



A motion denoising algorithm with Gaussian self-adjusting threshold for event camera

Wanmin Lin¹ · Yuhui Li¹ · Chen Xu² · Lilin Liu¹

Accepted: 11 November 2023 / Published online: 18 January 2024
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2024

Abstract

Event cameras, characterized by their low power consumption, expansive dynamic range, and high temporal resolution, have attracted great attentions in various computer vision tasks. Compared to frame-based cameras, event cameras exemplify a marked paradigmatic transition in data formation and output. However, the quality of event streams is compromised by background activity and hot pixels, leading to increased computational overheads and sub-optimal outcomes in subsequent applications, notably in recognition, video reconstruction, and target detection tasks. In this paper, a two-step denoising algorithm (referred as GMCM) is proposed to counteract these challenges. The GMCM algorithm comprises two steps: Gaussian denoising preprocessing and motion denoising. The former incorporates Gaussian temporal distribution and adaptive thresholding mechanisms to discern the inclusion of motion-related information within the event streams. Experimental results demonstrate that Gaussian denoising preprocessing can not only adeptly discern whether the event data stream contains motion information but also enhance computational efficiency. Conclusively, the GMCM algorithm achieves state-of-the-art performance, yielding SNR scores of 37.22 and 26.79 on the DVSCLEAN dataset at the noise ratios of 50% and 100%, respectively.

Keywords Event camera · Denoising · Gaussian time distribution · Motion estimation

List of symbols

SOTA	State-of-the-art	NS-Max	The Gaussian maximum value of noise stream
GMCM	A two-step denoising algorithm comprising Gaussian denoising preprocessing and motion denoising	EN	The number of real events remaining after denoising
CM	Contrast maximization algorithm	NN	The number of noise events remaining after denoising
MCM	An enhanced CM algorithm without Gaussian denoising preprocessing	t	The timestamp of each event
BA	Background activity	μ	The average timestamp of event sequence
Motion Stream	Event streams with motion information	σ	The timestamp variance of event sequence
Noise Stream	Event streams without motion information	Event Num	The number of events in a basic processing unit
MS-Max	The Gaussian maximum value of motion stream	ρ	The adaptive threshold
		x'_k	The x coordinate of the event after motion projection
		y'_k	The y coordinate of the event after motion projection
		x_k	The original x coordinate of the event
		y_k	The original y coordinate of the event
		t_k	The original timestamp
		v_x	Movement speed in x direction
		v_y	Movement speed in y direction
		λ_x	The nonlinear parameters in x direction

✉ Lilin Liu
liullin@mail.sysu.edu.cn

¹ The State Key Lab of Optoelectronic Materials and Technology, School of Electronics and Information Technology, Sun Yat Sen University, Xin'gang West Road, 135#, Guangzhou 510275, Guangdong, China

² The Pazhou Lab (Huangpu), Huangpu District, Guangzhou, Guangdong, China

λ_y	The nonlinear parameters in y direction
t_{ref}	The reference time which is the timestamp of the last event
w_x	The angular velocity of the rotational motion in x direction
w_y	The angular velocity of the rotational motion in y direction
\otimes	The multiplication of the corresponding position
ζ	Nonlinear terms in rotational motion
H	Reference time plane
θ	The motion trajectory
R	The reward function
SNR	Signal-to-noise ratio

1 Introduction

Event cameras, as an asynchronous vision sensor, hold substantial promise for groundbreaking advancements in fields like autonomous driving, robotics, and surveillance [1–5]. In contrast to frame-based cameras, which operate with fixed exposure durations, event cameras capture luminance changes individually at each pixel [6, 7]. The resultant output manifests as a pixel-specific event stream that directly encapsulates changes in scene reflectance, thereby mitigating data redundancy and preserving precise temporal data [1, 8, 9]. Therefore, merits of the event camera include high temporal resolution (microsecond order), low power consumption (milliwatt), and high dynamic range (~ 120 dB) [1, 10]. Exploiting these salient attributes, event cameras proficiently capture scene details, even in challenging environments such as poor lighting and high-speed motion, without necessitating extensive storage resources.

In theory, event cameras are designed to produce responses and relay event data exclusively in response to changes in light intensity or the presence of object motion [1, 4]. However, without log intensity changes, or even without any scene activity, spurious events could be inadvertently generated by the camera [8, 11, 12]. Such aberrant occurrences can critically compromise the quality of event data, potentially undermining the utility of event cameras for subsequent tasks, including optical flow, target detection, and object recognition. Therefore, it is crucial to remove these spurious event data. However, prior to distilling meaningful event data amidst such noise, certain considerations must be taken into account. For instance, in static scenarios devoid of motion or brightness fluctuations, is it necessary to allocate significant computational resources to process the false events induced by inherent camera hardware limitations? Additionally, in dynamic scenarios, what strategies can be effectively employed to mitigate the pervasive noise?

To address these issues, we propose a two-step denoising algorithm (referred as GMCM), composed of robust Gaussian denoising preprocessing and motion denoising. In the first step, the Gaussian denoising preprocessing employs a Gaussian time distribution coupled with adaptive thresholds, thereby preliminarily ascertaining the presence or absence of motion information within the event stream. If no motion information is present, the corresponding data are promptly discarded. Conversely, discernible motion cues prompt the activation of the subsequent denoising step. Here, event streams with and without motion information are referred as Motion Stream and Noise Stream, respectively. Experimental results demonstrate that our GMCM algorithm effectively distinguishes Motion Stream and Noise Stream. Furthermore, compared to the conventional contrast maximization (CM) algorithm, GMCM algorithm can more effectively distinguish effective events from noise events, resulting in an enhanced denoising outcome. In addition, compared to other SOTA denoising algorithms, GMCM algorithm achieves the best denoising performance on benchmark datasets such as DVSNOISE20 [13] and DVSCLEAN [14].

In general, our contributions are as follows:

1. A novel two-step denoising algorithm, consisting of Gaussian denoising preprocessing and motion denoising, is proposed.
2. To elevate computational efficiency, it is for the first time that Gaussian denoising preprocessing is introduced to preliminarily ascertain whether the event data stream encompasses motion information.
3. To expand the application range of the motion denoising algorithm and improve its robustness, nonlinear terms are incorporated into the motion equation of the motion denoising model.

2 Related works

In this section, the types of noise events and previous researches on event denoising algorithms are mainly reviewed.

2.1 Noise events

In event cameras, the primary sources of noise events can be attributed to background activity (BA) and hot pixels [11, 15–21]. BA refers to spurious signals emitted in the absence of notable shifts in luminance or motion. Such events may be triggered by the factors such as charge injection, leakage from the reset switching transistor section, and the thermal noise [12, 17–22]. Compared to real events, which exhibit higher spatiotemporal correlation, BAs occur randomly and

with lower frequency [17–20]. As a result, in event streams, the distribution of real events tends to be denser, while the distribution of BA is sparser. In addition, hot pixels in event cameras are often the result of inadequate pixel resetting processes, the same as in the frame-based camera [7, 23, 24].

2.2 Event denoising algorithm

Given that event cameras produce asynchronous event streams rather than standard frames, traditional frame-based denoising algorithms cannot be applied directly. As a result, some researchers have proposed a strategy where the sparse event data was firstly mapped to dense image frames, subsequently undergoing traditional denoising methods [25, 26]. However, this approach worked at the expense of temporal information. Moreover, the mapping of event streams to image frames is irreversible. In other words, the denoised event stream cannot be recovered from the denoised image frames later. Here, in our proposed algorithm, the motion information of the event stream is obtained and employed for denoising without the need for translation from event data to event frames. Furthermore, our denoised output is retained as an event stream, facilitating subsequent analyses to directly leverage the intrinsic benefits of the event stream eigenvalue.

To harness the temporal information and asynchronous sparsity inherent in event streams, several denoising algorithms employing filtering techniques have been proposed, such as temporal filtering [27], spatial filtering [28], and spatiotemporal filtering [16, 17, 29, 30]. Baldwin et al. [27] introduced a temporal filter using a time-surface approach, aiming to eliminate redundant events and noise. Wang et al. [28] proposed a spatial filter grounded in the notion that events triggered by the same edge motion exhibit similar spatiotemporal motion projections. Liu et al. [31] established a spatiotemporal correlation-based filter by considering the temporal correlation of event groups partitioned by subsampling. These aforementioned filters are primarily suitable for denoising simple scenes and may struggle to differentiate between noise and genuine events in high dynamic or high noise ratio scenes.

