Asynchronous Spatio-Temporal Spike Metric for Event Cameras

Jianing Li, Yihua Fu, Siwei Dong, Zhaofei Yu, Tiejun Huang, *Senior Member, IEEE,* and Yonghong Tian, *Senior Member, IEEE*

***Abstract*—Event cameras as bio-inspired vision sensors, have shown great advantages in high dynamic range and high temporal resolution in vision tasks. Asynchronous spikes from event cameras can be depicted using the marked spatio-temporal point processes (MSTPPs). However, how to measure the distance between asynchronous spikes in the MSTPPs still remains an open issue. To address this problem, we propose a general asynchronous spatio-temporal spike metric considering both spatio-temporal structural properties and polarity attribute for event cameras. Technically, the conditional probability density function is firstly introduced to describe the spatio-temporal distribution and polarity prior in the MSTPPs. Besides, a spatio-temporal Gaussian kernel is defined to capture the spatio-temporal structure, which transforms discrete spikes into the continuous function in a reproducing kernel Hilbert space (RKHS). Finally, the distance between asynchronous spikes can be quantified by the inner product in the RKHS. The experimental results demonstrate that the proposed approach outperforms the state-of-the-art methods and achieves significant improvement in computational efficiency. Especially, it is able to better depict the changes involving spatio-temporal structural properties and polarity attribute.**

***Index Terms*—Spatio-temporal point processes, spike metric, event cameras, kernel learning, neuromorphic engineering.**

I. Introduction

EVENT cameras, namely neuromorphic cameras, such as the dynamic vision sensor (DVS) [1]–[3], are bio-inspired vision sensors that, in contrast to frame-based cameras, work in a completely different way: pixels independently respond to intensity changes with a stream of asynchronous ***spikes***, instead of providing structured frames at a fixed rate. Indeed, event cameras are gaining more and more attentions in computer vision [4]–[15] owning to the advantages over conventional cameras: high temporal resolution and low latency (both in order of microseconds), high dynamic range (HDR), low power, and little redundancy.

Generally, the address-event representation (AER) [16] protocol for event cameras is utilized to output the spikes, and a spike can be described by a tuple $\langle x, y, t, p\rangle$ which consists of four essential elements: coordinates $\langle x, y\rangle$, firing time t, and polarity p respectively. As a consequence, a spike stream, namely spatio-temporal events with labeled polarity, is a set of discrete and sparse points, and it can be depicted using the marked spatio-temporal point processes (MSTPPs) [17], [18]. In fact, how to measure the distance between asynchronous spikes in the MSTPPs, as shown in Fig. 1, is one of the key challenges of event-based signal processing in many practical applications. Specifically, it refers to neuromorphic engineering (e.g., retinal prosthesis measurement [19], [20] and multi-neuron synchrony [21], [22]). Then, it is commonly related to spike coding involving motion estimation [23] and distortion measurement [24]. What's more, it is also important for machine learning towards event-based vision [20], [25]–[28] and robust learning with point process networks [29]–[32], in particular for supervised learning algorithms that deal with asynchronous spatio-temporal spikes, because of the shortcomings of loss functions to measure distance [20], [33] and train event-based models [27], [28], [34].

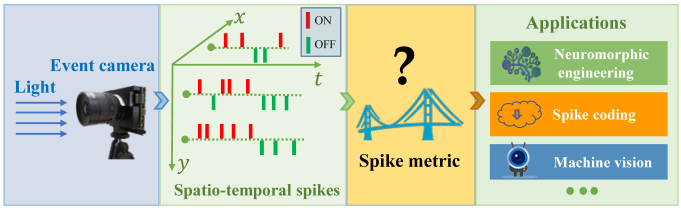


Fig. 1: The perspectives and challenges of asynchronous spatio-temporal ***spike metric*** for event cameras. How to measure the distance between asynchronous spikes in the marked spatio-temporal point processes (MSTPPs) still remains an open issue, and it is one of the key challenges of event-based signal processing in many practical applications involving neuromorphic engineering [19]–[22], spike coding [23], [24] and machine vision [20], [25]–[28], [33].

Since the spatio-temporal spike space is devoid of an algebra [17], [18], it imposes many challenges to event-based signal processing approaches, and existing image or video quality assessment techniques cannot be directly applied to this novel data. As a result, transformation strategies [18] have been made by mapping a set of spikes into image-like 2D representations prior to processing. Examples are the integration of spikes on the image planes [10], [33], [35] as well as time surfaces [36], [37], but those representations fail to completely depict raw spatio-temporal structure for asynchronous spikes. In addition, some works have been

This work is partially supported by grants from the National Natural Science Foundation of China under contract No. 61825101, No. 61425025 and No. 61806011.
Corresponding author: Yonghong Tian (e-mail: yhtian@pku.edu.cn).
J.Li, S.Dong and Z.Yu are with the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China.
Y. Fu is with the School of Electronic and Computer Engineering, Peking University, Shenzhen 518055, China.
T.Huang and Y.Tian are with the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China, and also with the Peng Cheng Laboratory, Shenzhen 518055, China.

done in order to quantify the similarity between two spike trains [17], [38]–[40], which treat single-neuron spike train as the operational unit using several positive-define kernels in Hilbert space. Recently, these have been further extended to multi-neuron [22], [23], [41]–[43]. However, for the most part, such attempts still remain a mere curiosity to explore the sum kernel for computational neuroscience, which is the unweighted sum of temporal kernels over single-neuron spike trains. As a matter of fact, for asynchronous spikes from event cameras which typically contain structural properties [13], [44], [45], this is the prime importance to characterize spatio-temporal distribution for spike metrics in the MSTPPs. Besides, the polarity $p\in\{1,-1\}$ is one significant attribute from the recordings of event cameras that represents the illumination change using ON or OFF spikes [11], [46]. In other words, spike metrics need to incorporate spatio-temporal structure and polarity attribute existing in the MSTPPs so as to better measure the distance between asynchronous spikes.

Toward this end, this paper proposes a general asynchronous spatio-temporal spike metric (ASTSM)[†] considering both spatio-temporal structural properties and polarity attribute for event cameras. Technologically, we first put forward the conditional probability density function to describe the spatio-temporal distribution and polarity prior in the MSTPPs. Then, a spatio-temporal Gaussian (i.e., 3D Gaussian) kernel is introduced to depict spatio-temporal structure, and it transforms discrete spikes into the conditional intensity function in a reproducing kernel Hilbert space (RKHS). Finally, the distance between asynchronous spikes can be measured using the inner product in the RKHS.

In summary, the main contributions of this paper are as follows:

- We propose an asynchronous spatio-temporal spike metric taking into account spatio-temporal structural properties and polarity attribute for real data from event cameras. Additionally, we provide the results and explanations on why our approach performs better than the state-of-the-art methods.
- We establish the conditional probability density function depicting polarity priors in the MSTPPs. In particular, a 3D kernel function, involving spatio-temporal structure, is introduced to map discrete spikes into the conditional intensity function in the RKHS, and we further build an optimization kernel parameters model to increase the flexibility and purse the better performance for a known statistical distribution.
- We provide a spike metric dataset containing simulating data and an augmentation of existing datasets with various distortion operation. We believe this dataset opens up an opportunity for the research of this challenging problem.

To the best of our knowledge, this is the first work to explore such a method for asynchronous spatio-temporal spike metric for event cameras. We believe that this methodology has the potential to enlarge the footprint of the point process applied to event-based signal processing and neuromorphic vision.

[†]The project's code and dataset are available on the following page: https://github.com/jianing-li/asynchronous-spatio-temporal-spike-metric

The rest of this paper is organized as follows. Spike metrics are reviewed in Section II. Section III describes the spike firing mechanism and then defines the MSTPPs for asynchronous spikes. Section IV presents the approach considering both spatio-temporal structure and polarity attribute. Finally, the experimental results and some discussions are reported in Section V, while some conclusions are drawn in Section VI.

II. RELATED WORK

In general, spike metrics aim at quantifying the difference between event-based streams [17], [22]. The early attempts, utilizing rate-based coding strategies [47], [48], have been demonstrated that the average spike count can primitively represent the discrimination between neuron responses in neuroscience. Moreover, transformation strategies have been made by integrating a set of spikes into the image planes [10], [33] as well as time surfaces [36], [37]. Obviously, those rate-based hypotheses fail to make the best of spike timings for neural processing. On the contrary, point process theory for analyzing spatio-temporal data develops primarily in statistics [18], and currently this theory has been the most widely used method to measure the distance between two or more spike trains [17], [22], [38]–[43] in computational neuroscience as well as in all other areas of neuromorphic engineering [23], [24], [49]. According to the literature, these approaches can be broadly classified as single-neuron spike train metrics and multi-neuron spike trains metrics as follows.

A. Single-Neuron Spike Train Metrics

In order to measure the distances between two spike trains, some single-neuron spike train metrics [17], [38]–[40] have been mainly focused on kernel methods. Rossum et al. [38] adopt the embedding approach where two spike trains are convolved with a Gaussian kernel function, then the distance is computed as the integral of the difference between two resulting functions. Paiva et al. [39] present a general framework in the RKHS to mathematically manipulate spike trains, the main idea of which is the definition of the inner product to allow event-based signal processing from basic principles. The model in [17], soon followed by the other work [40], incorporates simple mathematical analogies and attempts the positive definite function to quantify spike trains. Nevertheless, those metrics, without exploiting the spatio-temporal structural information in multi-neuron spike trains, are most commonly utilized in the single-neuron spike train.

