nonreference event denoising metric ESR for the first time. The scale of E-MLB is 12 times larger than the largest existing event-denoising dataset and rich in noise levels and scene types. The ESR represents the intrinsic property of events without needing any other information sources. With the proposed dataset and event denoising metric, we conduct extensive experiments with 11 state-of-the-art denoising methods and present a comparative analysis on event denoising.

However, there are still some limitations that need to be noted. As discussed in [41], the dominant event noise source changes from random photocurrent fluctuation to structural junction leakage current as light intensity increases. However, due to the complexity of the scene light sources, we do not discuss and classify the sources of various noise types in our proposed dataset. The proposed metric is easily affected by hot pixels, which are events emitted on some pixels at abnormally high rates. Therefore, we recommend eliminating these unexpected pixels in preprocessing. In future work, we will work on solving the above problems. We hope all these contributions can contribute to the event community to advance future research on event denoising.

## REFERENCES

- [1] G. Taverni, D. P. Moeys, C. Li, C. Cavaco, V. Motsnyi, D. S. S. Bello, and T. Delbruck, "Front and back illuminated dynamic and active pixel vision sensors comparison," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 5, pp. 677–681, 2018.
- [2] L. Wang, Y.-S. Ho, K.-J. Yoon *et al.*, "Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 081–10 090.
- [3] G. Gallego, T. Delbrück, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis *et al.*, "Event-based vision: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 1, pp. 154–180, 2020.
- [4] P. Lichtsteiner, C. Posch, and T. Delbruck, "A  $128 \times 128$  120 db  $15\mu$  s latency asynchronous temporal contrast vision sensor," *IEEE journal of solid-state circuits*, vol. 43, no. 2, pp. 566–576, 2008.
- [5] A. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Ev-flownet: Self-supervised optical flow estimation for event-based cameras," in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018.
- [6] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Unsupervised event-based learning of optical flow, depth, and egomotion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 989–997.
- [7] L. Pan, M. Liu, and R. Hartley, "Single image optical flow estimation with an event camera," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020, pp. 1669–1678.
- [8] Z. Yu, Y. Zhang, D. Liu, D. Zou, X. Chen, Y. Liu, and J. S. Ren, "Training weakly supervised video frame interpolation with events," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 589–14 598.
- [9] S. Tulyakov, D. Gehrig, S. Georgoulis, J. Erbach, M. Gehrig, Y. Li, and D. Scaramuzza, "Time lens: Event-based video frame interpolation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 155–16 164.