Recently, deep learning has gained significant attention in the denoising of event cameras, particularly for complex scenes [13, 18, 32, 33]. Baldwin et al. [13] utilized the EPM tags, which are consisted of APS, IMU, and camera parameters, to convert neighboring events into multiple time surfaces which were then fed into the convolutional neural network, EDnCNN, for denoising. Duan et al. [18] proposed EventZoom, a 3D U-Net neural network, aimed at both event denoising and super-resolution. Although EventZoom employed a noise-to-noise strategy, it did not incorporate event-to-image modules, which introduced the concept of frames to some extent [18]. In a word, existing neural network-based denoising algorithms often require preprocessing steps that

deviate from the intrinsic properties of event data. Extracting accurate features for matching the underlying information of events which is challenging due to their asynchronous and sparse nature, thereby affecting denoising performance. Moreover, the scarcity of real-world datasets poses a significant obstacle for the widespread adoption of neural network-based denoising algorithms. In contrast, our GMCM algorithm does not rely on vast datasets or annotated samples for learning. Additionally, the Gaussian denoising preprocessing step integrated within the GMCM algorithm aids in optimizing computational efficiency.

3 Proposed methods

In this section, a two-step GMCM algorithm is proposed for event denoising, as shown in Fig. 1. The first step focuses on denoising preprocessing, utilizing a Gaussian time distribution and an adaptive threshold. The second step engages in motion denoising by utilizing an enhanced contrast maximization technique.

3.1 First step: Gaussian denoising preprocessing

In the GMCM algorithm, event streams are first segmented into temporal bins, acting as the fundamental processing units. Conventionally, the events are mapped to representations using either a constant time window or a fixed number of events. In this case, a fixed number of events is selected as the basic processing unit. The rationale for this choice is elaborated subsequently.

According to Section II.A, noise events occur randomly with a low probability, whereas real events caused by motion occur in a large number within a short period of time. As a result, compared to the Noise Stream, the temporal distribution of the Motion Stream is more compact. Furthermore, in the processing units with a constant number of events, the Gaussian time distribution of the Motion Stream is more concentrated than that of the Noise Stream. (The equation for Gaussian time distribution is presented in Eq. 1.) However, with fixed time window units, due to the constant timeframe, the Gaussian time distributions of both Noise Stream and Motion Stream tend to be analogous, complicating the differentiation of the presence of motion information within the event stream (as illustrated in Fig. 2). To address this, the Gaussian maximum probability density value, specifically MS-Max versus NS-Max, is employed to discern between the Motion Stream and Noise Stream.

$$\text{Gaussian}(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right) \quad (1)$$

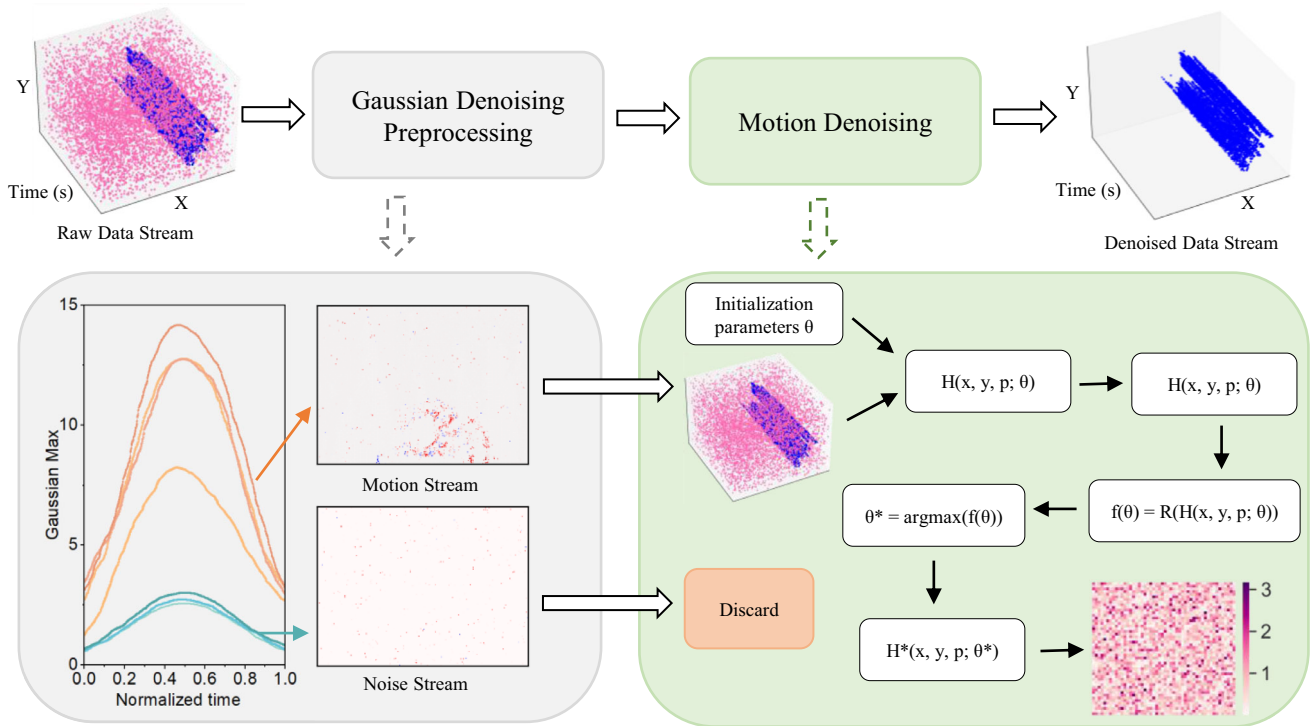


Fig. 1 Illustration is the framework for the proposed two-step GMCM algorithm. The input consists of sparse events, asynchronously generated by event cameras. Stage (I) Gaussian Denoising Preprocessing: first, all data are uniformly segmented into N groups of events. Then, the Gaussian time distribution for each event group is deduced. Finally, by contrasting the derived Gaussian maximum value with the adaptive threshold, a determination is made regarding whether it constitutes a

Motion Stream or a Noise Stream. Stage (II) Motion Denoise: first, the motion trajectory is estimated by the modified contrast maximization algorithm. Then, based on the optimal motion trajectory, the event number obtained by each pixel is calculated. If the event number is below a certain threshold, it is noise. The final output is the event stream that has been denoised. Blue/Pink dot: Real/Noise event

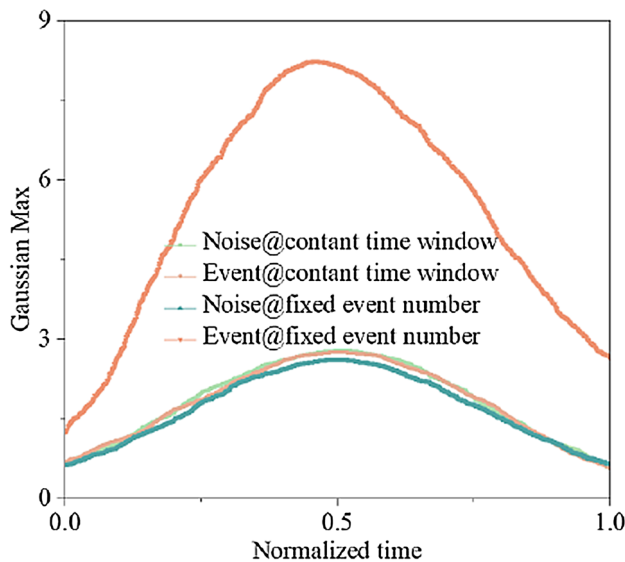


Fig. 2 Comparison of Gaussian time distribution based on constant time window and constant number of events

where, t is the timestamp of each event, μ is the average timestamp of this series of events, and σ is the variance of timestamps in this event series.