B. Multi-Neuron Spike Trains Metrics

On these bases, those approaches have been further developed to multi-neuron spike trains metrics [22], [23], [41], [42], namely spatio-temporal spike metrics. The work by Houghton et al. [41] explores an extension of the van Rossum metric [38] to multi-neuron measurement, where basis vectors are used to interpolate between two views in a geometric vector space. Brockmeier et al. [42] exhibit the problem of optimizing multi-neuron spike trains metric to decode the real recorded neural data. More recently, Tezuka [22] utilizes

the R-convolution kernel to measure the distance for neural decoding. What's more, Dong et al. [23] model one pixel as a single-neuron for event camera and then implements the unweighted sum of single-neuron spike train temporal kernels for motion estimation in spike compression. Unfortunately, a review of the existing methods [22], [23], [41], [42] on multi-neuron can be found that such attempts make no use of spatio-temporal structure in the MSTPPs, especially polarity prior for event cameras. Therefore, this paper proposes an asynchronous spatio-temporal spike metric for event cameras considering spatio-temporal structure and polarity attribute in the MSTPPs.

III. Theoretical Foundation

In this section, we will first present the spike firing mechanism and then define the MSTPPs for the output from event cameras. In addition, we further summarize some shortages for the state-of-the-art methods by analyzing spatio-temporal structure and polarity attribute.

A. Marked Spatio-Temporal Spikes

Event cameras [1], [3], [50], in contrast to conventional frame-based cameras, have independent pixels that respond to the changes in the illuminance $L(\boldsymbol{u}, t)$. Specifically, a spike $e_n = \langle x_n, y_n, t_n, p_n \rangle$ is fired from a pixel $\boldsymbol{u}_n = [x_n, y_n]$ at the time t_n when the intensity change reaches a firing threshold C_{th}, and it is defined as:

$$\Delta lnL \doteq lnL(\boldsymbol{u}_n, t_n) - lnL(\boldsymbol{u}_n, t_n - \Delta t_n) = p_n C_{th} \tag{1}$$

where Δt_n is the time since the last spike at the same pixel, and the polarity $p_n \in \{1, -1\}$ represents ON or OFF spikes respectively.

The spike train $T = \{t_n \in \Gamma : n = 1, ..., N\}$, as shown in Fig. 2(a), is a sequence of ordered spike firing timestamps for each pixel in event cameras, which can be mathematically presented as:

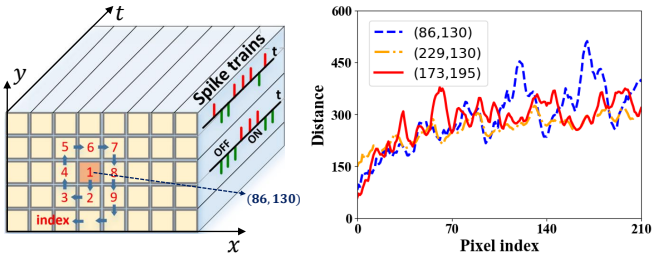
$$T(t) = \sum_{n=1}^{N} p_n \delta(t - t_n) \tag{2}$$

where N is the number of spikes in single pixel during the time interval Γ, and $\delta(\cdot)$ refers to the Dirac delta function, with $\delta(t) = 0, \forall x \neq 0$, and $\int \delta(t)\, dt = 1$.

Similarly, the pixels generate asynchronous spike stream $S = \{x_n, y_n, t_n \in \Gamma_s : n = 1, ..., N\}$ in the spatio-temporal interval Γ_s. S can be divided into spike cuboids [51], [52] $s \in S$, and it can be described as:

$$S(x, y, t) = \sum_{n=1}^{N} p_n \delta(x - x_n, y - y_n, t - t_n) \tag{3}$$

To give a measure-theoretical foundation to asynchronous spikes, some works have imposed a certain structure on set S. Based on these results [17], [18], [53], we assume that S is equipped with a metric d such that (S, d) is complete and separable. For instance, S could be a compact subset of $\mathbb{R}^2$ equipped with euclidean distance. In such a case, S is said to be a locally finite configuration, so the family of which can be denoted by N^{lf}. Formally, the MSTPPs can be defined as follows [18]:



(a) Spike trains (b) The distance [17]

Fig. 2: Measuring spatial correlation between spike trains. (a) We take one pixel as ordinate origin recorded spike train, which is a sequence of ordered spike firing timestamps for the current pixel. The pixel index is a spiral-like ordering around the current pixel. (b) The distance [17] between spike trains for three representative pixels shows that neighboring pixels are more relevant and asynchronous spikes exist spatial structure from event cameras.

Definition 1. (**Marked spatio-temporal point processes**). *Let (S, d) be a complete and separable metric space. The $S(x, y, t) = \sum_{n=1}^{N} p_n \delta(x - x_n, y - y_n, t - t_n)$, $0 < N < \infty$, is a measurable mapping from some probability space $[\Omega, \mathcal{F}, \mathbb{P}]$ into the measurable space N^{lf}. If $N < \infty$ almost surely (a.s.) then S is called a finite **MSTPPs**.*

The probability space $[\Omega, \mathcal{F}, \mathbb{P}]$ is a mathematical model for random experiments, where the sample space Ω is the set of all possible outcomes. $\mathcal{F}$ is the σ-algebra of subsets of the sample space, and $\mathbb{P} : \mathcal{F} \mapsto [0, 1]$ is probability measure.

B. Problem Statement

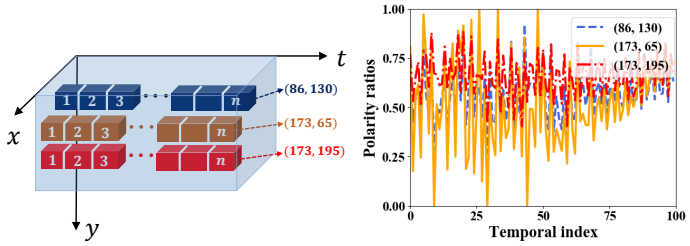
In practice, the core challenge is to quantify asynchronous spikes in the lack of stand algebraic operations such as linear projection and linear combination. Kernel methods [17], [23], [39], which have shined new light into this key problem by providing a general framework for measuring spike trains, can extend linear to non-linear modeling in input space, and especially map abstract object to Hilbert space. In this work, we adopt the kernel method to measure the distance between asynchronous spikes in the MSTPPs defined as follows:

Definition 2. (**Spatio-temporal spike distance**). *Let s_i and s_j be two spike cuboids in the spatio-temporal Γ_s respectively, and the inner product is introduced to measure the distance between asynchronous spikes in a Hilbert space by:*

$$\|s_i - s_j\| \triangleq \sqrt{\kappa(s_i, s_i) + \kappa(s_j, s_j) - 2\kappa(s_i, s_j)} \tag{4}$$

where $\kappa(s_i, s_j)$ is the inner product of two streams s_i and s_j.

To design an effective asynchronous spatio-temporal spike metric, we first conduct two experiments involving measuring the spatial correlation between spike trains and analyzing the statistical polarity attribute. The former depicted in Fig. 2,



(a) Spike cuboids (b) Polarity ratios

Fig. 3: Analyzing the statistical polarity attribute. (a) We regard the spike cuboid as computing element in temporal. (b) The ON polarity ratios, calculating in temporal index, are randomly changing, which illustrate that we can take polarity attribute as the prior probability distribution in spike metric.

takes one pixel as the ordinate origin from spike stream, and spike trains are made up of the surrounding pixels in Fig. 2(a). As shown in Fig. 2(b), the horizontal axis is the pixel index of spike train, and the vertical axis is the distance between two spike trains of ordinate origin and surrounding pixel index. We select three representative pixels as ordinate origins, respectively. The distances between spike trains are computed by using a single-neuron spike train metric [17], and which gain gradually with the increasing of pixel index. Note that spike trains from neighboring pixels are more relevant. In other words, spatial structural information exits in asynchronous spikes from event cameras. The latter one, regarding the spike cuboid as a computing element in Fig. 3(a), shows that the ON polarity ratios are randomly changing in Fig. 3(b). Hence, we can take polarity attribute as the prior probability distribution in spike metric. In fact, a review of existing approaches [22], [23], [41], [42] on multi-neuron spike trains metrics almost takes no use of spatio-temporal structure and polarity attribute.

In this respect, we propose the following *spatio-temporal structure* and *polarity attribute* to define our asynchronous spatio-temporal spike metric:

- ***spatio-temporal structure***: A 3D kernel function, involving spatio-temporal structure, is introduced to map discrete spikes into the conditional intensity function in the RKHS.
- ***polarity attribute***: By analyzing the spike firing mechanism, the conditional probability density function is established to depict polarity attribute in the MSTPPs.

Hereby, the potential of the above formulation will be highlighted in the following section by focusing on spatio-temporal structure and polarity attribute.

IV. OUR APPROACH

This section will give detailed descriptions of the proposed asynchronous spatio-temporal spike metric (ASTSM). Specifically, we first present the conditional intensity function (CIF) to depict the MSTPPs. Then, we introduce a 3D Gaussian kernel function to capture the spatio-temporal structure, and we further build an optimization model to learn kernel parameters. Finally, the distance between asynchronous spikes can be computed by the inner product in the RKHS. More details are described as follows.

A. Asynchronous Spatio-Temporal Spike Metric

In the MSTPPs, the CIF is extremely important factor contributing to describe spatio-temporal spikes [17], [18], which can depict the intensity in spiking history $H_t = \{e_n \in \Gamma_s | t_n < t\}$, and it is formulated as:

$$\lambda\left(x,y,t,p|H_t\right) = \frac{f\left(x,y,t,p|H_t\right)}{1-F\left(x,y,t|H_t\right)} \tag{5}$$

where $f\left(x,y,t,p|H_t\right)$ is the conditional probability density function in spiking history H_t, and $F\left(x,y,t|H_t\right)$ is the cumulative distribution function.

