NeuroZoom: Denoising and Super Resolving Neuromorphic Events and Spikes

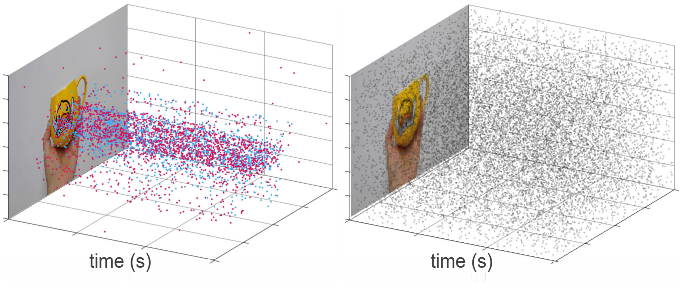
Peiqi Duan, Yi Ma, Xinyu Zhou, Xinyu Shi, Zihao W. Wang,
Tiejun Huang, *Senior Member, IEEE*, Boxin Shi[‡], *Senior Member, IEEE*

Abstract—Neuromorphic cameras are emerging imaging technology that has advantages over conventional imaging sensors in several aspects including dynamic range, sensing latency, and power consumption. However, the signal-to-noise level and the spatial resolution still fall behind the state of conventional imaging sensors. In this paper, we address the denoising and super-resolution problem for modern neuromorphic cameras. We employ 3D U-Net as the backbone neural architecture for such a task. The networks are trained and tested on two types of neuromorphic cameras: a dynamic vision sensor and a spike camera. Their pixels generate signals asynchronously, the former is based on perceived light changes and the latter is based on accumulated light intensity. To collect the datasets for training such networks, we design a display-camera system to record high frame-rate videos at multiple resolutions, providing supervision for denoising and super-resolution. The networks are trained in a noise-to-noise fashion, where the two ends of the network are unfiltered noisy data. The output of the networks has been tested for downstream applications including event-based visual object tracking and image reconstruction. Experimental results demonstrate the effectiveness of improving the quality of neuromorphic events and spikes, and the corresponding improvement to downstream applications with state-of-the-art performance.

Index Terms—Neuromorphic camera, vidar camera, high-quality imaging

1 Introduction

Neuromorphic computer vision is an emerging research field that executes visual tasks using neuromorphic sensory data and algorithms [1]–[3]. Neuromorphic sensors are designed to imitate biological neurons and synapses, which only perform work when there are events/spikes to process [4]. Such a sensor has high efficiency, low latency and high dynamic range (HDR) [1], [5]–[7]. However, the output events/spikes are asynchronous spatio-temporal ”point clouds” (Fig. 1) that are fundamentally different from conventional images. The novel modality brings novel challenges from a signal processing perspective. Restoration and enhancement of neuromorphic signals are fundamental problems yet are different from their image-based counterparts. Previous works address the denoising and super resolution problem of events and demonstrate the effectiveness for downstream applications [8], [9]. In this work, we propose a framework for *Neuromorphic signal Denoising and Super-Resolution (NDSR)*.



(a) Event sensing (b) Spike sensing

Figure 1: 3D spatial-temporal views of real event/spike data, which have unconventional forms that differ from traditional frame-based RGB images. Blue/red points indicate positive/negative events, gray points indicate spikes.

There are two types of neuromorphic cameras, namely differential-based [5], [7] and integral-based [2] types, according to how their signals relate with scene radiance. Differential-based neuromorphic sensors, also known as event cameras or Dynamic Vision Sensors (DVS) [6], [7], enable each pixel to only compare current and last light intensity states in log-scale and fire a binary-signed event whenever the log-intensity variation exceeds the preset thresholds [1], [7], [10]. This type of cameras is suitable for dynamic visual scenarios thanks to their high speed ($\sim 10\mu s$), and avoidance of absolute intensity values for static scenes (Fig. 1(a)). Integral-based neuromorphic sensors, also named spike cameras or the Vidar [2], enable each pixel to continuous exposure and fire a binary-signed spike whenever the intensity accumulation exceeds a threshold [11]. Spike cameras record the light intensity in spatial-temporal domain at high speed ($25\mu s$). High-frequency spike triggering corresponds to high intensity, and vice versa (Fig. 1(b)). Neuromorphic cameras have shown promising potential in solving video restoration [12]–[15], 3D vision [16], and robotics [17] tasks due to the low latency and HDR characteristics.

In spite of the popularity of neuromorphic-based vision, current event and spike sensor prototypes still bear low

[‡] *Corresponding author: shiboxin@pku.edu.cn*

P. Duan, Y. Ma, X. Shi, T. Huang, and B. Shi are with National Key Laboratory for Multimedia Information Processing and National Engineering Research Center of Visual Technology, School of Computer Science, Peking University.
X. Zhou is with the National Key Lab of General Artificial Intelligence, School of Intelligence Science and Technology, Peking University.
Z. W. Wang is with Northwestern University.
Project page: https://sites.google.com/view/EventZoom-NeuroZoom

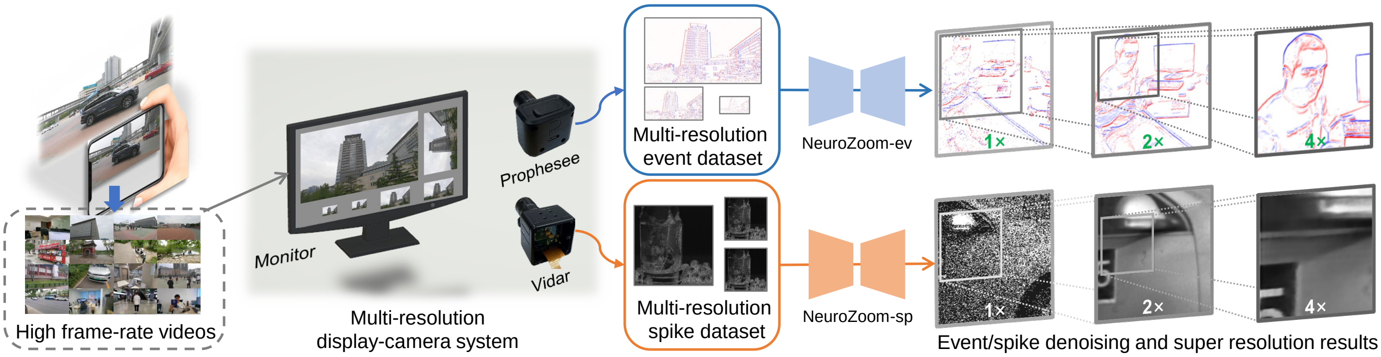

High frame-rate videos
Monitor
Multi-resolution display-camera system
Prophesee
Vidar
Multi-resolution event dataset
Multi-resolution spike dataset
NeuroZoom-ev
NeuroZoom-sp
1×
2×
4×
Event/spike denoising and super resolution results

Figure 2: We propose a framework that uses a display-camera system to collect real-captured multi-resolution event/spike datasets, which enables the proposed NeuroZoom-ev/-sp networks to learn to perform event/spike denoising and super resolution. The architecture of both two networks is shared. 1× data show that raw neuromorphic data suffer from low spatial resolution, noise, and data loss.

spatial resolution (*e.g.*, DAVIS346 [6] is 346 × 260 and the first generation of Vidar [2] is 400 × 250) and nonnegligible sensor noise, event sensors even suffer from event loss due to their special response and transmission mechanisms [1] (Fig. 2). A particular body of literature has attended to event-to-image [8], [18], [19] or spike-to-image reconstruction [14], [20], and shown that image-based visual algorithm can perform well on event- or spike-reconstructed images. Nonetheless, the gap between the spatial resolution of images captured by a modern camera and the spatial resolution of neuromorphic signals affects the practicability of the methods, and the severe noise of neuromorphic signals brings inevitable errors to their downstream tasks. Gehrig *et al*. [21] also verifies that directly increasing the spatial resolution of event sensors will lead to the inevitable event missing and the increase of pixel response delay. Therefore, a "compact" restoration and enhancement algorithm as the post-processing of NDSR is desired.