However, varying event counts can influence the Gaussian threshold for distinguishing the Noise Stream and Motion Stream. To circumvent the manual recalibration necessitated by alterations in event counts, an adaptive threshold method is proposed to distinguish between the Motion Stream and Noise Stream (as shown in Eq. 2).

$$\rho = \exp(a + bn + cn^2) + \epsilon \tag{2}$$

where n is the number of events in a basic processing unit (referred as Event Num); a , b , and c are related to the material and the motion of the object; ϵ is related to the light intensity.

3.2 Second step: Motion denoising

Contrast maximization (CM), a prevalent technique for object trajectory estimation, has demonstrated efficacy in applications such as target detection, feature tracking, and motion compensation [34–37]. Researchers optimize the CM

algorithm by minimizing the loss function [34, 35, 38]. However, it is also important to refine the trajectory equation.

3.2.1 Linear motion

Currently, the CM algorithm is grounded on linear motion equations, which postulates that an object moves at a constant velocity within a very short time interval. However, these equations may not suffice in accurately delineating complex linear motions, such as accelerated motion. To mitigate this limitation, the introduction of nonlinear terms to the CM motion equations is proposed. This modification seeks to accommodate intricate linear motions and permits the employment of arbitrarily chosen time intervals, thereby augmenting the algorithm’s versatility and precision in portraying a variety of motion patterns:

$$\begin{pmatrix} x'_k \\ y'_k \end{pmatrix} = \begin{pmatrix} x_k \\ y_k \end{pmatrix} - (t_k - t_{ref}) \begin{pmatrix} v_x \\ v_y \end{pmatrix} - (t_k - t_{ref})^2 \begin{pmatrix} \lambda_x \\ \lambda_y \end{pmatrix} \tag{3}$$

where, x'_k, y'_k are the event coordinates after motion projection; x_k, y_k, t_k are the original coordinates and timestamp, respectively; v_x, v_y are the motion speeds; λ_x, λ_y are the nonlinear parameters; t_{ref} is the reference time, which is the timestamp of the last event.

3.2.2 Rotational motion

The rotational motion model of the CM method is specifically designed for scenarios typified by pure rotational motions with constant angular velocities. However, in situations where these velocities subject to unpredictable fluctuations intrinsic to complex scenarios, the model encounters palpable challenges. Many naturally occurring rotational dynamics, particularly those swayed by external perturbations or intrinsic nonlinear forces, manifest characteristics which challenge the descriptive capacity of a strictly linear model. Building on the foundational work of Gallego et al. [34] regarding the motion equation, we propose the incorporation of a nonlinear component to aptly represent these nuanced motion behaviors:

$$\zeta = \left[(t_k - t_{ref}) \begin{pmatrix} v_x \\ v_y \end{pmatrix} + (t_k - t_{ref})^2 \begin{pmatrix} v'_x \\ v'_y \end{pmatrix} \right] * \begin{pmatrix} \lambda_x \\ \lambda_y \end{pmatrix} \tag{4}$$

$$\begin{pmatrix} x'_k \\ y'_k \end{pmatrix} = \begin{pmatrix} x_k \\ y_k \end{pmatrix} - (t_k - t_{ref}) \begin{pmatrix} w_x \\ w_y \end{pmatrix} \otimes \begin{pmatrix} x_k \\ y_k \end{pmatrix} - \zeta \tag{5}$$

where w_x, w_y represent the angular velocity of the rotational motion, \otimes represents the multiplication of the corresponding position, and ζ is a nonlinear term. Subsequent experimental evaluations, exemplified by experiments with rolling spheres,

substantiate that this modification yields a more faithful portrayal of rotational dynamics, especially those experiencing variable angular velocities. This innovation effectively augments the versatility of the CM algorithm in addressing a broader spectrum of rotational motion scenarios.

3.2.3 Motion trajectory estimation

Accurately estimating the motion trajectory is crucial for subsequent processing steps. In this approach, the motion trajectory is firstly initialized, following which events are mapped onto a reference time plane based on this trajectory. Thus, the events are aligned in a consistent temporal reference frame, facilitating ensuing analysis and processing:

$$H(x_k, y_k, p_k; \theta) = \sum_{k=1}^m p_k \delta((x_k, y_k) - (x'_k, y'_k)) \tag{6}$$

$$\theta = \begin{cases} (v_x, v_y, \lambda_x, \lambda_y) \\ (w_x, w_y, v_x \cdot \lambda_x, v_y \cdot \lambda_y, v'_x \cdot \lambda_x, v'_y \cdot \lambda_y) \end{cases} \tag{7}$$

where δ is the Dirac function, θ is the motion trajectory.

Subsequently, the reference time plane is processed via a reward function. Stoffregen et al. [35] categorized the reward function utilized in the CM algorithm into the sparsity reward and magnitude reward. The former excels in dealing with image uncertainty, while the latter demonstrates greater tolerance to noise. To harness the advantages of both sparsity reward and magnitude reward, a hybrid reward function was deemed beneficial. In our approach, the hybrid reward function, denoted as R , is directly applied to process the reference time plane, aiming to obtain the optimal trajectory. The equations for the reward function and the contrast maximization are delineated below:

$$R(H) = \sum_{x,y} h(x, y)^2 + \sum_{x,y} \exp(-h(x, y) * p_k) \tag{8}$$

$$\theta^* = \arg \max_{\theta} R(H(x_k, y_k, p_k; \theta)) \tag{9}$$

3.2.4 Distinguishing between real and noise events

To distinguish events and noise, event streams are projected onto the reference time plane using accurate motion estimates θ^* , and the quantity is enumerated at each pixel location on the time plane:

$$H(x_k, y_k, p_k; \theta^*) = \begin{cases} 0 & \text{None} \\ 1 & \text{Noise} \\ \text{otherwise} & \text{Event} \end{cases} \tag{10}$$

In this process, pixel locations manifesting a higher event count are indicative of actual real events, while those exhibiting a lower event count are likely to be associated with noise or spurious events. By establishing an appropriate threshold, a demarcation between actual events and noise can be effected based on the event count at each pixel location on the reprojected time plane.

4 Experiments and results

In this section, real-world scenes are captured by an event camera (DAVIS346). The data obtained is then processed with the proposed two-step GMCM algorithm to evaluate its applicability. Furthermore, for a comprehensive performance comparison of our GMCM algorithm with other denoising methods, both the DENOISE20 and DVSCLEAN benchmark dataset are employed. Our algorithm, along with other conventional algorithms, is operationalized on a CPU (Intel i7-12700F, 64-bit). Conversely, the denoising models based on neural networks operate on a GPU (GeForce RTX 3080Ti GPU), utilizing a software framework inclusive of PyTorch 1.13.0 and CUDA Toolkit 11.3.1. The encompassing software milieu was constructed with Python 3.9.12 and PyCharm 2022.2.

4.1 Dataset

4.1.1 Experimental setup of our dataset

During the experimental phase, we utilized the DAVIS346 camera, which boasts a spatial resolution of 346×260 pixels. The camera was securely mounted on a stand, maintaining a distance of 60 cm from the object in motion. We captured three distinct indoor motion scenarios, each involving objects with varying refractive indices. Each object was recorded under three different lighting conditions, with each condition being replicated thrice. An exhaustive description of the experimental conditions is provided in Table 1.

4.1.2 Benchmark datasets

DVSNOISE20 [13] is a genuine dataset that mirrors real-world scenarios, embracing 16 distinctive indoor and outdoor environments. Each of these locations has been carefully documented three times, each capture lasting approximately 16 seconds, culminating in a total of 48 sequences that cover a wide range of movements.

Regarding DVSCLEAN [14], our primary focus is on the simulated section of the dataset. This simulated portion is generated by utilizing the ESIM [39] technique to convert real scene videos into simulated real-world events, after which artificial random noise is intentionally added. The dataset

Table 1 Experimental parameters about object material, movement, and light intensity

Object	Material	Movement	Lighting
Statue	Diffuse reflection	Linear motion	Low lighting Normal lighting Over exposure
Metal iron	High reflective	Linear motion	Low lighting Normal lighting Over exposure
Basketball	Diffuse reflection	Roll	Low lighting Normal lighting Over exposure

houses 49 distinct scenes. Within each scene, noise is introduced at two separate magnitudes: one corresponding to 50% of the simulated real event frequency, and the other matching 100% of that frequency. To stay consistent with the terminology used by the primary researchers, these are denominated as the 50% noise ratio and the 100% noise ratio, respectively.