By applying the Bayesian theorem to Eq.(5), the CIF $\lambda\left(x,y,t,p|H_t\right)$ can be re-expressed as:

$$\begin{aligned}\lambda\left(x,y,t,p|H_t\right) &= \frac{f\left(x,y,t|H_t\right)}{1-F\left(x,y,t|H_t\right)} \cdot f\left(p|H_t,x,y,t\right) \\ &= \lambda\left(x,y,t|H_t\right) \cdot f\left(p|H_t,x,y,t\right)\end{aligned} \tag{6}$$

where $\sum\limits_{p\in\{1,-1\}} \iiint_{\Gamma_s} \lambda^2\left(x,y,t,p|H_t\right) dxdydt < \infty$, thus the intensity function $\lambda\left(x,y,t,p|H_t\right)$ is an element of $L_2\left(\Gamma_s\right)$ space.

In fact, we can choose a 3D smoothing function $h\left(x,y,t\right)$ to capture spatio-temporal structure, and use the 3D convolution to convert discrete spikes to continuous intensity function. It is similar to the CIF $\lambda\left(x,y,t,p|H_t\right)$ in history time H_t without considering the polarity, and it is computed as follows:

$$\begin{aligned}\lambda\left(x,y,t|H_t\right) &= s\left(x,y,t\right) * h\left(x,y,t\right) \\ &= \sum_{n=1}^{N} h\left(x-x_n, y-y_n, t-t_n\right)\end{aligned} \tag{7}$$

Then, $f\left(p|H_t,x,y,t\right)$ can be calculated based on the polarity probability distribution in history time H_t, and it is modeled as:

$$f\left(p|H_t,x,y,t\right) = \frac{\#\left\{e_n \in \Gamma_s | p_n = p, x_n < x, y_n < y, t_n < t\right\}}{\#\left\{e_n \in \Gamma_s\right\}} \tag{8}$$

where $\#\{\}$ represents the counting numbers in the spatio-temporal interval Γ_s.

Besides, for any two spike cuboids s_i and s_j, the inner product $\kappa\left(s_i,s_j\right)$ can be given by:

$$\begin{aligned}\kappa\left(s_i,s_j\right) &= \left\langle \lambda_{s_i}\left(x,y,t,p|H_t\right), \lambda_{s_j}\left(x,y,t,p|H_t\right)\right\rangle_{L_2(\Gamma_s)} \\ &= \sum_{p\in\{1,-1\}} \iiint_{\Gamma_s} \lambda_{s_i}\left(x,y,t,p|H_t\right) \lambda_{s_j}\left(x,y,t,p|H_t\right) dxdydt\end{aligned} \tag{9}$$

It follows from Eq.(6) and Eq.(7) that Eq.(9) can be rewritten as:

$$\begin{aligned}\kappa\left(s_i,s_j\right) = \sum_{p\in\{1,-1\}} \sum_{m=1}^{N_i} \sum_{n=1}^{N_j} \iiint_{\Gamma_s} & f_{s_i}\left(p|H_t,x,y,t\right) \\ & \cdot f_{s_j}\left(p|H_t,x,y,t\right) \cdot h\left(x-x_m^{(i)}, y-y_m^{(i)}, t-t_m^{(i)}\right) \\ & \cdot h\left(x-x_n^{(j)}, y-y_n^{(j)}, t-t_n^{(j)}\right) dxdydt\end{aligned} \tag{10}$$

where N_i, N_j are the numbers in spike cuboids s_i and s_j.

In order to facilitate the analysis and calculation, we further simplify that $f_{s_i}\left(p|H_t,x,y,t\right)$ and $f_{s_j}\left(p|H_t,x,y,t\right)$ can be

approximated to polarity statics $A(s_i, s_j)$ in the whole spike cuboid respectively, and it can be presented as:

$$A(s_i, s_j) \approx \sum_{p\in\{1,-1\}} f_{s_i}(p|H_t, x, y, t)\, f_{s_j}(p|H_t, x, y, t) \quad (11)$$

For two spikes $e_m^{(i)}$ and $e_n^{(j)}$ in spike cuboids s_i and s_j respectively, the inner product between two spikes can be given by:

$$\begin{aligned}\kappa\left(e_m^{(i)}, e_n^{(j)}\right) = \iiint_{\Gamma_s} h\left(x - x_m^{(i)}, y - y_m^{(i)}, t - t_m^{(i)}\right) \\ \cdot h\left(x - x_n^{(j)}, y - y_n^{(j)}, t - t_n^{(j)}\right) dxdydt\end{aligned} \quad (12)$$

For simplicity, a 3D Gaussian kernel is utilized as smoothing function in this study, and it can be defined as:

$$h(x, y, t) = \frac{e^{-\frac{x^2}{2\sigma_x^2}}}{l_x} \cdot \frac{e^{-\frac{y^2}{2\sigma_y^2}}}{l_y} \cdot \frac{e^{-\frac{t^2}{2\sigma_t^2}}}{l_t} \quad (13)$$

where σ_x, σ_y, and σ_t are the standard deviation parameters of the 3D Gaussian kernel, and $l_x = \sqrt{\sqrt{\pi}\sigma_x}$, $l_y = \sqrt{\sqrt{\pi}\sigma_y}$, and $l_t = \sqrt{\sqrt{\pi}\sigma_t}$.

From Eq.(12) and Eq.(13), so the $\kappa\left(e_m^{(i)}, e_n^{(j)}\right)$ can be re-expressed as:

$$\kappa\left(e_m^{(i)}, e_n^{(j)}\right) = e^{-\frac{\left(x_m^{(i)} - x_n^{(j)}\right)^2}{4\sigma_x^2} - \frac{\left(y_m^{(i)} - y_n^{(j)}\right)^2}{4\sigma_y^2} - \frac{\left(t_m^{(i)} - t_n^{(j)}\right)^2}{4\sigma_t^2}} \quad (14)$$

In addition, using Eq.(11) and Eq.(14), the inner product $\kappa(s_i, s_j)$ between two spike cuboids can be further rewritten as:

$$\begin{aligned}\kappa(s_i, s_j) &= A(s_i, s_j) \cdot \sum_{m=1}^{N_i}\sum_{n=1}^{N_j} \kappa\left(e_m^{(i)}, e_n^{(j)}\right) \\ &= \sum_{p\in\{1,-1\}} f_{s_i}(p|H_t, x, y, t)\, f_{s_j}(p|H_t, x, y, t) \\ &\cdot \sum_{m=1}^{N_i}\sum_{n=1}^{N_j} e^{-\frac{\left(x_m^{(i)} - x_n^{(j)}\right)^2}{4\sigma_x^2} - \frac{\left(y_m^{(i)} - y_n^{(j)}\right)^2}{4\sigma_y^2} - \frac{\left(t_m^{(i)} - t_n^{(j)}\right)^2}{4\sigma_t^2}}\end{aligned} \quad (15)$$

Note that the optimal kernel parameters $\boldsymbol{\theta} = \{\sigma_x, \sigma_y, \sigma_t\}$ can be learned by building the optimization problem, our ASTSM can become more flexible and pursue a better performance to measure the distance between asynchronous spikes.

B. Learning Kernel Parameters

Theoretically, the optimal kernel parameters $\boldsymbol{\theta}$ for 3D Gaussian kernel function can be learned by minimizing the fitting error [54]. In this paper, we further exploit the transformation strategy, which calculates the correlation coefficients between the distance d from spike metrics and performance score p_s (i.e. compression ratio [23], [24] or PSNR [55], [56]). Hence, the error function $\boldsymbol{J}(\boldsymbol{\theta})$ for given $\boldsymbol{\theta}$ can be written as:

$$\begin{aligned}&\boldsymbol{J}(\boldsymbol{\theta}) = \\ &\sum_{i\in R}\sum_{j\in D_i} \|d(S_i, S_j, \boldsymbol{\theta}) - f_b(p_s(S_i, S_j), b)\|^2 + \gamma\|b\|\end{aligned} \quad (16)$$

where d is the distance between two spike streams involving raw data S_i and distortion stream S_j, and the corresponding sets are R and D_i respectively. f_b is a polynomial function of degree b, which is possible to fit curve between the distance d and performance score p_s. Additionally, γ is a hyper-parameter that weights the relative contribution of the norm penalty term to avoid overfitting.

Then, we can solve the following minimization problem:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \boldsymbol{J}(\boldsymbol{\theta}) \quad (17)$$

In this case, the gradient with respect to the kernel parameters $\boldsymbol{\theta}$ is calculated, and we perform this update as follows:

$$\boldsymbol{\theta}^{(n+1)} = \boldsymbol{\theta}^{(n)} - \eta \cdot \nabla \frac{\partial \boldsymbol{J}}{\partial \boldsymbol{\theta}} \quad (18)$$

where the learning rate η is a hyper-parameter that controls how much we are adjusting $\boldsymbol{\theta}$ with respect to the loss gradient.

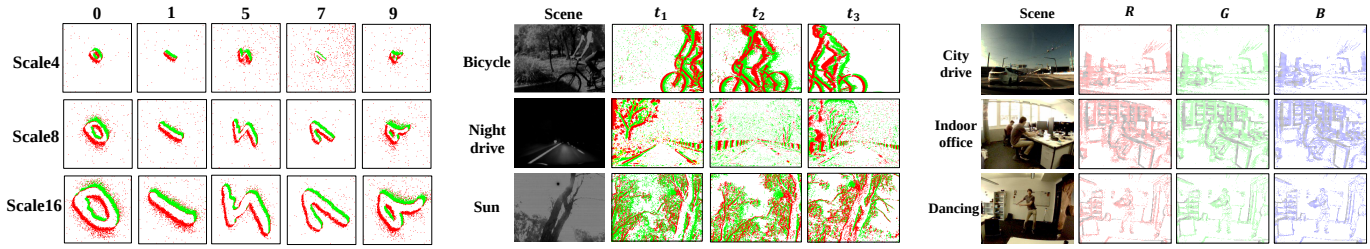
According to the gradient descent method, this updating equation can minimize the objective function $\boldsymbol{J}(\boldsymbol{\theta})$. It is noteworthy that the optimal kernel parameters depend on the statistical distribution from a dataset and specific computer vision tasks. In other words, the parameters $\boldsymbol{\theta}$ are closely linked to performance score p_s. The optimization process can be described as follows:

1) Set the learning rate η and the maximum iteration number N_o, and set ε to a very small positive number.
2) Initialize the kernel parameters $\boldsymbol{\theta^{(0)}}$ and set the iteration step $n = 0$.
3) Fit a polynomial function $f_b(p_s(S_i, S_j, \boldsymbol{\theta^{(0)}}), b)$ between the distance and performance score.
4) Update the kernel parameters $\boldsymbol{\theta^{(n)}}$ using gradient descent method in Eq.(18).
5) If $\boldsymbol{J}(\boldsymbol{\theta}) < \varepsilon$ or $n \geq N_o$, stop. Otherwise, set $n = n + 1$, go to step (4).