- [10] S. Tulyakov, A. Bochicchio, D. Gehrig, S. Georgoulis, Y. Li, and D. Scaramuzza, "Time lens++: Event-based frame interpolation with parametric non-linear flow and multi-scale fusion," *arXiv preprint arXiv:2203.17191*, 2022.
- [11] B. Kueng, E. Mueggler, G. Gallego, and D. Scaramuzza, "Low-latency visual odometry using event-based feature tracks," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 16–23.
- [12] D. Gehrig, H. Rebecq, G. Gallego, and D. Scaramuzza, "Ekl: Asynchronous photometric feature tracking using events and frames," *International Journal of Computer Vision*, vol. 128, no. 3, pp. 601–618, 2020.
- [13] G. Gallego, J. E. Lund, E. Mueggler, H. Rebecq, T. Delbruck, and D. Scaramuzza, "Event-based, 6-dof camera tracking from photometric depth maps," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 10, pp. 2402–2412, 2017.
- [14] D. Weikersdorfer, D. B. Adrian, D. Cremers, and J. Conradt, "Event-based 3d slam with a depth-augmented dynamic vision sensor," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 359–364.
- [15] H. Kim, S. Leutenegger, and A. J. Davison, "Real-time 3d reconstruction and 6-dof tracking with an event camera," in *European Conference on Computer Vision*. Springer, 2016, pp. 349–364.
- [16] 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 Robotics and Automation Letters*, vol. 2, no. 2, pp. 593–600, 2016.
- [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] X. Zhang and L. Yu, "Unifying motion deblurring and frame interpolation with events," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 765–17 774.
- [19] V. Rudnev, V. Golyanik, J. Wang, H.-P. Seidel, F. Mueller, M. Elgharib, and C. Theobalt, "Eventhands: real-time neural 3d hand pose estimation from an event stream," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 385–12 395.
- [20] D. Czech and G. Orchard, "Evaluating noise filtering for event-based asynchronous change detection image sensors," in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE, 2016, pp. 19–24.
- [21] Y. Nozaki and T. Delbruck, "Temperature and parasitic photocurrent effects in dynamic vision sensors," *IEEE Transactions on Electron Devices*, vol. 64, no. 8, pp. 3239–3245, 2017.
- [22] R. Baldwin, M. Almatrafi, V. Asari, and K. Hirakawa, "Event probability mask (epm) and event denoising convolutional neural network (edcnn) for neuromorphic cameras," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1701–1710.
- [23] P. Duan, Z. Wang, B. Shi, O. Cossairt, T. Huang, and A. Katsaggelos, "Guided event filtering: Synergy between intensity images and neuromorphic events for high performance imaging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [24] P. Duan, Z. W. Wang, X. Zhou, Y. Ma, and B. Shi, "Eventzoom: Learning to denoise and super resolve neuromorphic events," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 824–12 833.
- [25] V. Padala, A. Basu, and G. Orchard, "A noise filtering algorithm for event-based asynchronous change detection image sensors on truenorh and its implementation on truenorh," *Frontiers in neuroscience*, vol. 12, p. 118, 2018.
- [26] Y. Feng, H. Lv, H. Liu, Y. Zhang, Y. Xiao, and C. Han, "Event density based denoising method for dynamic vision sensor," *Applied Sciences*, vol. 10, no. 6, p. 2024, 2020.
- [27] S. Guo, L. Wang, X. Chen, L. Zhang, Z. Kang, and W. Xu, "Seqxfilter: A memory-efficient denoising filter for dynamic vision sensors," *arXiv preprint arXiv:2006.01687*, 2020.
- [28] T. Delbruck, "Frame-free dynamic digital vision," in *Proceedings of Intl. Symp. on Secure-Life Electronics, Advanced Electronics for Quality Life and Society*, vol. 1. Citeseer, 2008, pp. 21–26.
- [29] H. Liu, C. Brandli, C. Li, S.-C. Liu, and T. Delbruck, "Design of a spatiotemporal correlation filter for event-based sensors," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2015, pp. 722–725.
- [30] X. Lagorce, G. Orchard, F. Galluppi, B. E. Shi, and R. B. Benosman, "Hots: a hierarchy of event-based time-surfaces for pattern recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 7, pp. 1346–1359, 2016.
- [31] X. Xie, J. Du, G. Shi, J. Yang, W. Liu, and W. Li, "Dvs image noise removal using k-svd method," in *Ninth International Conference on Graphic and Image Processing (ICGIP 2017)*, vol. 10615. International Society for Optics and Photonics, 2018, p. 106153U.
- [32] A. Khodamoradi and R. Kastner, "o(n) o(n)-space spatiotemporal filter for reducing noise in neuromorphic vision sensors," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 15–23, 2018.
- [33] Z. W. Wang, P. Duan, O. Cossairt, A. Katsaggelos, T. Huang, and B. Shi, "Joint filtering of intensity images and neuromorphic events for high-resolution noise-robust imaging," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1609–1619.
- [34] R. Baldwin, M. Almatrafi, J. R. Kaufman, V. Asari, and K. Hirakawa, "Inceptive event time-surfaces for object classification using neuromorphic cameras," in *International conference on image analysis and recognition*. Springer, 2019, pp. 395–403.
- [35] E. Mueggler, C. Bartolozzi, and D. Scaramuzza, "Fast event-based corner detection," 2017.
- [36] Y. Wang, B. Du, Y. Shen, K. Wu, G. Zhao, J. Sun, and H. Wen, "Ev-gait: Event-based robust gait recognition using dynamic vision sensors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6358–6367.
- [37] J. Wu, C. Ma, X. Yu, and G. Shi, "Denoising of event-based sensors with spatial-temporal correlation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 4437–4441.
- [38] H. Kiani Galoogahi, A. Fagg, C. Huang, D. Ramanan, and S. Lucey, "Need for speed: A benchmark for higher frame rate object tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1125–1134.
- [39] S. Guo and T. Delbruck, "Low cost and latency event camera background activity denoising," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- [40] H. Rebecq, D. Gehrig, and D. Scaramuzza, "Esim: an open event camera simulator," in *Conference on robot learning*. PMLR, 2018, pp. 969–982.
- [41] Y. Hu, S.-C. Liu, and T. Delbruck, "√2e: From video frames to realistic dvs events," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1312–1321.
- [42] J. Wu, C. Ma, L. Li, W. Dong, and G. Shi, "Probabilistic undirected graph based denoising method for dynamic vision sensor," *IEEE Transactions on Multimedia*, vol. 23, pp. 1148–1159, 2020.
- [43] S. Guo, Z. Kang, L. Wang, S. Li, and W. Xu, "Hashheat: An o(c) complexity hashing-based filter for dynamic vision sensor," in *2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2020.
- [44] J. Xu, J. Zou, S. Yan, and Z. Gao, "Effective target binarization method for linear timed address-event vision system," *Optical Engineering*, vol. 55, no. 6, p. 063103, 2016.
- [45] S. Afshar, N. Ralph, Y. Xu, J. Tapson, A. v. Schaik, and G. Cohen, "Event-based feature extraction using adaptive selection thresholds," *Sensors*, vol. 20, no. 6, p. 1600, 2020.
- [46] S. Li, Y. Feng, Y. Li, Y. Jiang, C. Zou, and Y. Gao, "Event stream super-resolution via spatiotemporal constraint learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4480–4489.
- [47] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3857–3866.
- [48] J. Han, C. Zhou, P. Duan, Y. Tang, C. Xu, C. Xu, T. Huang, and B. Shi, "Neuromorphic camera guided high dynamic range imaging," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1730–1739.
- [49] S. Lin, Y. Zhang, L. Yu, B. Zhou, X. Luo, and J. Pan, "Autofocus for event cameras," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 344–16 353.
- [50] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3867–3876.
- [51] G. Gallego, M. Gehrig, and D. Scaramuzza, "Focus is all you need: Loss functions for event-based vision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 280–12 289.
- [52] J. Chen, Y. Wang, Y. Cao, F. Wu, and Z.-J. Zha, "Progressivemotionseg: Mutually reinforced framework for event-based motion segmentation," *arXiv preprint arXiv:2203.11732*, 2022.