4.2 Impact of the event num on the GMCM algorithm

The Event Num is a key factor affecting the denoising performance of the GMCM algorithm. Here, a series of GMCM algorithms, each accommodating a different Event Num, are applied to perform denoising processes on the DVSCLEAN benchmark dataset, and the corresponding SNR scores are calculated, as depicted in the Fig. 3. It reveals that the GMCM algorithm attains approximately the highest SNR scores when the Event Num is between 10,000 and 20,000. Moreover, with an increase in the Event Num, the SNR scores experience only a slight decline. This demonstrates that our denoising algorithm exhibits robustness with respect to the number of events. Furthermore, unless otherwise specified, in subsequent experiments, we will employ the GMCM algorithm with an event number of 10,000 and compare it to other advanced denoising algorithms (Table 2).

4.3 First step of GMCM: Gaussian denoising preprocessing

First, all data are impartially segregated into distinct groups of event streams, predicated on the pre-established Event Num values. As shown in Fig. 4, for each Event Num, the MS-Max surpasses NS-Max. Concomitant with the augmentation of Event Num, both MS-Max and NS-Max exhibit a gradual diminution. Furthermore, at identical Event Num,

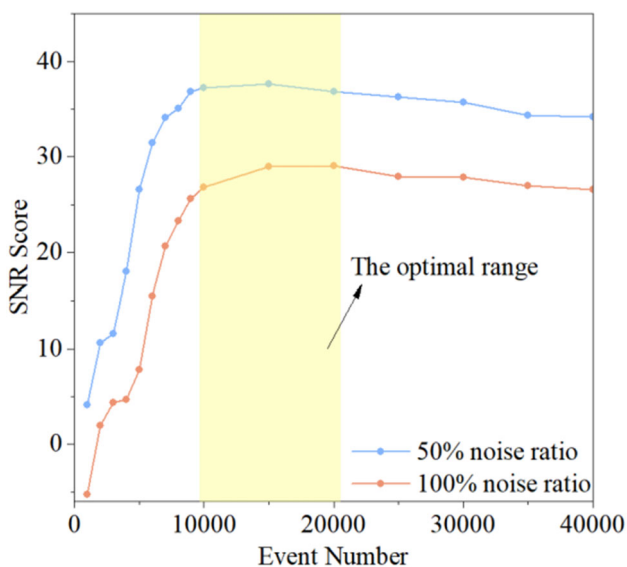


Fig. 3 The SNR scores of the GMCM Algorithm Under Different Event Num

Table 2 Comparison of SNR scores on DVSCLEAN dataset

	50% noise ratio	100% noise ratio	Average
STP [40]	20.34	14.53	17.44
Liu et al. [31]	23.80	18.70	21.25
ICM [22]	21.64	15.68	18.66
EDnCNN [13]	24.75	18.80	21.78
AEDNet [14]	26.11	25.08	25.60
GMCM	37.22	26.79	32.01

MS-Max and NS-Max manifest discrepancies under varying illumination conditions, namely low, normal, and overexposure. Additionally, different objects, such as diffuse objects and highly reflective objects (e.g., Statue and Metal Iron), alongside varying motion paradigms (e.g., linear motion and rolling as observed in Statue and Basketball), impart distinct impacts on MS-Max and NS-Max. The experimental results demonstrate that the proposed adaptive threshold effectively incorporates the impacts of object material, motion modality, light intensity, and Event Num on MS-Max and NS-Max. The adaptive threshold can successfully distinguish Motion Stream and Noise Stream.

4.4 Second step of GMCM: Motion denoising

Comparisons between the GMCM and CM algorithms are performed to evaluate their denoising performances in terms of motion, lighting scenes, and object materials. As shown in Figs. 5, 6, 7, for low light, normal light, and overexposure scenes, although the CM algorithm effectively eliminates

noise in the event stream generated by the linear motion of both diffuse reflection objects and highly reflective objects, it also eliminates a significant large number of valid events. In addition, subsequent to processing via the CM algorithm, a considerable amount of noise still remains in the events generated by the rolling. However, regardless of the type of scene, our GMCM algorithm manifests superior capability, mitigating extensive noise and concurrently retaining a more considerable number of valid events.

Moreover, the denoising performance of our GMCM algorithm demonstrates diminished sensitivity to Event Num. As shown in Fig. 8, with the reduction of Event Num from 30,000 to 10,000, the CM algorithm eradicates numerous valid events. In contrast, the GMCM algorithm still well preserves of valid events. These results indicate that the GMCM algorithm exhibits enhanced stability and robustness, making it more reliable in denoising event data across different event densities.

4.5 Comparison with other denoising algorithms

Leveraging the DVSNOISE20 dataset [13], the denoising performance of the proposed GMCM algorithm is compared to other algorithms, including the spatiotemporal correlation filter [31], Ev-gait [32], GEF [33], EDnCNN [13], and EventZoom [18]. The DVSNOISE20 dataset is a benchmark dataset, encompassing 16 disparate scenes captured with minimal camera movement and static objects. It is important to acknowledge that, due to the unavailability of the source code for the majority of denoising algorithms, we refer to the data graph from ref. [18]. Although DVSNOISE20 dataset was collected by DAVIS346, whose output size is 346 × 260, ref. [18] modified its dimensions.

Consequently, to maintain consistency in comparison, our images preserve the original camera dimensions.

Both our GMCM algorithm and the spatial-temporal correlation-based filter algorithm[31] are entrenched in model-based denoising methods. When contrasted with the filter algorithm, our GMCM algorithm can more effectively distinguish noise and real events (as shown in Fig. 9). Additionally, compared to the neural network-based denoising algorithms, our GMCM algorithm surpasses several, including GEF and EV-gait (as shown in Fig. 10), and it is slightly inferior to state-of-the-art methods such as EDnCNN and EventZoom. However, neural network-based denoising algorithms necessitate substantial volumes of labeled data for training—a prerequisite often elusive for event datasets, particularly within intricate scenes. Therefore, our GMCM algorithm emerges as a prospective alternative in the domain of event denoising, mitigating the aforementioned challenges.