For event cameras, state-of-the-art event restoration solutions rely on the intensity signal [22]–[24]. In EDnCNN [22], same-resolution images are used to label events for training dataset collection, and a classification network is further used for event denoising. However, event labeling can only remove wrongly-fired events and cannot recover unfired events, even with previous event removal filters [25]–[27]. Guided Event Filtering (GEF) [23], [24] retrieves missing events and provides super-resolution (SR) but requires high-quality, high-resolution (HR) images synchronized with events. Its performance is highly dependent on accurate optical flow estimation, which is computationally expensive. For spike cameras, existing methods [14], [20], [28] reconstruct images without specifically dealing with noise. Learning-based methods [29], [30] train the networks with simulated datasets where the real-simulation gap cannot be ignored. MGSR [31] enables image SR for spike cameras, but has a minute-level runtime per frame. In short, the bottlenecks of current NDSR methods include 1) reliance on high-quality images and lack of available learning datasets; 2) lack of reliable methods to integrate denoising and SR into a unified framework; 3) the expensive running cost.

Real-data driven is an available approach to study signal degradation avoiding the real-simulation gap, which has been verified in the field of image restoration [32], [33]. Our preliminary work EventZoom [9] introduced this fashion into event-based NDSR for the first time. We implemented a display-camera system to collect a multi-resolution event dataset, and trained a network building upon 3D U-Net [34] in a noise-to-noise fashion without ground truth annotation, while preserving computational efficiency.

However, the preliminary solution [9] has several limitations: 1) Network. Due to the lack of multi-scale constraint and cross-scale feature fusion to handle large-scale upsampling, the network can't handle 4× event SR, and it does not support handling spikes. 2) Dataset. Low resolution of DAVIS346 and low quality of source videos limit the 4× event SR capability, and a multi-resolution spike dataset is not yet available. 3) Validation. EventZoom [9] is evaluated by qualitative tests and downstream applications, because obtaining noise-free references for real events is challenging.

Overview of this work: This paper extends [9] to address above limitations and makes the following contributions:

- We propose a unified framework NeuroZoom to solve the NDSR problem for both events and spikes, the proposed 3D U-Net backbone is able to exploit temporal coherence which is especially important to achieve NDSR and even edge-filling. The network in [9] has been updated with a pyramid architecture and cross-scale feature fusion enabling both 2× and 4× SR jointly, bringing performance improvement.
- We upgrade our display-camera system to spike cameras and collect a multi-resolution spike dataset. The multi-resolution event dataset is expanded with newly captured source videos with high quality and an event camera with larger resolution (Prophesee Gen4.0 [35]), enabling to directly learn 4× event SR.
- We collect a full-reference test dataset for events by a controllable camera system, and quantitatively benchmark the restoration performance for existing NDSR methods. The quantitative and qualitative analyses for downstream applications are updated with new datasets and state-of-the-art methods.

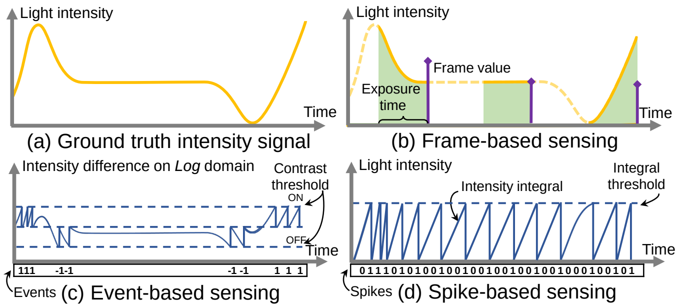
2 Related Works

Event denoising and super resolution. Existing works were mainly concerned with background activity noise produced by temporal noise and junction leakage currents [7],

[22], [25]–[27]. Liu *et al*. [27] proposed a denoising filter based on spatiotemporal correlation. Wang *et al*. [36] proposed to filter events by their motion association likelihood. This is based on an assumption that events are triggered by edge motion and therefore shall follow the same spatiotemporal motion projection within a local window if valid [23], [37], [38]. GEF [23], [24] uses motion compensation (MC) to align image and event signals, and employs guided image filtering to optimize the mutual structure between the low-resolution (LR) event and HR image signals. By leveraging this approach, GEF can perform super-resolution on the event signal up to the image resolution when the image signal has higher spatial resolution than the event signal. Although MC is highly useful for event processing [37], [39], [40], the computational complexity is beyond practical for downstream visual tasks. Another pathway for event-based NDSR is first by means of event-to-intensity conversion [8], [41]. The generated high quality images can then be converted back to events via video-to-events simulators [42]–[44]. The runtime and the real-simulation gap [45] are the main limitations. We introduce a learning-based and real-data driven strategy to enhance and restore event signals with computational efficiency.

Spike camera imaging. Spike cameras asynchronously record the intensity accumulation by spike streams at a high speed and high dynamic range . Due to these unique characteristics, the spikes are naturally useful to reconstruct high frame rate videos [14], [20], [28]–[31] or HDR images [15]. Two basic restoration models are the Texture From inter-spike Interval (TFI) [28] and the Texture From Playback (TFP) [28]. TFI infers the light intensity by calculating inter-spike intervals and TFP restores it by accumulating the spike numbers of a brief period. Both basic methods are not robust to noise. To improve the reconstruction quality, Zhu *et al*. [20] used a retina-like framework (*i.e.*, SNN) to match the characteristic of spike signals, they also proposed NeuSpike-Net [30] that combined events and spike to reconstruct HDR images like bio-inspired sampling. Zhao *et al*. [31] proposed a multi-frame based method MGSR to handle the challenges of both noise and high-speed motion. Nonetheless, these methods are computationally expensive and are not specially designed to consider unavoidable noise with spikes. We propose a compact spike restoration and enhancement method with both denoising and SR.

Neuromorphic camera systems and datasets. While the majority of existing datasets have addressed various visual tasks, very few of them focused on event-based NDSR, let alone spike-based NDSR. DVSNOISE20 [22] proposed a noise annotation approach by deriving an event probability mask using APS frames and IMU motion data. The dataset in [18] used HR smartphone videos as reference but did not convert them back to raw data form to obtain intensity information required for event-based NDSR. Both MVSEC [46] and RGB-DAVIS [23] have provided HR machine vision images up to 2× and 8× respectively. Particularly, RGB-DAVIS leveraged a beam splitter to collocate an HR RGB camera and LR DAVIS event camera [23]. There has not been a multi-resolution event dataset provided in the literature due to the significant challenges in camera calibration and the lack of HR event camera prototypes. In event datasets from [47], a display-camera system was used to convert existing video datasets to event datasets. We use a similar setup with upgraded hardware in both the display and event cameras, and collect a high frame-rate video dataset to minimize temporal aliasing caused by large motion.



(a) Ground truth intensity signal (b) Frame-based sensing (c) Event-based sensing (d) Spike-based sensing

Figure 3: An illustration of different sensing mechanisms (intensity, events, and spikes) and the corresponding visual signal outputs. Given a latent intensity signal, frame-based sensors sample the average intensity over the exposure time at a fixed frame rate, while neuromorphic sensors record intensity differences (events) or intensity integrals (spikes) asynchronously with binary-signed neuromorphic signals.