Finally, the distance $d(S_i, S_j, \boldsymbol{\theta}) = \|S_i - S_j\|$ between two spike steams S_i and S_j from event cameras can accumulate the distance between two spike cuboids s_i and s_j, which are regraded as basic computing units. After partitioned into multiple spike cuboids, the distance $\|s_i - s_j\|$ between spike cuboids s_i and s_j can be computed by Eq.(4) and Eq.(15), and more details are demonstrated in **Algorithm 1**.

Algorithm 1 Asynchronous spatio-temporal spike metric

Input: Two spike streams S_i and S_j
Output: The distance $\|S_i - S_j\|$ between two streams
Initialize : $\|S_i - S_j\| = 0$; spike cuboid parameters W, H and L; 3D Gaussian kernel parameters $\boldsymbol{\theta} = \{\sigma_x, \sigma_y, \sigma_t\}$
Learning 3D Gaussian kernel parameters : $\boldsymbol{\theta}^{(n+1)} \leftarrow \boldsymbol{\theta}^{(n)} + \eta \cdot \nabla \frac{\partial \boldsymbol{J}}{\partial \boldsymbol{\theta}}$ by Eq. (18)
Asynchronous streams S_i and S_j are divided into K spike cuboids s_i^k and s_j^k respectively
for $k = 1, ..., K$ **do**
  $f_{s_i}^k$, $f_{s_j}^k \leftarrow$ getPolarity(s_i^k, s_j^k) based on Eq.(8)
  $A(s_i^k, s_j^k) \leftarrow$ computePolarity($f_{s_i}^k$, $f_{s_j}^k$) by Eq.(11)
  $\kappa(s_i^k, s_j^k) \leftarrow$ getInnerProduct(s_i^k, s_j^k) using Eq.(15)
  $\left\|s_i^k - s_j^k\right\| \leftarrow$ getDistance(s_i^k, s_j^k) utilized Eq.(4)
  $\|S_i - S_j\| \leftarrow \|S_i - S_j\| + \left\|s_i^k - s_j^k\right\|$
end for
return $\|S_i - S_j\|$



(a) MNIST-DVS dataset [57] (b) DVS reconstruction dataset [55] (c) Color event camera dataset [58]

Fig. 4: For a better perspective, we directly map asynchronous spatio-temporal spikes into an image-like 2D representation in a time interval. (a) The MNIST-DVS [57] is captured by a DVS128 [1] for dynamic handwritten digits in the existing MNIST dataset. (b) The DVS reconstruction dataset [55] is recorded using a DAVIS240C [2] in natural scenes. (c) The color event camera dataset shows representative indoor and outdoor scenes based on a Color-DAVIS346 [3]. Notably, ON spikes are red, OFF spikes are green in (a) and (b). ON spikes are colored by the corresponding filter color, and OFF spikes are black in (c).

V. EXPERIMENTS AND DISCUSSIONS

This section will first introduce the detailed experimental settings. Then, we describe the implementation details of spike metrics and report the representative results. Moreover, two effective tests, involving polarity attribute and spatio-temporal structure, are conducted for performance evaluation. Finally, motivated by the previous work [53], we design a transformation strategy to quantify spike metrics.

A. Experimental Settings

To verify the effectiveness of our approach, we provide a spike metric dataset including the simulating dataset and real spatio-temporal spike streams, in which each spike stream has raw data and the corresponding distortions. Specifically, the simulating dataset depicts the moving target with an actual motion trajectory. Besides, the real dataset consists of three parts in Fig. 4, which are collected from the MNIST-DVS [57], the DVS reconstruction dataset [55], and the color event camera dataset (CED) [58], respectively. More precisely, the first part represents moving digits with three scales (i.e. scale4, scale8, and scale16) by a DVS128 [1]. The second part records natural scenes (i.e. bicycle, night drive, and sun) using a DAVIS240C [2], and the rest part captures a wide variety of indoor and outdoor scenes (i.e. city drive, indoor office, and dancing) based on a Color-DAVIS346 [3].

In order to conduct a comprehensive evaluation of the proposed asynchronous spatio-temporal spike metric (ASTSM), we compare our ASTSM and ASTSM$^+$ (i.e., ASTSM adopts the learning kernel parameters) with the state-of-the-art methods [17], [23] and two baselines, including:

1) KMST [17]: Multi-neuron spike trains metric that implements the unweighted sum of kernel method for each spike train (KMST).
2) KMST-P [23]: The approach that takes the interference polarity into measure spike train followed by the former work [17], which regards the interaction between ON and OFF polarities for event camera.
3) KMST$^+$: The KMST utilizes the proposed learning kernel parameters.
4) KMST-P$^+$: The KMST-P uses the proposed learning kernel parameters.

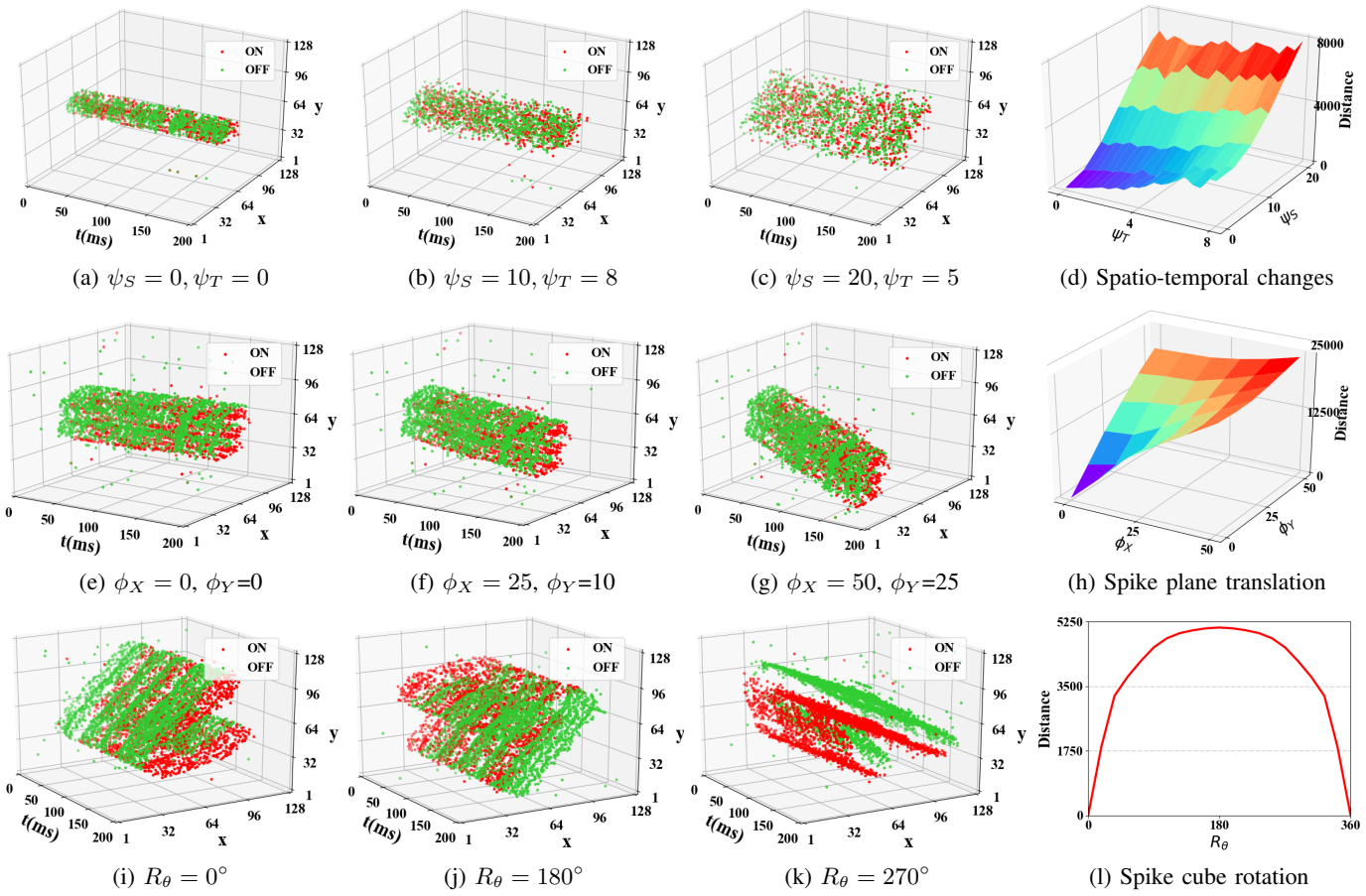
B. Validation Issues

The objective of the first experiment is to assess the validity of our approach, so that we adopt three representative distortion operations on raw spike stream with three scales (i.e. scale4, scale8, and scale16) from MNIST-DVS [57], which consist of randomly spatio-temporal coordinate changes, spike plane translation and spike cube rotation from raw data. Besides, the spike cuboid parameters are set as $W = 128$, $H = 128$, and $L = 1200\mu s$ in this experiment. Meanwhile, the parameters of the 3D Gaussian kernel are initialized as $\sigma_x = 5$, $\sigma_y = 5$, and $\sigma_t = 5000$. Some representative results and experimental analyses can be found as follows.