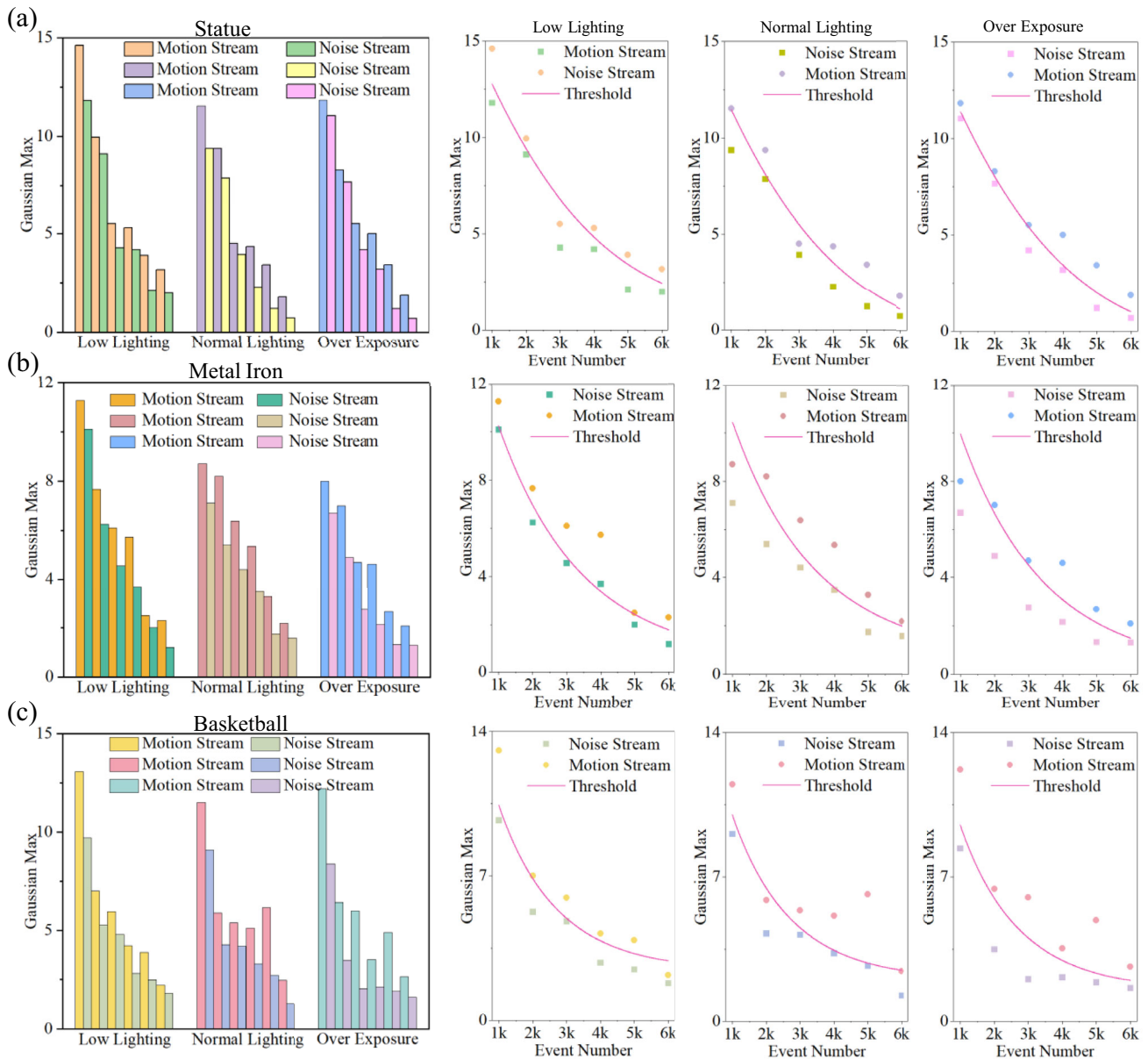


Fig. 4 Histograms and curves showing the Gaussian maximum value and threshold value under different object materials, motion modes, light brightness, and Event Numbers. In addition, the Gaussian maximum value of Noise Stream (abbreviated as NS-Max) refers to the

largest Gaussian maximum value in all Noise Streams; the Gaussian maximum value of Motion Stream (abbreviated as MS-Max) refers to the smallest Gaussian maximum value in all Motion Streams

To quantitatively evaluate the denoising performance, we employ the DVSCLEAN dataset [14] which provides distinct labels for the simulated data. Here, the index of SNR proposed in ref [14] is adopted to evaluate the performance of denoising. The SNR for benchmarking denoising performance is described as:

$$\text{SNR} = 20 \cdot \log_{10}(\text{EN}/\text{NN}) \quad (11)$$

where EN and NN refer to the number of real events and noise events remaining after denoising, respectively. As shown in

Eq. 11, a higher SNR value denotes superior denoising efficacy. Compared to other SOTA denoising algorithms, such as EDnCNN [13], AEDNet [14], ICM [22], Liu et al. [31], and STP [40], our GMCM algorithm achieves the highest SNR scores at both 50% and 100% noise ratio. Specifically, at a 50% noise ratio, our GMCM algorithm achieves the SNR score of 37.22, and at a 100% noise ratio, it achieves the SNR score of 26.79. These outcomes underline the proficiency of our GMCM algorithm in discriminating between noise and valid events, even in environments saturated with substantial noise levels within the experimental spectrum.

Fig. 5 Raw event data, the CM denoised, and the GMCM denoised event data under low illumination

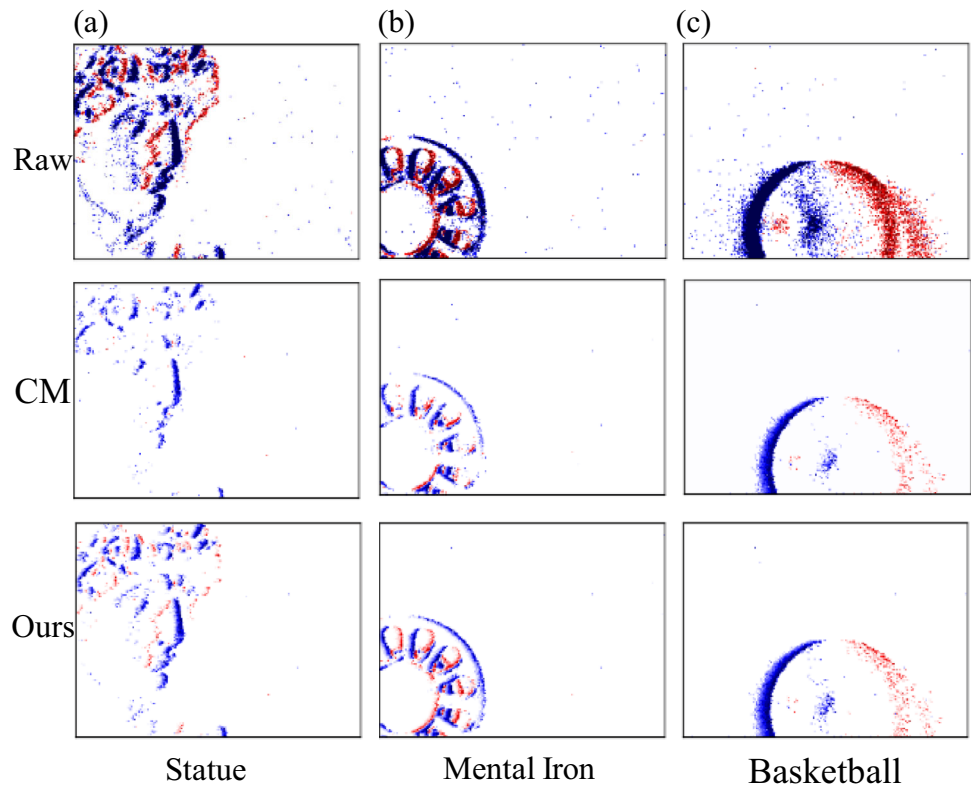


Fig. 6 Raw event data, the CM denoised, and the GMCM denoised event data under normal illumination

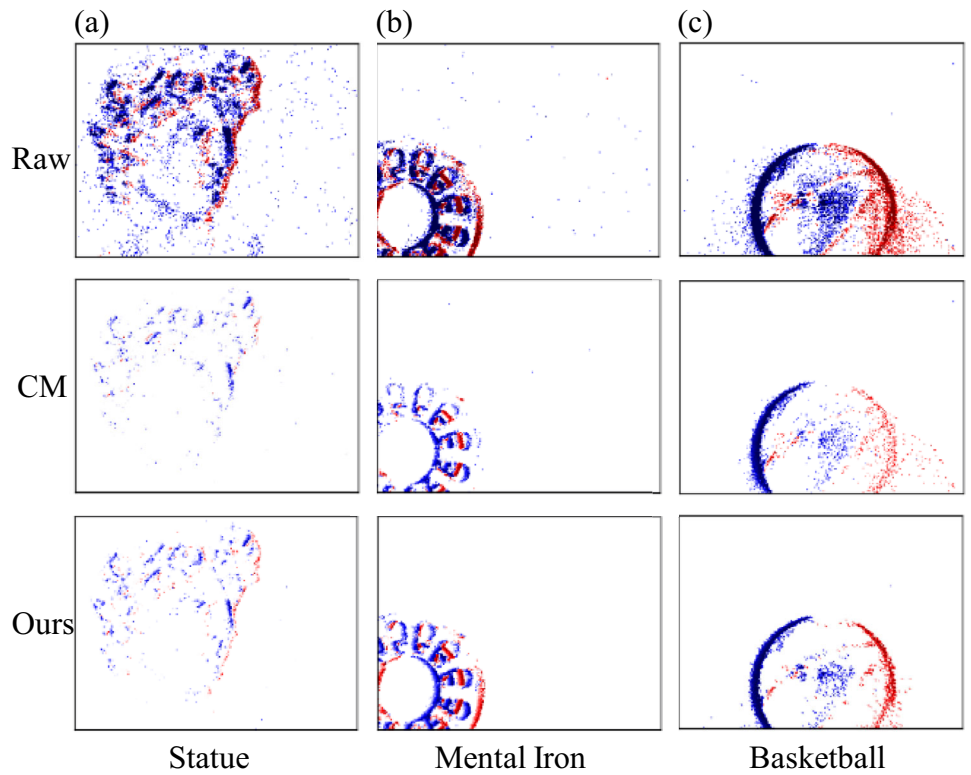


Fig. 7 Raw event data, the CM denoised, and the GMCM denoised event data under overexposure