Neuromorphic neural models. Neuromorphic signals are bio-inspired visual signals resembling the form of asynchronous neural spike trains. Several bio-inspired learning architectures have been proposed for event- and spike-based learning, including SNNs [20], [48], LSTM/RNNs [49], [50], and MLPs [51], [52]. CNNs are widely adopted for event-based NDSR-related tasks. Wang *et al*. [53] proposed to use the sigmoid function to approximate the intensity-event relation, and employed a residual net for image enhancement. Before performing convolutions, the input events were first binned or stacked into event frames which induced temporal interruptions [41], [53], [54]. This issue was alleviated by explicitly incorporating inter-stack flow estimation modules [41], [55]. As shown in GEF [23], [24], 2D convolutional SR nets did not perform well on binned event frames as the activation sites are sparse. Messikommer *et al*. [56] adopted sparse convolutions with an asynchronous activation mechanism for high-level visual tasks. Gehrig *et al*. [57] proposed volumetric spatio-temporal tensors to form an event feature space that is trained w.r.t. specific tasks. For NDSR, we employ 3D U-Net [34] as it has a volumetric encoder-decoder structure and performs 3D convolutions.

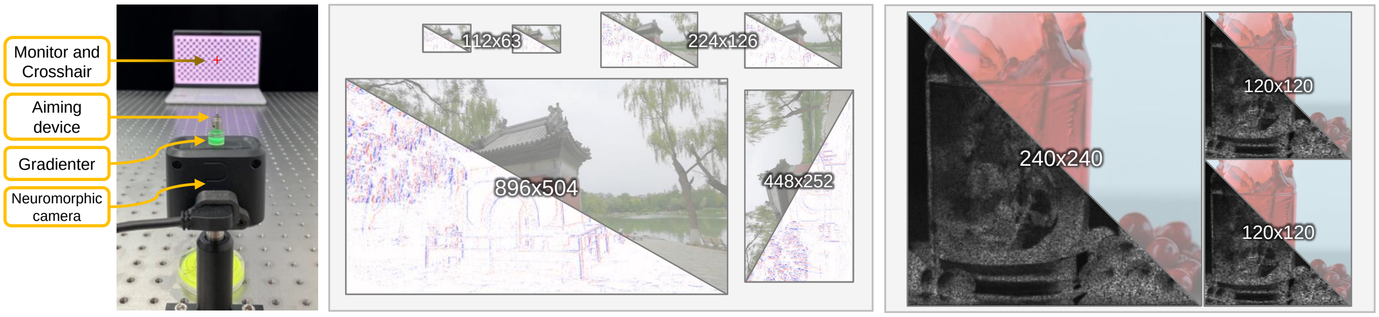
3 NeuroZoom Approach

3.1 Neuromorphic signal formation model

We first demonstrate the event/spike formation model and its relationship to the image-based counterpart. In image denoising and super resolution, the basic formation model assumes that the LR image $\hat{I}^{\mathrm{LR}}$ is the result of a downscaling operation from a degraded HR image I^{HR} added by noise:

$$\hat{I}^{\mathrm{LR}} = (I^{\mathrm{HR}} * k) \downarrow_{\mathrm{s}} + n_{\mathrm{image}}, \tag{1}$$

where k denotes a degradation for isotropic Gaussian blur [58], which can be ignored since the latent clear image is needed to formulate neuromorphic signals. $\downarrow_{\mathrm{s}}$ is a downscaling operation with a scale factor of s, and n_{image} represents



(a) Setup (b) View of Prophesee event camera (1280 × 720) (c) View of Vidar spike camera (400 × 250)

Figure 4: (a) We build a display-camera system to investigate the formation and degradation of events and spikes. (b) RGB video source views and corresponding calibrated event views. The display is divided into 6 segments and 4 resolution-levels. (c) Video source and calibrated spike views. The display is divided into 3 segments and 2 resolution-levels.

the additive image noise. We use $\hat{I}^{\text{LR}}$ to denote I^{LR} has been noise corrupted. We formulate the generation of events and spikes via latent images in spatiotemporal field as follows:

Event formation. For the case of events, the event sensor output at time t_0 can be described as (refer to Fig. 3(c)):

$$E_{t_0} = \Gamma\{\log(\frac{I_{t_0}+b}{I_{t_0-1}+b}), \epsilon\}, \tag{2}$$

where $\Gamma\{\theta,\epsilon\}$ represents the conversion function from log-intensity to events, and b is an offset value to prevent $\log(0)$. $\Gamma\{\theta,\epsilon\}=1$ when $\theta \geq \epsilon$, indicating a positive event; $\Gamma\{\theta,\epsilon\}=-1$ when $\theta \leq -\epsilon$, indicating a negative event; and $\Gamma\{\theta,\epsilon\}=0$ when $|\theta|<\epsilon$, indicating that no event has been fired. The dead pixels can be interpreted as ϵ being significantly low or high.

Equation (6) is the noise-free model of the intensity-to-event conversion. The event formation model considering both the downscaling and noise can be represented as:

$$\hat{E}^{\text{LR}}_{t_0} = \Gamma\Big\{\log\big(\frac{(I^{\text{HR}}_{t_0})\downarrow_{\text{s}}+b}{(I^{\text{HR}}_{t_0-1})\downarrow_{\text{s}}+b}\big), \epsilon+n_{\text{event}}\Big\}, \tag{3}$$

where n_{event} represents the perturbation noise pivoted at the firing threshold. According to previous studies [6], [7], n_{event} can be viewed as a Gaussian random process with a mean value of 0. Note that this model does not consider all the event sensor noise types but can be used to explain several experimental observations [7] and has been adopted in previous event simulator for generating noise-corrupted events [42]. Our goal is to recover the latent HR event signal $E^{\text{HR}}_{t_0}=\Gamma\{\log(\frac{I^{\text{HR}}_{t_0}+b}{I^{\text{HR}}_{t_0-1}+b}),\epsilon\}$ from the LR noisy signal $\hat{E}^{\text{LR}}_{t_0}$.

Spike formation. For the case of spikes, the spike sensor output at time t_0 can be described as (refer to Fig. 3(d)):

$$S_{t_0} = \Big\lfloor\Big(\int_0^{t_0} I_t \text{d}t\Big)/\varphi\Big\rfloor - \Big\lfloor\Big(\int_0^{t_0-1} I_t \text{d}t\Big)/\varphi\Big\rfloor. \tag{4}$$

$\Big\lfloor\Big(\int_0^{t_0} I_t\text{d}t\Big)/\varphi\Big\rfloor$ denotes that whenever the integral of latent images $\int_0^{t_0} I_t\text{d}t$ exceeds the threshold φ, the spike count between $t=0$ and $t=t_0$ for each pixel can be recorded by a round-down process. In this way, the matrix S_{t_0} represents the difference of spike count between time t_0 and t_0-1, indicating whether each pixel has a spike triggered at time $t=t_0$ (values of 1 in the matrix S_{t_0} represents a spike, and 0 represents that no spike is triggered).

The intensity-to-spike model considering both the downscaling and noise degradation can be expressed as:

$$\begin{aligned}\hat{S}^{\text{LR}}_{t_0} = &\Big\lfloor\Big(\int_0^{t_0} (I^{\text{HR}}_t)\downarrow_{\text{s}} \text{d}t\Big)/(\varphi+n_{\text{spike}})\Big\rfloor \\ &-\Big\lfloor\Big(\int_0^{t_0-1} (I^{\text{HR}}_t)\downarrow_{\text{s}} \text{d}t\Big)/(\varphi+n_{\text{spike}})\Big\rfloor.\end{aligned} \tag{5}$$

Here, n_{spike} is the perturbation noise pivoted at φ, which can also be viewed as a Gaussian random process [28]. The aim of spike-based NDSR is to recover $S^{\text{HR}}_{t_0} = \Big\lfloor\Big(\int_0^{t_0} I^{\text{HR}}_t\text{d}t\Big)/\varphi\Big\rfloor - \Big\lfloor\Big(\int_0^{t_0-1} I^{\text{HR}}_t\text{d}t\Big)/\varphi\Big\rfloor$ from the $\hat{S}^{\text{LR}}_{t_0}$.