Validation on spatio-temporal coordinate changes. As illustrated in Fig. 5(a)-(c), we present raw data and two degraded spike streams, in which spatio-temporal coordinates for raw data are randomly changed. More precisely, ψ_S and ψ_T are the maximum changing values of x and y coordinates, respectively. We can see that two degraded spike streams become more sparse and scattering than raw data when $\psi_S = 10$ and $\psi_T = 8$ in Fig. 5(b), and $\psi_S = 20$ and $\psi_T = 5$ in Fig. 5(c). Much to our surprise, the distance between raw data and degraded spike streams is gradually growing with the increase of ψ_S and ψ_T in Fig. 5(d). This may be caused by the fact that our ASTSM can measure the distance between raw data and degraded spike streams via randomly spatio-temporal coordinate changes.

Validation on spike plane translation. Degraded spike streams are generated by translating spike plane with constant speed. In other words, we can implement the maximum spatio-temporal moving parameters ϕ_X and ϕ_Y within a constant temporal window. Note that the location of raw data has a liner movement in spatio-temporal domain when $\phi_X = 25$ and $\phi_Y = 20$ in Fig. 5(f), and $\phi_X = 50$ and $\phi_Y = 25$ in Fig. 5(j). The discrimination between raw data and degraded streams in Fig. 5(k), which is a smoothing 3D curve, indicates clearly that our ASTSM can also measure the distance under spike plane translation operation.

Validation on spike cube rotation. Distortion streams are obtained by rotating spike cube in x-y plane. More precisely, we first select a fixed point in x-y plane. Then, spike cube is rotated with different angles around the fixed point. Fig. 5(i)-



(a) $\psi_S = 0, \psi_T = 0$ (b) $\psi_S = 10, \psi_T = 8$ (c) $\psi_S = 20, \psi_T = 5$ (d) Spatio-temporal changes

(e) $\phi_X = 0$, ϕ_Y=0 (f) $\phi_X = 25$, ϕ_Y=10 (g) $\phi_X = 50$, ϕ_Y=25 (h) Spike plane translation

(i) $R_\theta = 0°$ (j) $R_\theta = 180°$ (k) $R_\theta = 270°$ (l) Spike cube rotation

Fig. 5: Three representative operations, involving randomly spatio-temporal coordinate changes in (a)-(c), spike plane translation in (e)-(g), and spike cube rotation from raw data in (i)-(k), are conducted to assess the validity of our ASTSM. The raw spike streams are three digits (i.e., "0", "3", and "7") of three scales (i.e. scale4, scale8, and scale16) within $200ms$ from MNIST-DVS [57], respectively. From (d), (h), and (l), we can see that our ASTSM can measure the distance between raw data and degraded spike streams.

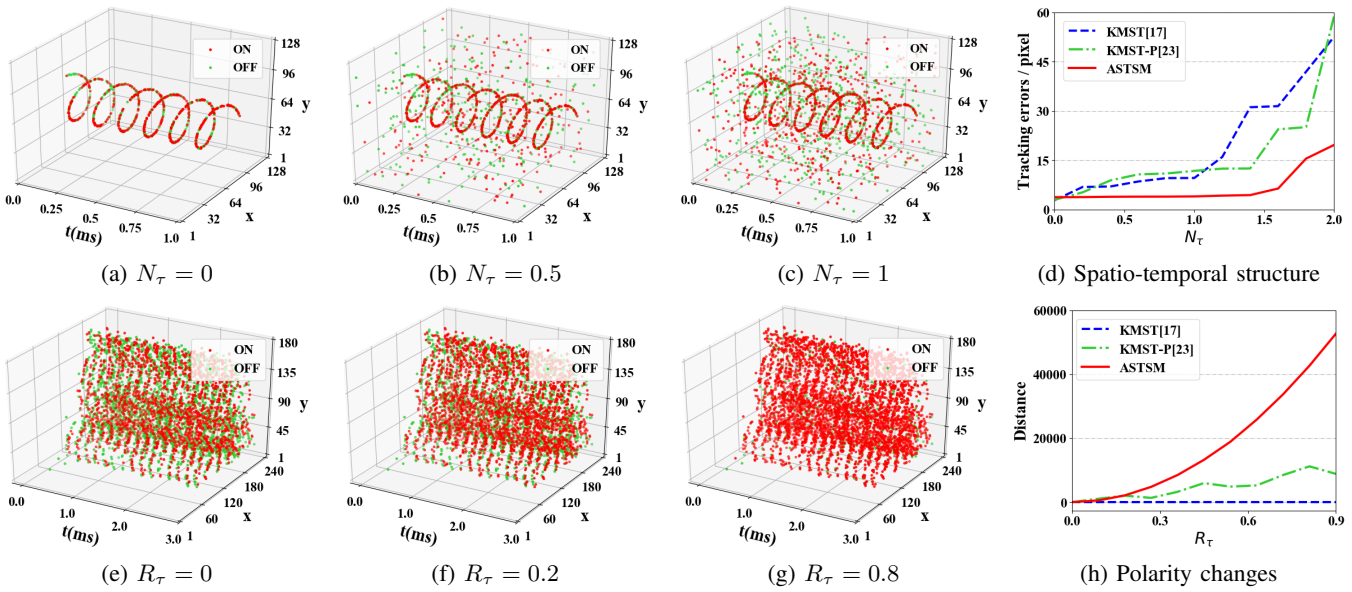
(l) report raw data and two degraded spike streams by rotating spike cube around the fixed point $(64, 64)$ in x-y plane. Degraded spike streams exist visual rotation in spatial domain when rotation angle $R_\theta = 180°$ in Fig. 5(j), and $R_\theta = 270°$ in Fig. 5(k). Obviously, Fig. 5(l) shows that the distance is symmetric curve along with R_θ, which is gradually increasing from the rotation angle $R_\theta = 0°$ to $R_\theta = 180°$ and then starts to decrease until $R_\theta = 360°$. Hence, our ASTSM can reflect this distortion operation involving spike cube rotation.

In summary, as shown in Fig. 5, we can see that the curves of our ASTSM are fairly consistent with three representative degraded operations. In other words, the proposed approach, considering both spatio-temporal structure and polarity attribute, can measure the discrimination between raw data and degraded spike streams in various degrees.

C. Effective Test

The second experiment aims to effectively evaluate the proposed spike metric from two perspectives: spatio-temporal structure and polarity attribute. To this end, two effectiveness tests are conducted on real asynchronous spikes and simulating data compared with the state-of-the-art methods [17], [23] as follows.

Evaluation on spatio-temporal structure. In this part, to understand how the changes of spatio-temporal structure influence spike metrics, we implement a motion estimation experiment, namely tracking a moving target in the spatio-temporal domain. Actually, tracking performances can reflect the accuracy of motion estimation by spike metrics for event camera. From a mathematical standpoint, we can consider two metric indexes including tracking error and robustness. We use the following simple rule: a cycling moving target is continuously tracked on various noise intensities N_τ that is more suitable in a practical setting. In such a case, Fig. 6(a)-(c) present moving target trajectories along with the increasing noise intensity N_τ. Besides, Fig. 6(d) describes three tracking error curves for spike metrics. Surprisingly, the red curve remains lower compared with other methods [17], [23] and is relatively stable by the end of N_τ near to 1.5, then it gradually starts to increase. Hereby, this indicates that our ASTSM, defining a 3D Gaussian kernel to capture spatio-temporal structure in the MSTPPs, can measure better the distance between spike streams than other methods [17], [23].



(a) $N_\tau = 0$ (b) $N_\tau = 0.5$ (c) $N_\tau = 1$ (d) Spatio-temporal structure