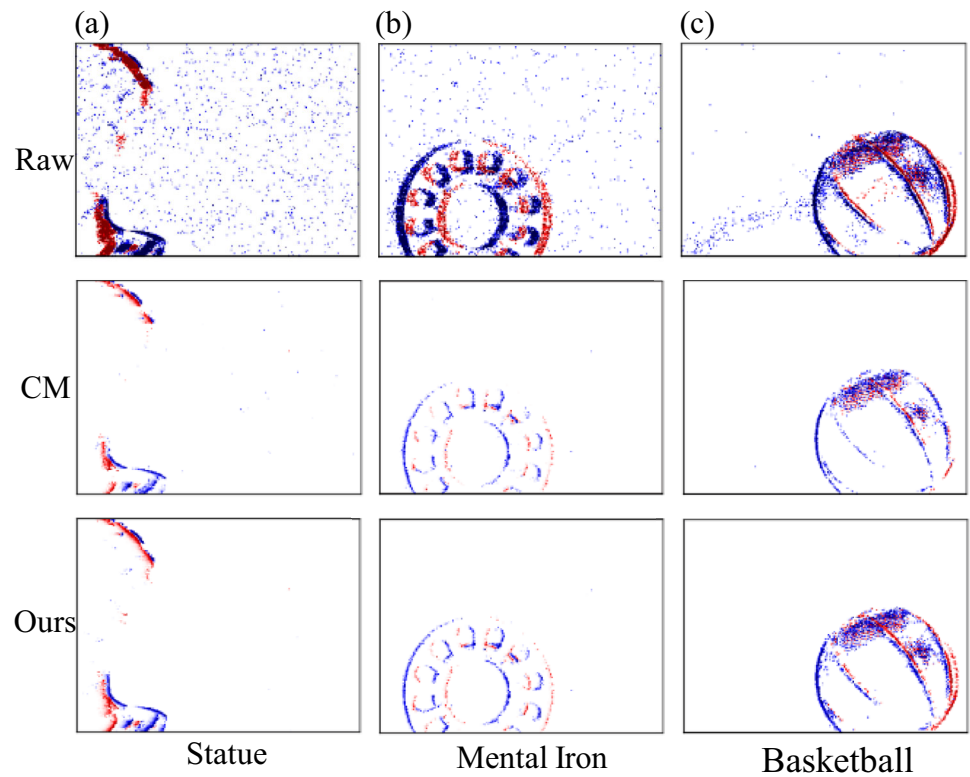
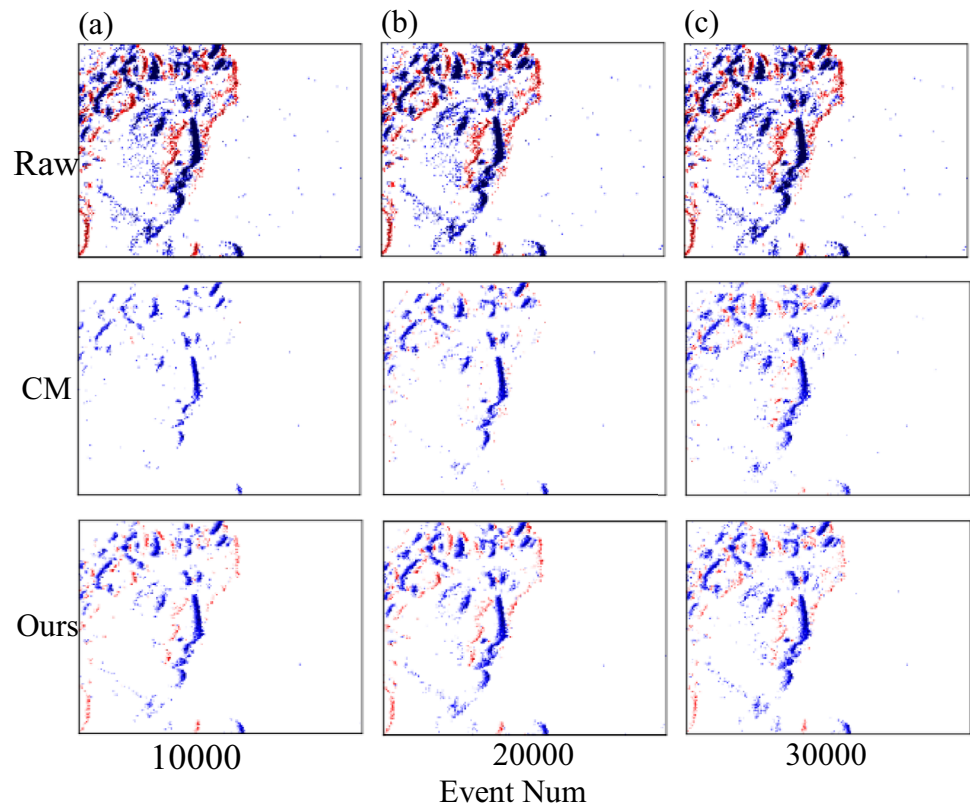


Fig. 8 Raw event data, the CM denoised, and the GMCM denoised CM denoised event data at different Event Num