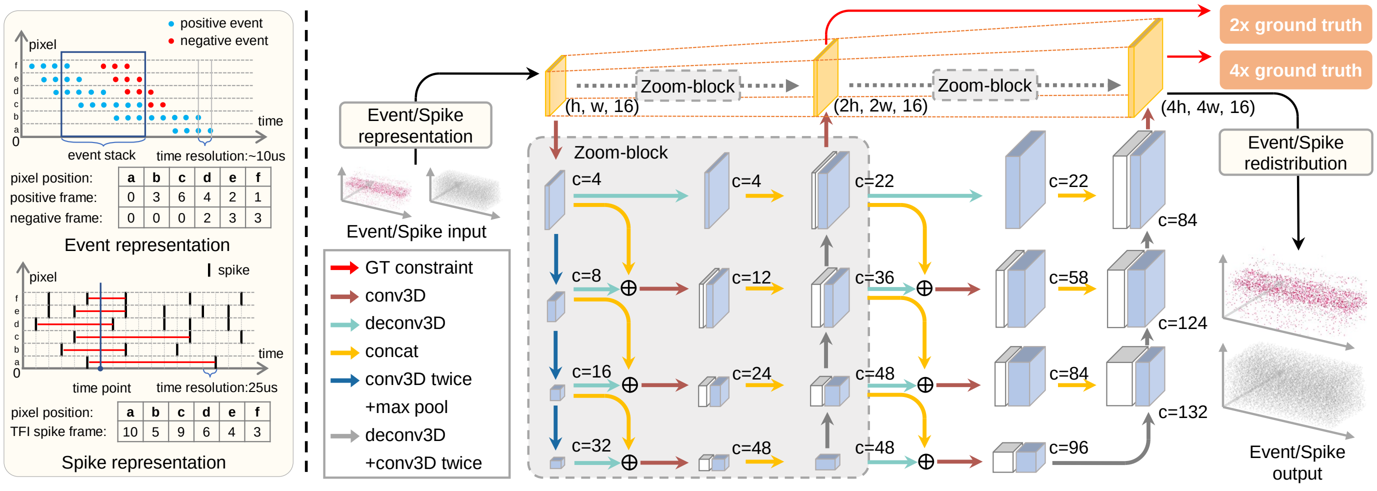
The relationship between events and spikes. According to Huang *et al.* [2], the intensity $I_{t_0}(x,y)$ of a pixel (x,y) can be estimated by taking the reciprocal of the time length between the two adjacent spikes that occur closest to the time point t_0 on the timeline. This time length is also known as the inter-spike interval. We define $\mathcal{T}(\cdot)$ as the process for obtaining the inter-spike intervals from spikes. Then, the matrix $\mathcal{T}(S_{t_0})$ records the inter-spike-intervals for all pixels at time point t_0. The intensity image at t_0 can be calculated as $I_{t_0}=c/\mathcal{T}(S_{t_0})$, where c refers to the maximum dynamic range of the reconstruction [2], [28]. Putting $I_{t_0}=c/\mathcal{T}(S_{t_0})$ into Equation (2), we can convert spikes to events as follows:

$$E_{t_0} = \Gamma\{\log(\frac{c/\mathcal{T}(S_{t_0})+b}{c/\mathcal{T}(S_{t_0-1})+b}), \epsilon\}, \tag{6}$$

which establishes the mathematical relationship between the events and spikes. Using the real-captured spike-event calibrated data [30], we test the effectiveness of the spike-to-event transformation on real data. The conversion results showed in the supplementary material demonstrate that the spike-converted events are highly similar to the real events.

3.2 Display-camera system for NDSR

The recovery from $\hat{E}^{\text{LR}}_{t_0}$ to $E^{\text{HR}}_{t_0}$ and from $\hat{S}^{\text{LR}}_{t_0}$ to $S^{\text{HR}}_{t_0}$ are ill-posed problems as there are many unknown parameters that need to be estimated, including the image degradation kernel k, the threshold value ϵ (for event) and φ (for spike), and the noise n_{event}, n_{spike}. Even when all the unknown parameters are correctly estimated, the surjective property of $\Gamma(\cdot)$ mapping from intensity to event and module process from intensity to spike make NDSR a difficult problem.



pixel position:	a	b	c	d	e	f
positive frame:	0	3	6	4	2	1
negative frame:	0	0	0	2	3	3

pixel position:	a	b	c	d	e	f
TFI spike frame:	10	5	9	6	4	3

(a) (b)

Figure 5: (a) An illustration of event and spike representation we used. For events, we stack events of a time interval into a positive frame and a negative frame, which correspondingly record the integral per pixel for positive/negative events. The time interval is not fixed and depends on the different clip strategies. For spikes, we employ TFI [28] (*i.e.*, use spike interval) to represent features of each time point. The length of the red line segment (the number of time units it spans) represents the time interval between two adjacent spikes. (b) NeuroZoom architecture for both 2× and 4× NDSR. The input LR neuromorphic signals are first represented into a 3D tensor and then fed into two cascaded Zoom-block. Finally, the HR tensor is redistributed to spatiotemporal point clouds.

We develop a display-camera system to observe real-world neuromorphic data at multiple scales to approach NDSR. The system setup is presented in Fig. 4(a), consisting mainly of a display and a neuromorphic camera. We choose the Prophesee Gen 4.0 [35] (1280×720) to capture events and the VidarOne camera [2] to capture spikes. Note the event camera we used has a larger resolution than the DAVIS346 mono camera (346 × 260) [6] used in [9]. An F/1.4 lens is mounted on the cameras. The camera is placed at a distance of $\sim 180cm$ away from the display to avoid lens distortion. To calibrate the alignment between the camera plane and the display plane, we use a gradienter to limit one rotational degree of freedom. The other two degrees of freedom are limited by the collinearity of the camera view center, aiming device and the crosshair on display center. The influence from other light sources is minimized during recording.

For collecting event dataset, we use a HUAWEI P40 pro+ mobile phone to shot 45 high resolution (1280 × 720) and high frame-rate (240*FPS*) video clips as source videos and play them on an AUO80ed display (1920 × 1080, 144*Hz*). Compared with the Need-for-Speed (NFS) dataset [59] used in [9], our new source videos do not contain scenes with low-light, flickering and severe noise, which avoids introducing noise and artifacts to neuromorphic signals from source videos. As Fig. 4(b) shows, the display view is divided into 6 segments and 4 levels with 2 extra-low (1×), 2 low (2×), 1 medium (4×) and 1 high (8×) resolution scales. The original frames are bicubically downsized to alleviate spatial aliasing. The new videos are played at 90*FPS* to avoid exceeding the highest refresh rate of the display.

For spikes, in order to improve the compatibility of the dataset for ultra-high-speed motion scenes that spike cameras are mainly used to shoot, we select 25 videos with slow motion (*i.e.*, Slo-mo) effect (shooting high-speed scenes such as water polo bursting or propeller rotation) from the Internet as video sources. We set them back to 240*FPS* to simulate real high-speed motion on a display with the refresh rate of 360*Hz* (ASUS PG259QNR, 1920 × 1080). As Fig. 4(c) shows, the view is divided into 3 segments and 2 levels with 2 low (1×) and 1 high (2×) resolution scales.

With this setup, we obtain 45 multi-resolution event clips with a total time length of 20 minutes and 25 multi-resolution spike clips with a total time length of 12 minutes. We refer to these newly captured multi-resolution datasets as "Multi-E" and "Multi-S" respectively. The comparison of the differences between events generated by real captured, display-camera system captured, and simulator-simulated ways, and the quantitative comparison for our display-camera training dataset and V2E-simulated training dataset, have been added to the supplementary material.

3.3 Noise-corrupted HR-LR correspondence

Figure 4(b) and Fig. 4(c) show multi-resolution event frame and spike frame examples. It is clear that, despite representing the same motion, LR frames have different appearance due to noise, either events or spikes. For event frames, some edges are missing due to the increase of the event firing threshold caused by n_{event}, while some noisy events are fired at non-edge positions due to the delayed response of the sensor. For spike frames, severe noise covering the full field of view is also caused by n_{spike}. Such randomness make the ground truth data annotation difficult because both the HR and LR signals have been noise-corrupted.