(e) $R_\tau = 0$ (f) $R_\tau = 0.2$ (g) $R_\tau = 0.8$ (h) Polarity changes

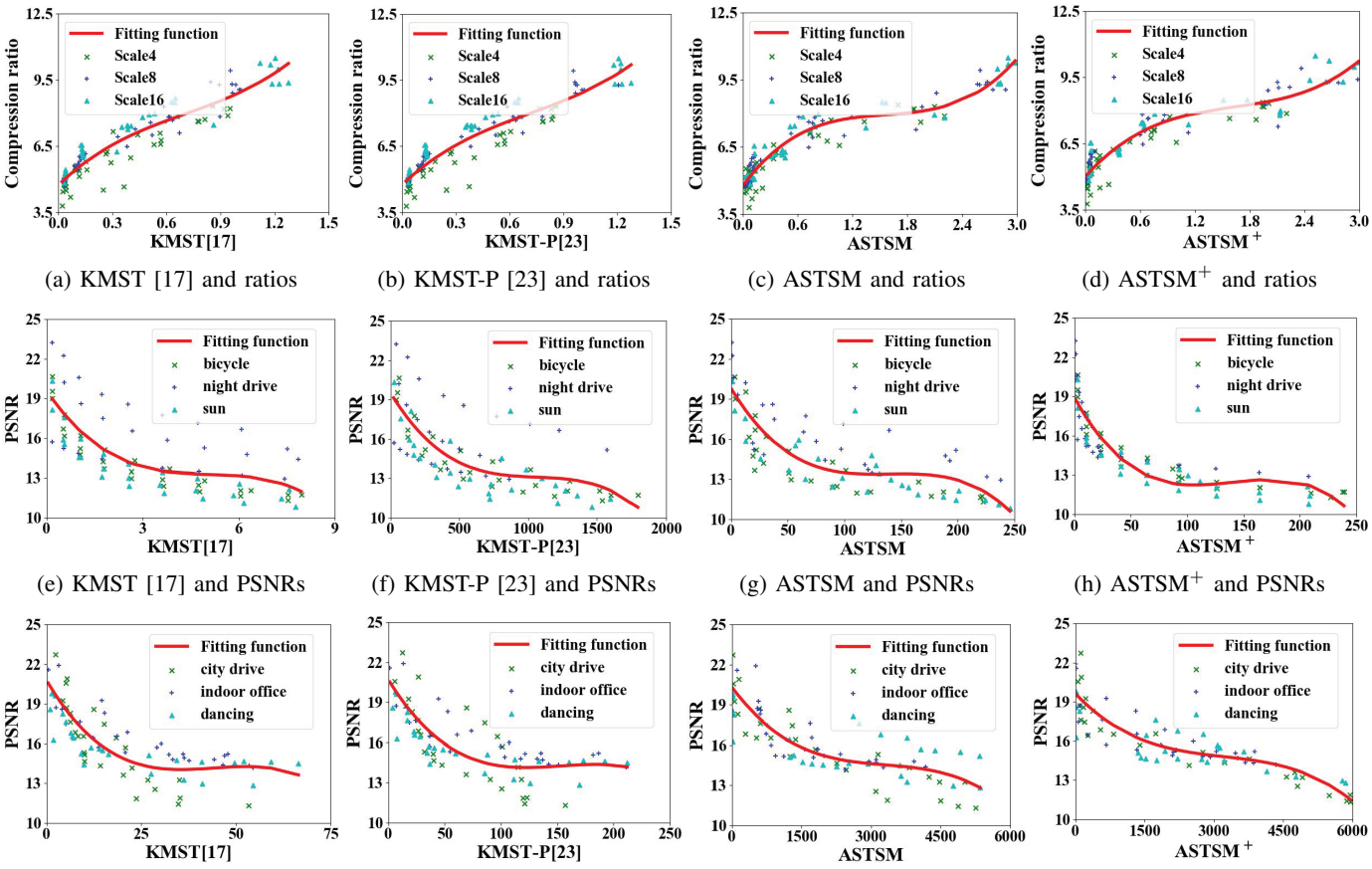
Fig. 6: Effective test from the perspectives of spatio-temporal structure and polarity attribute, respectively. (a)-(c) We illustrate raw data depicting cycling moving target from simulating dataset and two degraded spike streams by adding random noise with the intensity N_τ. (e)-(f) We show raw data and two degraded spike streams by changing the ratio R_τ of ON-OFF polarity, which record the bicycle scene using a DAVIS240C [2] with 5000 spikes from the DVS reconstruction dataset [55]. (d) and (h) indicate that our ASTSM outperforms the state-of-the-art methods [17], [23] involving the changes of spatio-temporal structure and polarity attribute.

Evaluation on polarity attribute. In this second part, we will explore the experiment to see why and how our ASTSM works in the changes of polarity attribute. Following the strategy, we remain on the coordinates and then change the ratio R_τ of ON-OFF polarity for spike stream. As we can see in Fig. 6(e)-(g), the greater the ratio R_τ is, the more ON polarity spikes exists in spatio-temporal domain, which takes in a greater variety of polarity attribute. Meanwhile, Fig. 6(h) plots three curves along with the increase of the ratio R_τ and reports the distances based on Our ASTSM and the state-of-the-art methods [17], [23]. Obviously, the blue dotted line indicates that the KMST [17], without taking into account polarity attribute, keeps unchanged and fails to depict the discrimination between raw data and degraded spike streams. In deed, the green dotted line illustrates that the KMST-C [23], considering the interference polarity into spike trains, can highlight the growing trend of distortion degrees with the increasing ratios R_τ. Nevertheless, this curve demonstrates the phenomenon of random fluctuation in spike metric. On the contrary, the proposed approach follows a smoothing curve in the red line, rather than the fluctuating curve. Notably, the proposed approach, introducing the conditional probability density function in the MSTPPs, can measure better than the state-of-the-art methods [17], [23] involving polarity changes.

D. Scalability Test

In this third experiment to provide the quantitative measurement on the performance of all spike metrics and further verify the generality of our method, we follow the evaluation procedures employed in the video quality test (e.g., PSNR [59] and SSIM [60]), where four evaluation metrics, consisting of Pearson linear correlation coefficient (PLCC), Spearman rank correlation coefficient (SRCC), Kendall's rank correlation coefficient (KRCC), and root mean-squared error (RMSE), are used to calculate the correlation coefficients between performance scores p_s (i.e. compression ratio [23], [24] and PSNR [55], [56]) and the distance d from all spike metrics. The larger PLCC, SRCC, and KRCC the better, whereas RMSE is the opposite. Due to asynchronous spatio-temporal spikes lack of the mean opinion score (MOS) by direct observation, which is obtained from subjective experiments. Motivated by the previous work [53], utilizing the recognition accuracy instead of the MOS to quantify the assessment for fingerprint images, we also adopt the same transformation strategy that two performance indexes are from compression ratio in lossy spike coding [24] and PSNR for real-time intensity reconstruction [55], [56] for event cameras. Additional details are provided in the supplementary material.

Besides, the spatial parameters of spike cuboid in our ASTSM are set as the length and the width of event camera array size, and the corresponding temporal parameter is set as $L = 1200\mu s$. We set the learning rate $\eta = 0.001$, the maximum number iteration number $N_o = 100$, and the initial 3D kernel parameters $\boldsymbol{\theta}^{(0)} = \{10, 10, 1000\}$ by the empirical values. According to the iteration using gradient descent in *Algorithm 1*, the optimization kernel parameters $\boldsymbol{\theta}^{(*)}$ for our ASTSM$^+$ in three collected datasets (i.e., the MNIST-DVS [57], the DVS reconstruction dataset [55], and the color event camera dataset (CED) [58]) are $\{8.96, 9.21, 983.21\}$, $\{9.21, 8.93, 932.39\}$, and $\{8.75, 9.35, 962.17\}$, respectively.



(a) KMST [17] and ratios (b) KMST-P [23] and ratios (c) ASTSM and ratios (d) ASTSM$^+$ and ratios

(e) KMST [17] and PSNRs (f) KMST-P [23] and PSNRs (g) ASTSM and PSNRs (h) ASTSM$^+$ and PSNRs

(i) KMST [17] and color PSNRs (j) KMST-P [23] in color PSNRs (k) ASTSM and color PSNRs (l) ASTSM$^+$ and color PSNRs

Fig. 7: Scalability test by fitting the curves between performance scores and the distance. Each sample point refers to one testing spike stream. (a)-(d) The curves show the trends of compression ratios and quantified distances from spike metrics, including KMST [17], KMST-P [23], our ASTSM, and ASTSM$^+$ (i.e., learning kernel parameters), in which loss spike coding [24] is conducted on the MNIST-DVS [57] with three scales (i.e. scale4, scale8, and scale16). (e)-(h) We present the correlation between the distance and PSNRs by reconstructing images on the DVS reconstruction dataset [55] (i.e. bicycle, night drive, and sun). (i)-(l) The results for color intensity reconstruction are reported on the color event camera dataset [58] (i.e. city drive, indoor office, and dancing) using events-to-video algorithm [56].

TABLE I
QUANTITATIVE EVALUATION ON SPIKE METRICS.

Methods	Learning	Compression [24]				DVS reconstruction [55]				DVS color reconstruction [56]			
		PLCC	SRCC	KRCC	RMSE	PLCC	SRCC	KRCC	RMSE	PLCC	SRCC	KRCC	RMSE
KMST [17]	No	0.9386	0.9334	0.7871	0.5545	0.7593	0.7584	0.5773	1.8714	0.8058	0.7942	0.5906	1.5704
KMST-P [23]	No	**0.9431**	0.9325	0.7852	0.5343	0.7610	0.7730	0.5969	1.8656	0.7926	0.8063	0.6215	1.6168
ASTSM	No	0.9425	**0.9369**	**0.7936**	**0.5336**	**0.8335**	**0.8038**	**0.6246**	**1.5884**	**0.8388**	**0.8173**	**0.6429**	**1.4175**
KMST$^+$	Yes	0.9423	0.9389	0.7965	0.5335	0.8697	0.8712	0.7147	1.3215	0.8685	0.8660	0.6936	1.3945
KMST-P$^+$	Yes	**0.9436**	0.9391	0.7974	0.5331	0.8779	0.8819	0.7263	1.3189	0.8631	0.8722	0.7141	1.3816
ASTSM$^+$	Yes	0.9428	**0.9412**	**0.8019**	**0.5329**	**0.9025**	**0.9230**	**0.7589**	**1.2385**	**0.8763**	**0.8868**	**0.7172**	**1.2545**

Quantitative evaluation on spike metrics. Specifically, we first conduct two types of degraded operations including lossy spike coding [24] on the MNIST-DVS [57] and adding random noise to the DVS reconstruction dataset [55] and the color event camera dataset [58]. Then, performance scores are computed by implementing lossy spike coding [24] and intensity reconstruction [55], [56] for spike streams, respectively. Finally, we further measure the distance between raw data and degraded spike streams using our ASTSM$^+$, in which we build an optimization model to learn kernel parameters and use the cubic polynomial function in a fitting procedure to provide a nonlinear mapping between performance scores and the distance. Fig. 7 depicts that the fitting curves and scatter diagrams are the correlational relationships between performance scores and the distance from spike metrics. As illustrated in Fig. 7(a)-(d), each sample point refers to one testing spike stream,

TABLE II
TIME COMPLEXITY OF SPIKE METRICS.