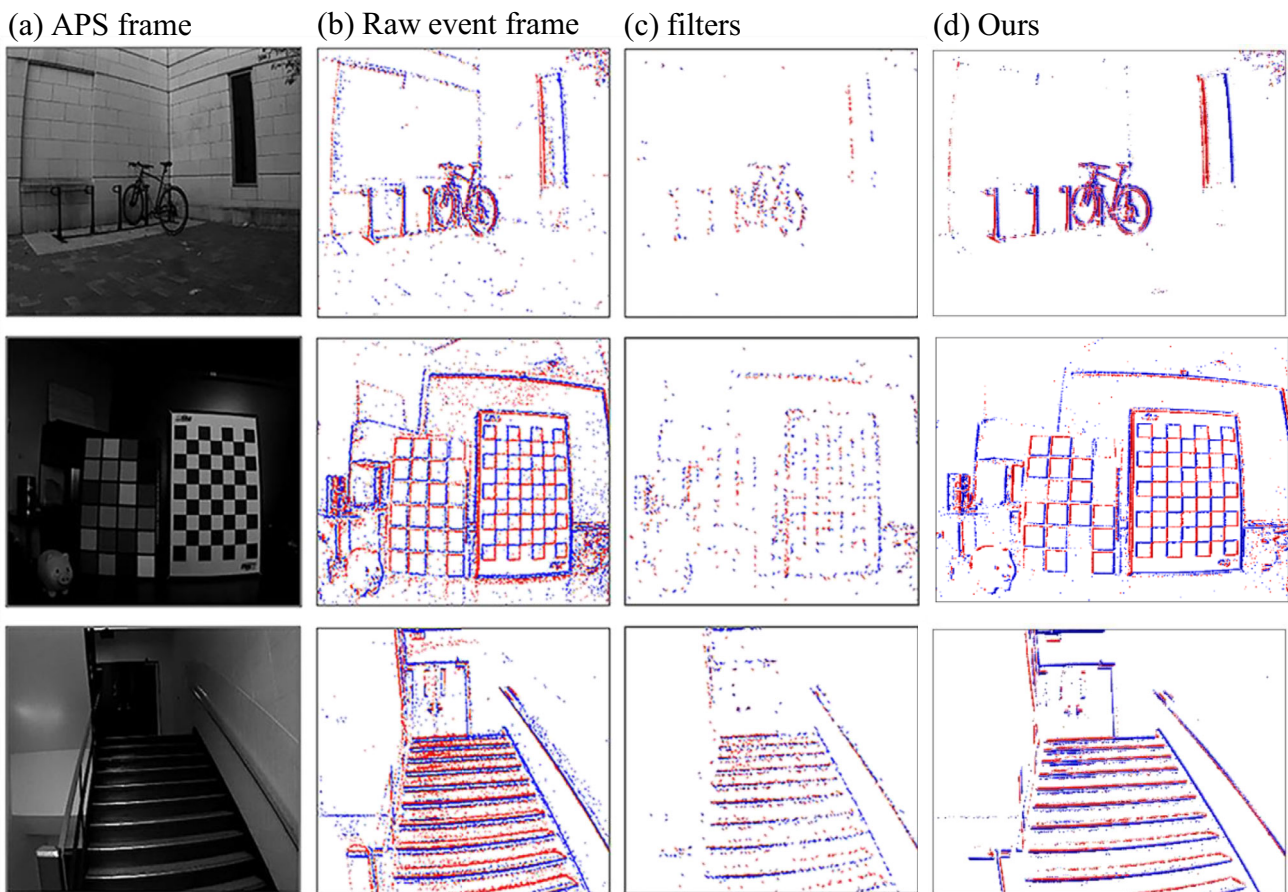


Fig. 9 **a** APS frame. **b** Raw event frame **c** denoising results based on filters. **d** Denoising results of the GMCM algorithm

4.6 Computational time

Benchmarking of the runtime for traditional denoising algorithms is summarized in Table 3. Experimental outcomes illustrate that, in contrast to the enhanced CM algorithm without Gaussian denoising preprocessing (denoted as MCM), our GMCM algorithm showcases a markedly improved runtime. This suggests that Gaussian denoising preprocessing can bolster computational efficiency, curbing unnecessary computational drain. Compared with Liu et al.[31], GMCM algorithm outperforms in terms of runtime, realizing a boost in computational efficiency of 166%.

5 Conclusion

In conclusion, a two-step GMCM algorithm for event denoising is proposed, encompassing Gaussian denoising preprocessing and motion denoising algorithm. The experimental

results show that the GMCM algorithm not only effectively distinguishes the Motion Stream and Noise Stream but also curtails unnecessary computational resource expenditure. Compared to the advanced spatiotemporal filter algorithm, the GMCM algorithm has achieved an upgraded computational efficiency of 166% in run time. Moreover, the GMCM algorithm achieves excellent denoising performances on DVSNOISE20 dataset and obtains the highest SNR score on DVSCLEAN dataset. In addition, low sensitivity to the number of events and high robustness to noise exhibited by the GMCM algorithm are other merits. The proposed two-step GMCM algorithm framework provides a new way to implement event denoising.

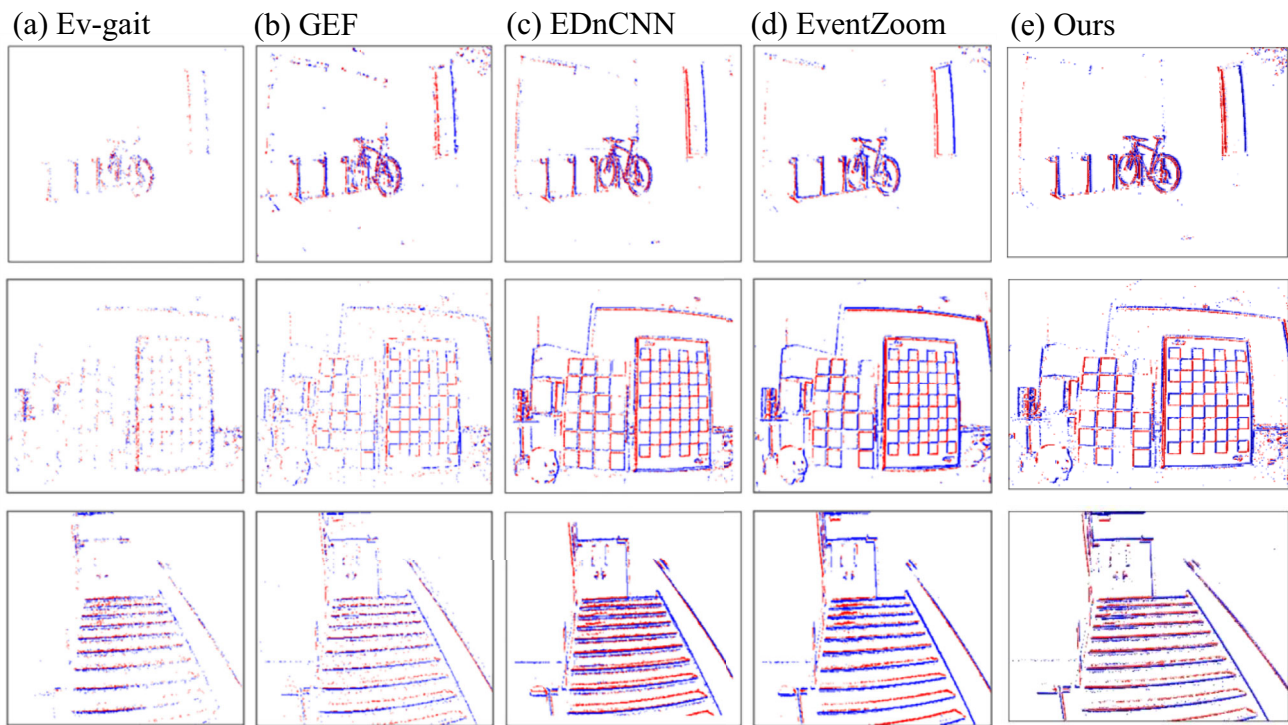


Fig. 10 a–d Denoising algorithm based on neural network. (e) Denoising results of the GMCM algorithm

Table 3 Comparison of average denoising runtimes for DVSNOISE20, DVSCLEAN, and our proprietary experimental dataset. (Units: seconds)

	CM	MCM	GMCM	Liu et al
DVSNOISE20	467.94	401.65	325.07	458.80
Our data	214.20	360.46	188.35	428.06
DVSCLEAN	9.98	6.47	6.47	413.73
Average	230.71	256.19	173.30	433.53

Bolding implies optimal runtime on datasets and average

Acknowledgements This work was supported by the Guangdong Province Key R&D projects under Grant 2019B010154002 and National Natural Science Foundation of China (11772359). And the authors would thank to Peiqi Duan for providing assistance and data.

Data availability The datasets used in this paper are public datasets.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Gallego, G., Delbruck, T., Orchard, G., Bartolozzi, C., Tabbara, B., Censi, A., et al.: Event-based vision: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 154–180 (2020)
2. Li, J., Dong, S., Yu, Z., Tian, Y., Huang, T.: Event-based vision enhanced: a joint detection framework in autonomous driving. 2019 IEEE International Conference on Multimedia and Expo (ICME). pp. 1396–401 (2019).
3. Kamiński, K., Cohen, G., Delbruck, T., Żołnowski, M., Gędek, M.: Observational evaluation of event cameras performance in optical space surveillance. 1st NEO and Debris Detection Conference. pp. (2019).
4. Liu, S.-C., Rueckauer, B., Ceolini, E., Huber, A., Delbruck, T.: Event-driven sensing for efficient perception: vision and audition algorithms. *IEEE Signal Process. Mag.* **36**(6), 29–37 (2019)
5. Zhang, J., Zhao, K., Dong, B., Fu, Y., Wang, Y., Yang, X., et al.: Multi-domain collaborative feature representation for robust visual object tracking. *Vis. Comput.* **37**(9–11), 2671–2683 (2021)
6. Parihar, A.S., Varshney, D., Pandya, K., Aggarwal, A.: A comprehensive survey on video frame interpolation techniques. *Vis. Comput.* **38**(1), 295–319 (2021)
7. Guo, S., Wang, W., Wang, X., Xu, X.: Low-light image enhancement with joint illumination and noise data distribution transformation. *The Visual Computer*. pp. (2022).
8. Hu, Y., Liu, S.-C., Delbruck, T.: v2e: From video frames to realistic DVS Events. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 1312–21 (2021).