Inspired by [32], we use the noise-to-noise fashion to learn the denoising and SR mappings for neuromorphic signals. For event-based NDSR, we obtain a series of noise-corrupted HR-LR event signal pairs, *i.e.* $(\hat{E}^{\text{LR}}_{(i)}, \hat{E}^{\text{HR}}_{(i)})$. Here, the timestamp t is omitted and replaced by the sample index i. According to our image formation model in Eq. (3), the event data have an expectation of $\mathbb{E}[\hat{E}^{\text{HR}}_{(i)}|\hat{E}^{\text{LR}}_{(i)}] = E^{\text{HR}}_{(i)}$ as the noise-corrupted event signal has a zero-mean noise model [32]. This enables us to train a regressor Ω_E that learns a mapping from noise-corrupted LR event data $\hat{E}^{\mathrm{LR}}$ to noise-free HR event data E^{HR} without ground truth supervision:

$$\underset{\Omega_E}{\operatorname{argmin}} \mathcal{L}\{\Omega_E(\hat{E}^{\mathrm{LR}}_{(i)}), \hat{E}^{\mathrm{HR}}_{(i)}\}, \tag{7}$$

where $\mathcal{L}$ denotes a loss function. In our case, we use the mean squared error loss. In the same way, the regressor Ω_S of spikes can be represented as:

$$\underset{\Omega_S}{\operatorname{argmin}} \mathcal{L}\{\Omega_S(\hat{S}^{\mathrm{LR}}_{(i)}), \hat{S}^{\mathrm{HR}}_{(i)}\}. \tag{8}$$

3.4 Representation of neuromorphic signals

Figure 5(a) shows the data perprocessing we used, *i.e.*, neuromorphic signal representation. For events, we divide both LR and HR raw event stream into 3D tensors $(\mathbb{R}^{h\times w\times 2cn})$ as event stacks and feed them as pairs into the network sequentially, where $2cn$ means the corresponding events of one 3D tensor are divided into cn channels and each part events are summed pixel-wisely corresponding to their polarities (*i.e.*, positive/negative) within two sub-channels. In the training stage, there are two event stream-dividing strategies, *i.e.*, fixing the time interval of each channel and fixing the event number of each channel. We conduct an ablation study on the above strategies and find that the best performance is obtained when the event number of each channel is fixed and $cn = 16$. In the test stage, to ensure consistent channel durations for easy downstream connection, we choose to fix the time interval of each channel. Different channel numbers are also tested by the ablation study and we find setting $cn = 16/32$ and the time interval of each event stack covering a duration of about $11ms$ in the test stage gives reasonable results. The output event stack is rounded to integer values and then redistributed in each channel by assigning a timestamp for each event. We have experimentally analyzed the impact of this redistribution strategy on downstream tasks at different channel time intervals. The redistribution strategy analysis and ablation study are included in the supplementary material.

For spike, we choose TFI [28] to preprocess the raw spike stream. TFI [28] restores intensity value per pixel from the reciprocal of inter-spike-intervals, which can roughly reconstruct the image corresponding to each time point without losing temporal accuracy, despite the image suffers from severe noise. Both LR and HR spikes have a time resolution of $25\mu s$, so we first reconstruct $40000FPS$ frame sequences by TFI, and then clip each 16 frames into a 16-channel spike stack to perform supervision. By inverting the output frame, the interval between two adjacent spikes per pixel at one time point can be obtained, and a binary spike stream can be easily reconverted to be further adapted for downstream applications.

3.5 NeuroZoom network architecture

The network takes as input a spatiotemporal 3D point cloud and outputs its HR enhanced version. The captured neuromorphic signals are mostly sparse in space but dense over time. Inspired by previous study [23] where quantitative results showed 2D-CNN-based SR networks are not suitable for event-based NDSR, we employ 3D convolutions for the purpose of learning spatiotemporal features. The neural network is built upon 3D U-Net [34], as shown in Fig. 5(b) for 4× event-based SR. Compared to other multi-channel 2D-CNN-based approaches, 3D U-Net takes more channels in the time dimension to better exploit temporal coherence.

Inspired by the single image SR method LapSRN [60], we design NeuroZoom as a two-level pyramid architecture with two Zoom-block cascades to learn both the 2× and 4× mappings for event-based NDSR. Two different scales of HR events at the corresponding level are used as the mutli-scale supervision, which enables the network to learn to predict both 2× and 4× SR events in a feed-forward step. Each Zoom-block is built up by a modified 3D U-Net [34] architecture, where we incorporate additional 3D de-convolution layers for each scale of skip connections, as well as add a cross-scale feature fusion that features of each level are concatenated to the lower level. We use the NeuroZoom architecture for both events (NeuroZoom-ev) and spikes (NeuroZoom-sp), which are constrained by the corresponding Mutli-E/-S dataset. To avoid blocking artifacts for the output of NeuroZoom-sp, we replace the 3D de-convolution layer of the third last to second last output layer of each Zoom-block with a bicubic upsampling layer.

EventZoom [9] introduces an event-to-image (E2I) conversion to leverage HR information from images. But reconstructing 4× image from noisy 1× image restored by E2VID [61] is challenging and would yield less benefit for the event SR. For NeuroZoom, we discard this module and introduce high-quality dataset and Zoom-block module to handle texture restoration of event SR. During training, we randomly select 40 multi-resolution event clips from the Mutli-E and generate 7500 $\{1\times, 2\times, 4\times\}$ LR-HR event pairs as the training set of NeuroZoom-ev, and select 20 multi-resolution spike clips from the Mutli-S and generated 24000 $\{1\times, 2\times\}$ LR-HR spike pairs as the training set of NeuroZoom-sp. We use a batch size of 8 and train for 100 epochs. The Adam optimizer is used with an initial learning rate of 0.001, decayed by a factor of 0.5 every 50 epochs. By ablation experiments, we choose MSE loss for NeuroZoom-ev, and Charbonnier loss [62] and TV loss [63] for NeuroZoom-sp with a weight of $[1, 0.005]$. Both networks are implemented using PyTorch 1.6, with a time cost of about 12 hours on an NVIDIA 2080 Ti GPU.

4 NDSR Evaluation

4.1 Results of NeuroZoom-ev

To evaluate the performance of event-based NeuroZoom, the experimental results are organized as follows: 1) NeuroZoom-ev are compared with state-of-the-art denoisers on our real-captured ten samples. 2) For event SR, NeuroZoom-ev are compared with EventZoom [9].

Denoising. NeuroZoom-ev are compared with two types of event denoisers. The first type includes three basic methods that have been embedded in the software of the Prophesee camera, *i.e.*, activity noise filter (ACT) [64], trail noise filter (TRA), and spatial-temporal-contrast noise filter (STC). The second type of denoiser includes two state-of-the-art methods, EV-gait [36] and EDnCNN [22]. To evaluate