Methods	Simulating data		MNIST-DVS		DVS rec		Color rec	
	ACT	ANS	ACT	ANS	ACT	ANS	ACT	ANS
KMST [17]	2.623	0.381	45.63	0.039	135.8	0.035	129.7	0.038
KMST-P [23]	2.317	0.236	46.27	0.038	137.3	0.038	126.4	0.039
ASTSM	**0.121**	**8.325**	**2.172**	**0.692**	**7.432**	**0.672**	**7.811**	**0.631**

and the curves display the trends of compression ratios and quantified distance for our ASTSM, ASTSM$^+$ and the state-of-the-art methods [17], [23]. As is shown in Fig. 7(e)-(h), we also present the correlation between the distances and the PSNRs, which are computed by measuring reconstruction images on the DVS reconstruction dataset [55] (i.e. bicycle, night drive, and sun). Besides, we further report intensity reconstruction on the color event camera dataset [58] (i.e. city drive, indoor office, and dancing) using the events-to-video algorithm [56] in Fig. 7(i)-(h). We can see that scatter sample points for our ASTSM are closer to the fitting curves in Fig. 7(c), (g), and (k) than other methodss [17], [23]. Besides, our ASTSM$^+$, using a learnable kernel parameters strategy, can further follow the function of performance score p_s in Fig. 7(d), (h), and (l). In other words, our ASTSM, utilizing the hand-crafted parameters, can obtain better performance than other metrics [17, 23], and our ASTSM$^+$ can become more flexible and pursue a better performance by learning the optimized kernel parameters for a known distribution dataset.

Then, the quantitative evaluation results for spike metrics are given in Table I, we can see that our ASTSM$^+$ can achieve the best performance in contrast to the state-of-the-art methods [17], [23] and our three baselines in three representative datasets. Note that, all spike metrics (i.e., KMST$^+$, KMST-P$^+$, and our ASTSM$^+$), utilizing the learning kernel parameters, can better follow the function of performance score p_s than using the hand-crafted kernel parameters according to personal experience. In particular, our ASTSM, without adopting the learning kernel parameters, has the greater correlation coefficients in four evaluation metrics (i.e., PLCC, SRCC, SRCC, and KRCC) and better reflect the discrimination between raw data and degraded spike streams than other methods [17], [23]. Note that, our approach has a slight improvement in those four criteria compared with other methods [17], [23] on lossy spike coding [24]. In fact, the quantified strategy for spikes is designed for the temporal domain. Single-neuron spike train metrics [17], [23] are also applied to measure spike streams without spatial domain changes. Surprisingly, experimental results on intensity estimation [55], [58] show the superiority of our ASTSM over other spike metrics in those four criteria. The reason for this is because, adding random noise into asynchronous spikes, the degraded operation changes spatio-temporal structure in the MSTPPs. This may be caused by the fact that our ASTSM can better measure the distance between asynchronous spikes in the MSTPPs, especially involving spatio-temporal structure changes.

Evaluating time complexity. In order to evaluate time complexity of our ASTSM compared with the state-of-the-art methods [17], [23], we take average computation times (ACT, s) for all sequences and the average number of spikes processed per second (ANS, Ksp/s) on our provided spike metric dataset in Table II. All tests are implemented in Python3 and run on a Windows Server equipped with an Inter E5 CPU (64bits, 2.6GHz) and 256GB of RAM. As we can see in Table II, our approach is clearly 20$\times$ faster than the works [17], [23]. This is due to the strategy of introducing a 3D Gaussian kernel to spike cuboid architecture instead of spike trains. Actually, the computing elements are significantly decreasing by the sum of spike cuboids rather than one by one pixel. Meanwhile, we can define a reasonable window that can maintain performance while reducing time complexity, which need not all pairwise comparisons if spikes are far away. In fact, the latency is a crucial characteristic for many applications requiring fast and real-time reaction, which includes visual navigation in autonomous driving [10] and ego-motion estimation in mobile robotics [8], [9], [11]. In short, Our ASTSM outperforms the state-of-the-art methods and achieves significant improvement in computational efficiency.

E. Discussion

In fact, an effective and robust spike metric will further highlight the potential of event-based signal processing towards many practical applications. Here, we further discuss the major limitation of the proposed approach.

Our approach provides the quantified distance between asynchronous spikes on a non-normalized set, which depends on spike numbers, kernel parameters, and time length of spike stream. Given an unseen distribution for a novel dataset, our approach fails to build an optimization model to learn the kernel parameters and only utilizes hand-crafted kernel parameters via personal experience. Besides, asynchronous spatio-temporal spikes from event cameras are sparse and discrete points rather than traditional structured frames or videos, which are normalized in quality assessment for conventional cameras. Although, our ASTSM can obtain impressive performance to quantify the distance between raw data and degraded spike streams. Investigating a robust method to regularize any number and time length into a uniform formulation is also interesting and of large applicability for spike metrics. Hence, it has promising potentials to further explore in the future.

VI. CONCLUSION

This paper proposes an asynchronous spatio-temporal spike metric (ASTSM) considering both spatio-temporal structural properties and polarity attribute for event cameras. Our goal is to illustrate that asynchronous stream in the marked spatio-temporal point processes (MSTPPs) is possible to be quantified and measured, and our approach can measure the distance between raw data and degraded spike streams. The experimental results demonstrate that the proposed approach outperforms the state-of-the-art methods and achieves significant improvement in computational efficiency, especially better describing the changes involving spatio-temporal structure and polarity attribute. In particular, we believe this work is a major step towards building an effective spike metric for event-based signal processing applied to neuromorphic engineering, data compression, and machine learning to event-based vision.

REFERENCES

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

[2] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, "A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor," *IEEE J. Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014.

[3] D. P. Moeys, F. Corradi, C. Li, S. A. Bamford, L. Longinotti, F. F. Voigt, S. Berry, G. Taverni, F. Helmchen, and T. Delbruck, "A sensitive dynamic and active pixel vision sensor for color or neural imaging applications," *IEEE Trans. Biomed. Circuits Syst.*, vol. 12, no. 1, pp. 123–136, 2017.

[4] L. A. Camuñas-Mesa, T. Serrano-Gotarredona, S.-H. Ieng, R. Benosman, and B. Linares-Barranco, "Event-driven stereo visual tracking algorithm to solve object occlusion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 9, pp. 4223–4237, 2018.

[5] D. R. Valeiras, X. Clady, S.-H. Ieng, and R. Benosman, "Event-based line fitting and segment detection using a neuromorphic visual sensor," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1218–1230, 2018.

[6] A. D. Rast, S. V. Adams, S. Davidson, S. Davies, M. Hopkins, A. Rowley, A. B. Stokes, T. Wennekers, S. Furber, and A. Cangelosi, "Behavioral learning in a cognitive neuromorphic robot: an integrative approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6132–6144, 2018.

[7] G. Cohen, S. Afshar, G. Orchard, J. Tapson, R. Benosman, and A. van Schaik, "Spatial and temporal downsampling in event-based visual classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 5030–5044, 2018.

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

[9] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 10, pp. 2402–2412, 2018.

[10] J. Li, S. Dong, Z. Yu, Y. Tian, and T. Huang, "Event-based vision enhanced: a joint detection framework in autonomous driving," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2019.

[11] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Unsupervised event-based learning of optical flow, depth, and egomotion," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019.

[12] L. Pan, C. Scheerlinck, X. Yu, R. Hartley, M. Liu, and Y. Dai, "Bringing a blurry frame alive at high frame-rate with an event camera," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019.

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

[14] F. Paredes-Valles, K. Y. W. Scheper, and G. C. H. E. De Croon, "Unsupervised learning of a hierarchical spiking neural network for optical flow estimation: From events to global motion perception," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.

[15] G. Gallego, T. Delbruck, 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 Trans. Pattern Anal. Mach. Intell.*, 2020.

[16] K. A. Boahen, "Point-to-point connectivity between neuromorphic chips using address events," *IEEE Trans. Circuits Syst. II. Analog Digit. Signal Process.*, vol. 47, no. 5, pp. 416–434, 2000.

[17] I. M. Park, S. Seth, A. R. Paiva, L. Li, and J. C. Principe, "Kernel methods on spike train space for neuroscience: a tutorial," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 149–160, 2013.

[18] J. A. González, F. J. Rodríguez-Cortés, O. Cronie, and J. Mateu, "Spatio-temporal point process statistics: A review," *Spatial Statistics*, vol. 18, pp. 505–544, 2016.

[19] J. F. Maya-Vetencourt, D. Ghezzi, M. R. Antognazza, E. Colombo, M. Mete, P. Feyen, A. Desii, A. Buschiazzo, M. Di Paolo, S. Di Marco *et al.*, "A fully organic retinal prosthesis restores vision in a rat model of degenerative blindness," *Nature Materials*, vol. 16, no. 6, p. 681, 2017.

[20] N. P. Shah, S. Madugula, E. Chichilnisky, J. Shlens, and Y. Singer, "Learning a neural response metric for retinal prosthesis," *Proc. Int. Conf. Learn. Represent.*, 2018.

[21] J. Aljadeff, B. J. Lansdell, A. L. Fairhall, and D. Kleinfeld, "Analysis of neuronal spike trains, deconstructed," *Neuron*, vol. 91, no. 2, pp. 221–259, 2016.

[22] T. Tezuka, "Multineuron spike train analysis with r-convolution linear combination kernel," *Neural Netw.*, vol. 102, pp. 67–77, 2018.

[23] S. Dong, Z. Bi, Y. Tian, and T. Huang, "Spike coding for dynamic vision sensor in intelligent driving," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 60–71, 2019.

[24] Y. Fu, J. Li, S. Dong, Y. Tian, and T. Huang, "Spike coding: Towards lossy compression for dynamic vision sensors," in *Proc. IEEE Data Compression Conf.*, 2019.

[25] D. Neil, M. Pfeiffer, and S.-C. Liu, "Phased lstm: Accelerating recurrent network training for long or event-based sequences," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3882–3890.

[26] S. B. Shrestha and G. Orchard, "Slayer: Spike layer error reassignment in time," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1419–1428.