9. Koseoglu, B., Kaya, E., Balcisoy, S., Bozkaya, B.: ST sequence miner: visualization and mining of spatio-temporal event sequences. *Vis. Comput.* **36**(10–12), 2369–2381 (2020)
10. Liu, H.-C., Zhang, F.-L., Marshall, D., Shi, L., Hu, S.-M.: High-speed video generation with an event camera. *Vis. Comput.* **33**(6–8), 749–759 (2017)
11. Lichtsteiner, P., Posch, C., Delbruck, T.: A 128x128 120 dB 15 μ s latency asynchronous temporal contrast vision sensor. *IEEE J. Solid-State Circuits* **43**(2), 566–576 (2008)
12. Guo, S., Delbruck, T.: Low cost and latency event camera background activity denoising. *IEEE Trans Pattern Anal Mach Intell.* PP. pp. (2022).
13. Baldwin, R.W., Almatrafi, M., Asari, V., Hirakawa, K.: Event probability mask (EPM) and event denoising convolutional neural network (EDnCNN) for neuromorphic cameras. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1698–707 (2020).
14. Fang, H., Wu, J., Li, L., Hou, J., Dong, W., Shi, G.: AEDNet: Asynchronous event denoising with spatial-temporal correlation among irregular data. *Proceedings of the 30th ACM International Conference on Multimedia.* pp. 1427–35 (2022).
15. Guo, S., Kang, Z., Wang, L., Li, S., Xu, W.: HashHeat: An O(C) complexity hashing-based filter for dynamic vision sensor. 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC). pp. 452–7 (2020).
16. Czech, D., Orchard, G.: Evaluating noise filtering for event-based asynchronous change detection image sensors. 2016 6th IEEE International Conference on Biomedical Robotics and Biomechanics (BioRob). pp. 19–24 (2016).
17. Feng, Y., Lv, H., Liu, H., Zhang, Y., Xiao, Y., Han, C.: Event density based denoising method for dynamic vision sensor. *Appl. Sci.* **10**(6), 2024 (2020)
18. Duan, P., Wang, Z.W., Zhou, X., Ma, Y., Shi, B.: EventZoom: learning to denoise and super resolve neuromorphic events. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12819–28 (2020).
19. Khodamoradi, A., Kastner, R.: O(N)-space spatiotemporal filter for reducing noise in neuromorphic vision sensors. *IEEE Trans. Emerg. Topics Comput.* pp. 15–23 (2018).
20. Wang, Z.W., Duan, P., Cossairt, O., Katsaggelos, A., Huang, T., Shi, B.: Joint filtering of intensity images and neuromorphic events for high-resolution noise-robust imaging. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1606–16 (2020).
21. Suh, Y., Choi, S., Ito, M., Kim, J., Lee, Y., Seo, J., et al.: A 1280 \times 960 dynamic vision sensor with a 4.95- μ m pixel pitch and motion artifact minimization. 2020 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 1–5 (2020).
22. Wu, J., Ma, C., Li, L., Dong, W., Shi, G.: Probabilistic undirected graph based denoising method for dynamic vision sensor. *IEEE Trans. Multimed.* **23**, 1148–1159 (2021)
23. Xu, N., Zhao, J., Ren, Y., Wang, L.: A noise filter for dynamic vision sensor based on spatiotemporal correlation and hot pixel detection. *Proceedings of 2021 International Conference on Autonomous Unmanned Systems (ICAUS 2021).* Lecture Notes in Electrical Engineering. pp. 792–9 (2022).
24. Yang, J., Ma, M., Zhang, J., Wang, C.: Noise removal using an adaptive Euler's elastica-based model. *The Visual Computer.* pp. (2022).
25. Cheng, W., Luo, H., Yang, W., Yu, L., Chen, S., Li, W.: DET: A high-resolution DVS dataset for lane extraction. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 1666–75 (2019).
26. Xie, X., Du, J., Shi, G., Hu, H., Li, W.: An improved approach for visualizing dynamic vision sensor and its video denoising. *Proceedings of the International Conference on Video and Image Processing.* pp. 176–80 (2017).
27. Baldwin, R.W., Almatrafi, M., Kaufman, J.R., Asari, V., Hirakawa, K.: Inceptive event time-surfaces for object classification using neuromorphic cameras. *image analysis and recognition. Lecture Notes in Computer Science.* pp. 395–403 (2019).
28. Ieng, S.-H., Posch, C., Benosman, R.: Asynchronous neuromorphic event-driven image filtering. *Proc. IEEE* **102**(10), 1485–1499 (2014)
29. Guo, S., Kang, Z., Wang, L., Zhang, L., Chen, X., Li, S., et al.: HashHeat: a hashing-based spatiotemporal filter for dynamic vision sensor. *Integration.* **81**, 99–107 (2021)
30. Wu, J., Ma, C., Yu, X., Shi, G.: Denoising of event-based sensors with spatial-temporal correlaTION. *ICASSP* **2020**, 4437–4441 (2020)
31. Liu, H., Brandli, C., Li, C., Liu, S.-C., Delbruck, T.: Design of a spatiotemporal correlation filter for event-based sensors. 2015 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 722–5 (2015).
32. Wang, Y., Du, B., Shen, Y., Wu, K., Zhao, G., Sun, J., et al.: EV-Gait: event-based robust gait recognition using dynamic vision sensors. pp. 6351–60 (2019).
33. Duan, P., Wang, Z.W., Shi, B., Cossairt, O., Huang, T., Katsaggelos, A.K.: Guided event filtering: synergy between intensity images and neuromorphic events for high performance imaging. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(11), 8261–8275 (2022)
34. Gallego, G., Rebecq, H., Scaramuzza, D.: A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3867–76 (2018).
35. Stoffregen, T., Kleeman, L.: Event cameras, contrast maximization and reward functions: an analysis. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12292–300 (2019).
36. Xiang, X., Zhu, L., Li, J., Tian, Y., Huang, T.: Temporal up-sampling for asynchronous events. 2022 IEEE International Conference on Multimedia and Expo (ICME). pp. 01–6 (2022).
37. Xu, J., Jiang, M., Yu, L., Yang, W., Wang, W.: Robust motion compensation for event cameras with smooth constraint. *IEEE Trans. Comput. Imag.* **6**, 604–614 (2020)
38. Gallego, G., Gehrig, M., Scaramuzza, D.: Focus is all you need: loss functions for event-based vision. *CVPR.* pp. 12280–12289 (2019).
39. Rebecq, H., Gehrig, D., Scaramuzza, D.: ESIM: an open event camera simulator. In: Aude B, Anca D, Jan P, Jun M, editors. *Proceedings of The 2nd Conference on Robot Learning. Proceedings of Machine Learning Research: PMLR.* pp. 969–82 (2018).
40. Huang, T., Zheng, Y., Yu, Z., Chen, R., Li, Y., Xiong, R., et al.: 1000 \times Faster camera and machine vision with ordinary devices. *Engineering.* pp. (2022).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



Wanmin Lin is currently a graduate student at the School of Electronics and Information Engineering, Sun Yat-sen University, China, under the supervision of Prof. Lilin Liu. She received her B.S. degree at the Sun Yat-sen University in June 2021. Her research interests include computer vision, data processing of dynamic vision sensors (event camera), deep learning.



Chen Xu received his Ph.D. degree in Machine Electronics Engineering from South China University of Technology, Guangzhou, China. Now he is working at Guangzhou Huangpu Pazhou Laboratory, China. His current research interests include artificial intelligence, intelligent sensing, image processing, machine vision, 3D reconstruction, and deep learning.



Yuhui Li received his Master's degree at South China Normal University in Guangzhou, China. He is currently working toward the Ph.D. degree in the State Key Lab of Optoelectronic Materials and Technology, School of Electronics and Information Technology, Sun Yat-Sen University, Guangzhou, China. His research interests include event camera, 3D reconstruction, image processing, deep learning.



Lilin Liu received her Ph.D. degree from Department of Mechanical Engineering at Hong Kong University of Science & Technology. She obtained her Master and Bachelor degrees from Beihang University. She is an full professor at School of Electronics and Information Technology, Sun Yat-Sen University. Her research interests include 3D information acquisition, imaging and display, machine vision, and optical communication.