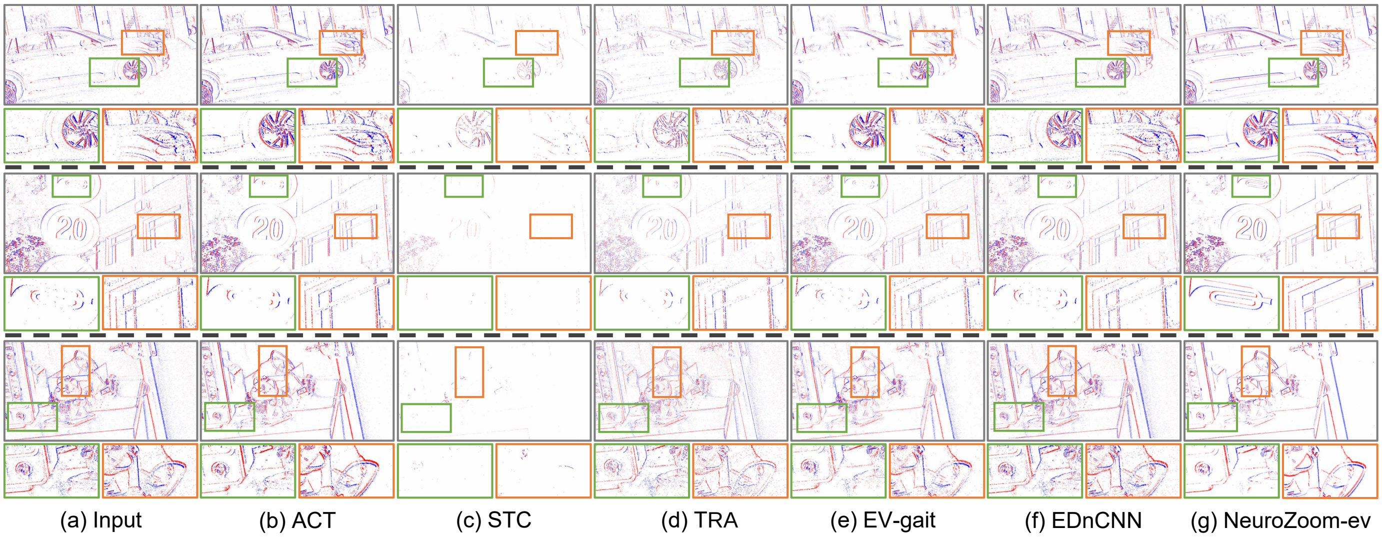

(a) Input (b) ACT (c) STC (d) TRA (e) EV-gait (f) EDnCNN (g) NeuroZoom-ev

Figure 6: Same-resolution denoising comparison results on the real-captured dataset. (a) Event frames clipped from four raw event streams which are captured by a Prophesee Gen 4.0 camera at a spatial resolution of 1280×720. (b)-(d) Denoising results of (a), processed by three basic noise filters provided by Prophesee [35]. (e)-(g) Denoising results of (a), processed by EV-gait [36], EDnCNN [22] and the proposed NeuroZoom-ev respectively. Closed-up views of green and orange boxes are shown below the results. Additional denoising results are included in the supplementary video.

Table 1: Denoising runtime comparison on real-captured data (unit: second)

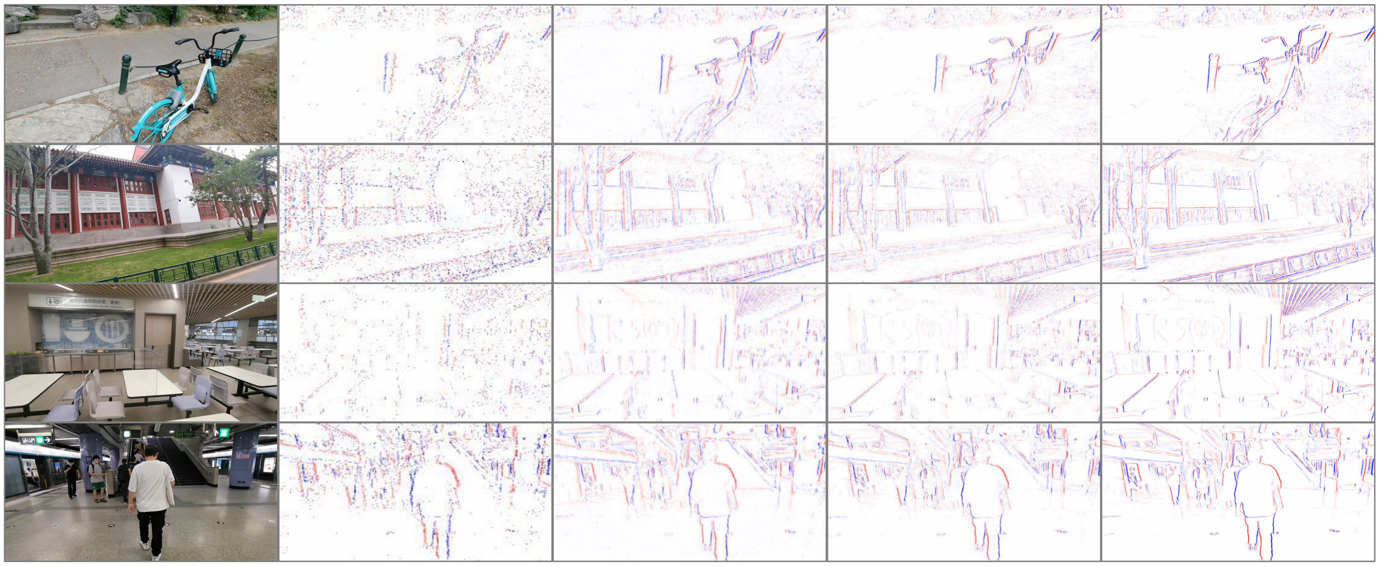
			bike	building	car_fast	car_slow	cat	leaf	person	sign	tower	**Average**
ACT [64]	Matlab	CPU	0.012	0.013	0.011	0.010	0.015	0.009	0.013	0.018	0.016	**0.013**
STC			0.009	0.010	0.011	0.008	0.011	0.007	0.012	0.013	0.012	**0.010**
TRA			0.009	0.010	0.011	0.008	0.011	0.007	0.012	0.013	0.012	**0.010**
EV-gait [36]			44.784	126.479	59.193	90.412	53.025	23.031	113.586	48.357	88.104	**71.886**
EDnCNN [22]			24.637	31.019	25.743	34.144	31.360	21.346	28.000	33.558	38.471	**29.809**
Ours	Pytorch		0.226	0.253	0.242	0.217	0.222	0.234	0.233	0.217	0.223	**0.230**
Ours		GPU	0.013	0.014	0.013	0.012	0.012	0.013	0.014	0.013	0.011	**0.013**

whether the NeuroZoom-ev enables to handle real event data, we collect ten real-captured event streams by a Prophesee event camera as the test dataset. The test scenarios cover common situations like indoor/outdoor settings, fast/slow motion and far/close scenes. In the denoising case, the NeuroZoom-ev $1\times$ are trained with same-resolution input-output pairs. The 3D de-convolution layers for skip connections shown in Fig. 5(b) are not used so that the output can keep the original size. The denoising results are shown in Fig. 6. As can be seen, NeuroZoom-ev is able to reveal and enhance the scene structures and effectively remove noisy events. ACT [64] and EV-gait [36] mainly focuses on removing background noises, STC pays attention to retain events where have high spatial-temporal-contrast, and TRA majors on eliminating trails of events. The effects of EDnCNN [22] seem not obvious because it is designed for DAVIS346 [6] and the network is trained on small resolution data labeled with the corresponding APS images. The qualitative results also show that NeuroZoom-ev hallucinate excessive events beyond the raw data, our method fills the missing information on edges, such as the car in the 2nd row and the trumpet sign in the 3rd row.

The runtime of all the denoisers is benchmarked in Table 1. We record the coding language, processor type, and runtime of each denoiser for processing an $11ms$-duration event stream per sample. Both EV-gait and EDnCNN require long run time compared to others. Ours method runs slowly than three basic denoisers on a CPU, but achieves the same speed when running on a GPU.

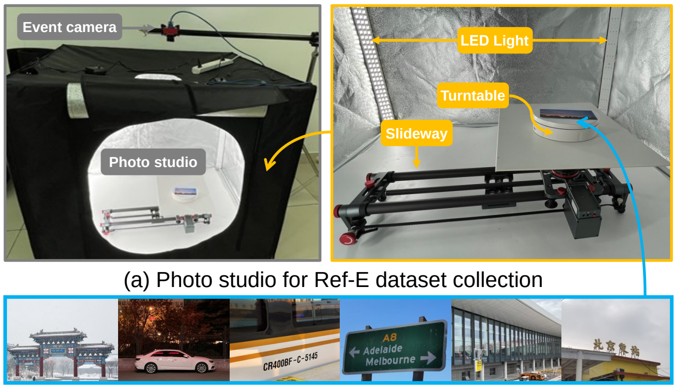
In summary, NeuroZoom-ev provides a fast event enhancement solution for event denoising and edge completion. We will further quantitatively compare the performance of all denoisers by a controlled experiment in Sec. 4.2.