[27] Y. Sekikawa, K. Hara, and H. Saito, "Eventnet: Asynchronous recursive event processing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3887–3896.

[28] D. Gehrig, A. Loquercio, K. G. Derpanis, and D. Scaramuzza, "End-to-end learning of representations for asynchronous event-based data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019.

[29] S. Xiao, M. Farajtabar, X. Ye, J. Yan, L. Song, and H. Zha, "Wasserstein learning of deep generative point process models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3247–3257.

[30] S. Xiao, H. Xu, J. Yan, M. Farajtabar, X. Yang, L. Song, and H. Zha, "Learning conditional generative models for temporal point processes," in *Proc. AAAI Conf. on Artificial Intell.*, 2018.

[31] J. Yan, X. Liu, L. Shi, C. Li, and H. Zha, "Improving maximum likelihood estimation of temporal point process via discriminative and adversarial learning." in *Proc. Int. Joint Conf. Artificial Intell.*, 2018, pp. 2948–2954.

[32] S. Xiao, J. Yan, M. Farajtabar, L. Song, X. Yang, and H. Zha, "Learning time series associated event sequences with recurrent point process networks," *IEEE Trans. Neural Netw. Learn. Syst.*, 2019.

[33] G. Gallego, M. Gehrig, and D. Scaramuzza, "Focus is all you need: Loss functions for event-based vision," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 12 280–12 289.

[34] M. Pfeiffer and T. Pfeil, "Deep learning with spiking neurons: Opportunities and challenges," *Frontiers in Neuroscience*, vol. 12, 2018.

[35] A. I. Maqueda, A. Loquercio, G. Gallego, N. García, and D. Scaramuzza, "Event-based vision meets deep learning on steering prediction for self-driving cars," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5419–5427.

[36] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1346–1359, 2016.

[37] A. Sironi, M. Brambilla, N. Bourdis, X. Lagorce, and R. Benosman, "Hats: Histograms of averaged time surfaces for robust event-based object classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1731–1740.

[38] M. C. van Rossum, "A novel spike distance," *Neural Computat.*, vol. 13, no. 4, pp. 751–763, 2001.

[39] A. R. Paiva, I. Park, and J. C. Príncipe, "A reproducing kernel hilbert space framework for spike train signal processing," *Neural Computat.*, vol. 21, no. 2, p. 424, 2009.

[40] N. Fisher and A. Banerjee, "A novel kernel for learning a neuron model from spike train data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010.

[41] C. Houghton and K. Sen, "A new multineuron spike train metric," *Neural Computat.*, vol. 20, no. 6, pp. 1495–1511, 2008.

[42] A. J. Brockmeier, J. S. Choi, E. G. Kriminger, J. T. Francis, and J. C. Principe, "Neural decoding with kernel-based metric learning." *Neural Computat.*, vol. 26, no. 6, pp. 1080–1107, 2014.

[43] E. Torre, C. Canova, M. Denker, G. Gerstein, M. Helias, and S. GrãN, "Asset: Analysis of sequences of synchronous events in massively parallel spike trains," *Plos Comput. Biology.*, vol. 12, no. 7, p. e1004939, 2016.

[44] J. V. Toups, J.-M. Fellous, P. J. Thomas, T. J. Sejnowski, and P. H. Tiesinga, "Finding the event structure of neuronal spike trains," *Neural Computat.*, vol. 23, no. 9, p. 2169, 2011.

[45] A. Ignacio and C. Margarita, "Asynchronous corner detection and tracking for event cameras in real-time," *IEEE Robot. Autom. Lett.*, 2018.

[46] C. Posch, T. Serrano-Gotarredona, B. Linares-Barranco, and T. Delbruck, "Retinomorphic event-based vision sensors: Bioinspired cameras with spiking output," *Proceedings of the IEEE*, vol. 102, no. 10, pp. 1470–1484, 2014.

[47] R. Kempter, W. Gerstner, and J. L. V. Hemmen, "Spike-based compared to rate-based hebbian learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 1999.

[48] J. P. Cunningham, M. Y. Byron, K. V. Shenoy, and M. Sahani, "Inferring neural firing rates from spike trains using gaussian processes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 329–336.
[49] A. Khodamoradi and R. Kastner, "O(n)-space spatiotemporal filter for reducing noise in neuromorphic vision sensors," *IEEE Trans. Emerging Topics Comput.*, vol. PP, no. 99, pp. 1–1, 2018.
[50] C. Posch, D. Matolin, and R. Wohlgenannt, "A qvga 143 db dynamic range frame-free pwm image sensor with lossless pixel-level video compression and time-domain cds," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 259–275, 2011.
[51] R. Ghosh, A. Gupta, A. Nakagawa, A. Soares, and N. Thakor, "Spatiotemporal filtering for event-based action recognition," *arXiv*, 2019.
[52] R. Ghosh, A. Gupta, S. Tang, A. Soares, and N. Thakor, "Spatiotemporal feature learning for event-based vision," *arXiv*, 2019.
[53] R. F. Teixeira and N. J. Leite, "A new framework for quality assessment of high-resolution fingerprint images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 10, pp. 1905–1917, 2017.
[54] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *J. Mach. Learn. Research*, vol. 12, no. Jul, pp. 2211–2268, 2011.
[55] C. Scheerlinck, N. Barnes, and R. Mahony, "Continuous-time intensity estimation using event cameras," in *Proc. Asian Conf. Comput. Vis.* Springer, 2018.
[56] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3857–3866.
[57] T. Serrano-Gotarredona and B. Linares-Barranco, "The MNIST-DVS database," *http://www2.imsecnm.csic.es/caviar/MNISTDVS.html*, 2014.
[58] C. Scheerlinck, H. Rebecq, T. Stoffregen, N. Barnes, R. Mahony, and D. Scaramuzza, "Ced: Color event camera dataset," in *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2019.
[59] S. Wang, K. Gu, X. Zhang, W. Lin, S. Ma, and W. Gao, "Reduced-reference quality assessment of screen content images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 1, pp. 1–14, 2018.
[60] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

Jianing Li is currently a PH.D. student in the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing, China.

He received a Lixin Tang Scholarship from the Chongqing University in 2016. His research interests include neuromorphic vision, machine learning, neuromorphic engineering and spatio-temporal point processes.

Yihua Fu received the B.S. degree in the School of Microeletronics, Tianjin University, Tianjin, China, in 2017. She is currently working towards the master's degree at the school of Electronic and Computer Engineering, Peking University, Shenzhen, China.

Her research interests include data mining, event-based signal processing, machine learning and neuromorphic engineering.

Siwei Dong received the B.S. degree from the College of Computer Science, Chongqing University, Chongqing, China in 2012, and the Ph.D. degree from the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing, China in 2019.

His current research interests include video coding and neuromorphic computing.

Zhaofei Yu received the B.S. degree from the Hong Shen Honors School, College of Optoelectronic Engineering, Chongqing University, Chongqing, China in 2012, and the Ph.D. degree from the Automation Department, Tsinghua University, Beijing, China in 2017.

He is a Post-Doctoral Fellow with the National Engineering Laboratory of Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing. His current interests include artificial intelligence, brain-inspired computing, and computational neuroscience.

Tiejun Huang (M'01-SM'12) is a professor with the Department of Computer Science, School of EE&CS, Peking University, and the Director of the Beijing Academy for Artificial Intelligence. His research areas include visual information processing and neuromorphic computing.

He published two books, 200+ peer-reviewed papers on leading journals and conferences, holds 50+ granted patents, and is the co-editor of 4 ISO/IEC standards, 5 National standards of China and 4 IEEE standards. Professor Huang received the Ph. D. degree in pattern recognition and intelligent system from Huazhong (Central China) University of Science and Technology in 1998, was awarded the Distinguished Young Scholar by the National Natural Science Foundation of China in 2014, the Distinguished Professor of the Chang Jiang Scholars Program by the Ministry of Education of China in 2015. Professor Huang received National Award for Science and Technology of China (Tier-2) for three times. He is a Fellow of CAAI, CCF, the secretary general of the Artificial Intelligence Industry Technology Innovation Alliance, and vice chair of the China National General Group on AI Standardization.

Yonghong Tian (S'00-M'01-SM'12) is currently a Boya Distinguished Professor with the Department of Computer Science and Technology, Peking University, China, and is also the deputy director of Artificial Intelligence Research Center, PengCheng Laboratory, Shenzhen, China. His research interests include neuromorphic vision, brain-inspired computation and multimedia big data.

He is the author or coauthor of over 200 technical articles in refereed journals such as IEEE TPAMI/TNNLS/TIP/TMM/TCSVT/TKDE/TPDS, ACM CSUR/TOIS/TOMM and conferences such as NeurIPS/CVPR/ICCV/AAAI/ACMMM/WWW. Prof. Tian was/is an Associate Editor of IEEE TCSVT (2018.1-), IEEE TMM (2014.8-2018.8), IEEE Multimedia Mag. (2018.1-), and IEEE Access (2017.1-). He co-initiated IEEE Int'l Conf. on Multimedia Big Data (BigMM) and served as the TPC Co-chair of BigMM 2015, and aslo served as the Technical Program Co-chair of IEEE ICME 2015, IEEE ISM 2015 and IEEE MIPR 2018/2019, and General Co-chair of IEEE MIPR 2020 and ICME2021. He is the steering member of IEEE ICME (2018-) and IEEE BigMM (2015-), and is a TPC Member of more than ten conferences such as CVPR, ICCV, ACM KDD, AAAI, ACM MM and ECCV. He was the recipient of the Chinese National Science Foundation for Distinguished Young Scholars in 2018, two National Science and Technology Awards and three ministerial-level awards in China, and obtained the 2015 EURASIP Best Paper Award for Journal on Image and Video Processing, and the best paper award of IEEE BigMM 2018. He is a senior member of IEEE, CIE and CCF, a member of ACM.