Super resolution. We compare NeuroZoom-ev with EventZoom [9] for $2\times$ and $4\times$ event SR. Unlike EventZoom that uses image information as the auxiliary (E2I module) and performing EventZoom-$2\times$ twice to achieve $4\times$ SR, NeuroZoom-ev achieve $4\times$ SR directly, without help from images. It benefits from the improved network with a pyramid architecture and cross-scale feature fusion, and the upgraded Multi-E dataset in which the resolution of $1\times$-scale data is large enough to train the $4\times$ model effectively. We use the test data splitted from our Multi-E dataset. The SR results are shown in Fig. 7. Compared with EventZoom, the proposed NeuroZoom-ev reconstructs sharper and cleaner edges in $2\times$ event SR. For $4\times$ SR, our



(a) APS (b) Input (c) EventZoom (2x) (d) NeuroZoom-ev (2x) (e) NeuroZoom-ev (4x)

Figure 7: Comparison of $2\times$ and $4\times$ event SR performance between EventZoom [9] and our method.



(a) Photo studio for Ref-E dataset collection

(b) High-quality image sources

Figure 8: The experiment setting of the full-reference evaluation dataset collecting.

method can recover edges and textures even without any LR priors in the input event frames. This is due to the relatively longer time window it process. We will further verify the effectiveness through downstream applications in Sec. 5.

4.2 Full-reference evaluation for NeuroZoom-ev

Over the past decades, we have experienced the success of full-reference image quality assessments (*e.g.*, PSNR, FSIM [65], 2stepQA [66]) for objectively evaluating and benchmarking image processing methods. However, there are no such full-reference assessments for NDSR yet. EventZoom [9] is only evaluated by qualitative tests and downstream applications. It motivates us to quantitatively evaluate the performance of event enhancement methods with a full-reference fashion. To this end, we explore collecting a test dataset that includes raw event samples and their corresponding ground truth, then all methods process raw event samples and their performance are benchmarked by calculating the errors between the outputs and ground truth.

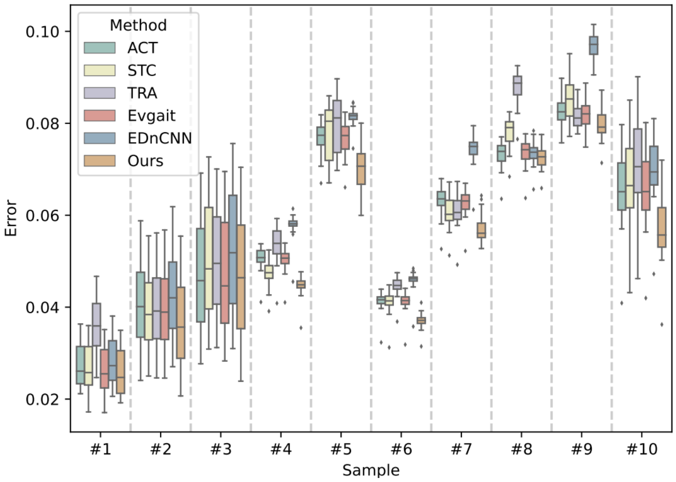


Figure 9: A boxplot to show the comparison results of the denoising performance in full reference evaluation. We compare existing denoisers with our proposed NeuroZoom-ev on our collected full reference test dataset Ref-E.

However, it is challenging to obtain the ground truth of raw events because existing DVS prototypes suffer from severe noise. An intuitive approach is to use an event simulator to simulate noisy and texture-missing events and corresponding noise-free and no missing texture events. Nevertheless, the published event simulators [42], [43] generate the corresponding event stream based on the interpolation of input videos, where the input videos lack microsecond-level temporal information, and the errors introduced by frame interpolation lead to an obvious gap between real and simulation. To deal with these problems, we design a controlled experiment where the intensity frame at the starting time and the motion speed at any time are known, which means that the intensity frame at any time is available, so that corresponding events can be calculated by the event formation model Eq. (6).

Reference events generation. We select uniform linear, uniform circular, and uniform circular linear motion as three basic motions, which are easy to realize in the controlled experiment. For uniform linear motion, suppose one pixel $\mathrm{x}_0 = [x_0, y_0]^\top$ moves at linear velocity $\mathbf{v} = [v_x, v_y]^\top$, the position transformation matrix of at any time t is given by:

$$\mathbf{T}(\mathbf{v}, t) \doteq \begin{bmatrix} 1 & 0 & v_x t \\ 0 & 1 & v_y t \\ 0 & 0 & 1 \end{bmatrix}. \tag{9}$$

Similarly, for uniform circular motion, suppose the pixel x_0 rotates around the pixel $\mathrm{x}_c = [x_c, y_c]^\top$ with angular velocity ω, the position transformation matrix of at any time t can be given by:

$$\mathbf{R}(\mathrm{x}_c, \omega, t) \doteq \begin{bmatrix} 1 & 0 & x_c \\ 0 & 1 & y_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\omega t & -\sin\omega t & 0 \\ \sin\omega t & \cos\omega t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -x_c \\ 0 & 1 & -y_c \\ 0 & 0 & 1 \end{bmatrix}. \tag{10}$$

For uniform circular linear motion, the position transformation matrix is a fusion of two motions:

$$\mathbf{M}(\mathbf{v}, \mathrm{x}_c, \omega, t) = \mathbf{T}(\mathbf{v}, t)\mathbf{R}(\mathrm{x}_c, \omega, t). \tag{11}$$

For an event stream sample S_i, we use the starting point intensity frame I_{s_i,t_0} to calculate the intensity frame of any time point t using Eq. (8-10), and synchronously simulate the event stream $E^{\mathrm{ref}}_{\mathrm{s}_i}$ according to Eq. (6). The spatial resolution of $E^{\mathrm{ref}}_{\mathrm{s}_i}$ is available by downsampling I_{s_i,t_0}.

Settings of the controlled experiment. We design a controlled experiment to collect a full-reference event test dataset, named Ref-E. Figure 8(a) shows our experiment setting. We use a photo studio to create a lighting environment without external interference, by evenly placing four LED light strips on the inner wall of the photo studio to approximate uniform illumination. In addition, we use a linear slideway and a turntable to combine three basic motion types. The camera is fixed at the top of the studio with its field of view plane parallel to the bottom plane of the studio. Here we choose a machine vision camera (Point Grey Chameleon3) and an event camera (DAVIS346 mono) to capture samples. As the examples shown in Fig. 8(b), we choose high-quality images as simulated shooting scenes, which are printed at high DPI and glued on a white flatbed. These static scenes will become dynamic by placing them on the running slideway or turntable. Since the motion of the slideway in an entire stroke includes three parts: acceleration, uniform, and deceleration motion phases, we use a high frame-rate camera to shoot strokes of the slideway multiple times to determine the start and end points of the uniform motion in one stroke.

We collect ten samples by this setting. For each sample, the collection process is: i) Selecting one picture as the simulated scene; ii) Selecting one movement form; iii) Placing the picture on the chosen machine; iv) Moving the picture to the marked point (*e.g.*, the uniform motion starting points of the slideway). We shoot images with the machine vision camera centered above the studio, and then take ten images in succession and average them to avoid image noise. v) Replacing the machine vision camera with the event camera, and use a gradienter and reference points to keep the field of view of the two cameras overlapping. vi) Turning on the event camera shooting mode, then start motors, the event camera records the entire movement of the simulated scene. vii) Turning off the camera and motors, and clip events within a uniform motion stroke. To the end, we integrate

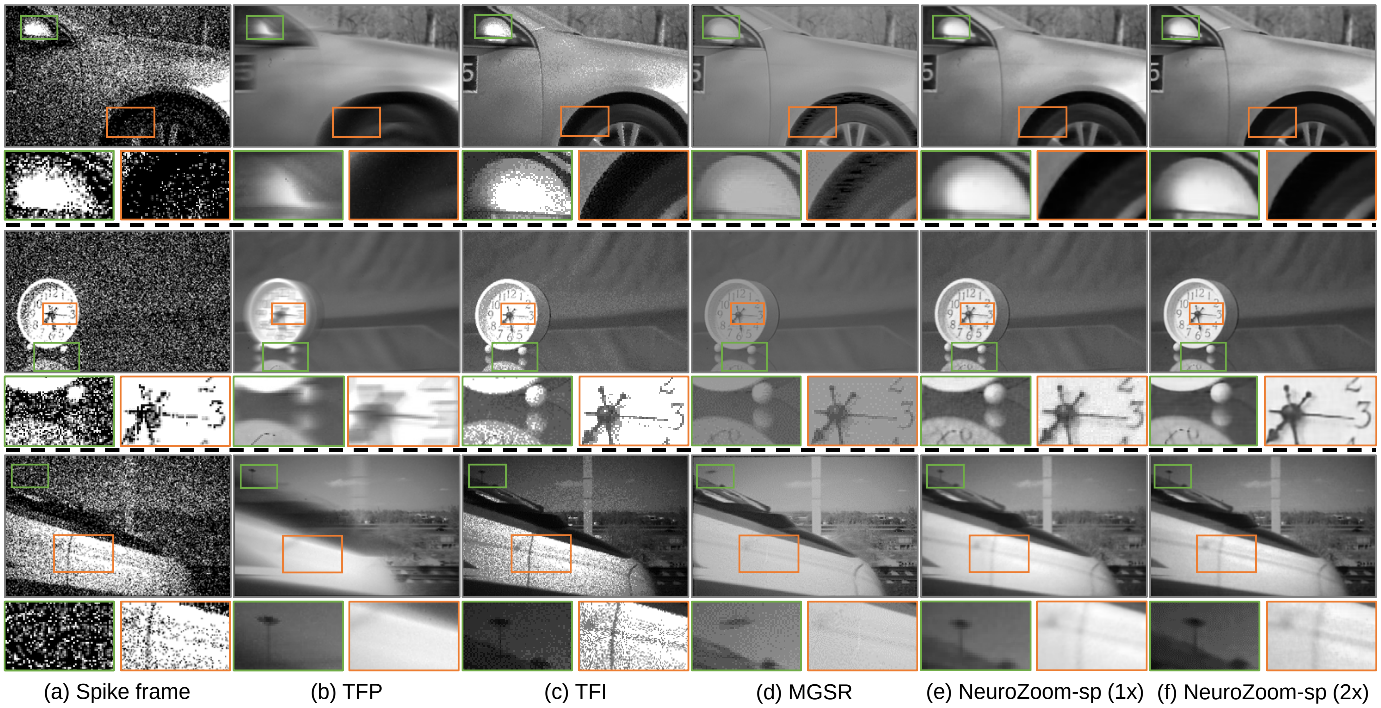


Figure 10: Spike denoising and SR (spike-based NDSR) comparison results. Closed-up views of green and blue boxes are shown below the results. Additional results are included in the supplementary video.

Table 2: Quantitative comparison results based on NIQE [67] and 2D-entropy [68] metrics. Red/blue: 1st/2nd best values.

Metrics	Methods	Class A			Class B					Average
		doll	car	train	railway	clock	keyboard	fruits	v-b	
NIQE(↓)	**TFP [28]**	8.20	7.64	10.62	6.57	7.84	8.61	9.00	8.15	**8.32**
	TFI [28]	7.96	13.02	6.49	8.14	22.60	14.73	24.05	10.18	**13.42**
	TVS [20]	7.48	9.31	6.78	7.01	13.42	11.36	12.34	8.23	**9.50**
	MGSR [31]	4.51	4.38	3.14	4.80	6.28	6.14	6.22	6.51	**5.25**
	Ours (1x)	4.77	4.79	3.73	6.44	4.29	4.26	3.85	5.35	**4.69**
	Ours (2x)	4.60	4.11	4.21	4.65	4.45	4.82	4.45	4.13	**4.43**
2D-entropy(↑)	**TFP [28]**	4.97	6.04	6.16	5.04	4.49	4.05	4.76	5.54	**5.13**
	TFI [28]	3.10	3.70	2.85	3.44	1.90	2.85	2.22	2.78	**2.86**
	MGSR [31]	5.88	6.53	6.53	6.11	5.10	5.17	5.10	6.12	**5.82**
	Ours (1x)	6.99	7.65	7.68	7.44	6.02	6.51	6.56	7.20	**7.01**
	Ours (2x)	6.88	7.56	7.69	7.52	5.96	6.41	6.62	7.18	**6.98**

the collected raw event $E_{s_i}^{\text{raw}}$ and the reference events $E_{s_i}^{\text{ref}}$ as the i-th sample.

Results. We input $E_{s_i}^{\text{raw}}$ into denoisers, and compare their outputs with $E_{s_i}^{\text{GT}}$ to evaluate the performance of each denoiser. We choose to stack the output and "ground truth" to form a sequence of event frames at 90*FPS*, and then indicate the difference between the two by computing the frame-to-frame RMSE. We have tried direct event-level comparisons without stacking preprocessing, but the comparisons are meaningless due to the large quantitative difference between them. By calculating RMSE values, we benchmark each denoisers as Fig. 9 shows. In this boxplot figure, for each box, the median line (horizontal line in box) denotes the average error of the event frame sequence between outputs and $E_{s_i}^{\text{GT}}$, the span between lower and upper quartiles (length of the box) denotes the central interval of the error distribution. Therefore, the lower the median line, the better the denoising performance of the denoiser. As can be seen, the proposed NeuroZoom-ev gets smaller denoising errors than other methods in this full-reference evaluation.

4.3 Results of NeuroZoom-sp

We compare the image reconstruction performance of the proposed NeuroZoom-sp, the current main application of spike cameras, with existing methods, *i.e.*, Texture From Playback (TFP) [28], Texture From Interval (TFI) [28], Texture via Spiking neural model (TVS) [20], and MGSR [31]. We choose the real captured spike dataset released by [31] as the test dataset. As the authors described, this dataset is captured with 20000/40000*Hz* and at a resolution of 400 × 250. All test samples record high speed scenes (*e.g.*, sample *train* records a high speed train traveling at $350km/h$) and are divided into two classes based on motion types, where Class A corresponds to object motion and Class B corresponds to ego-motion. For qualitative comparison, we compare the image restoration quality for two Class A samples and two Class B samples, and the results of all methods are shown in Fig. 10. The results of MGSR [31] are provided by the authors, and others are from our implemented code. As can be seen, the results of TFP suffer from severe motion blur, TFI clearly restores image textures but introduces undesired noise, MGSR [31] significantly eliminates noise and motion blurry, but there are still obvious artifacts. In contrast, NeuroZoom-sp (1×) restores textures without introducing artifacts. With the increase of resolution, NeuroZoom-sp (2×) further restores the details, *e.g.*, the car mirror in the first sample and the clock hand in the third sample.

Our method also achieves a trade-off between restoration quality and runtime. For this test dataset, our implementation uses an NVIDIA 2080 Ti GPU, and the average runtime per spike frame of TFP is less than $0.5ms$, TFI is $2ms$, NeuroZoom-sp (1×) is $7ms$ and NeuroZoom-sp (2×) is $48ms$. MGSR [31] is implemented using Matlab on CPU, with an average runtime of about $2min$ per spike frame.

For quantitative comparison, we use two no-reference image quality assessments as the metrics (*i.e.*, naturalness image quality evaluator (NIQE) [67] and 2D-entropy [68]) to evaluate the spike-based NDSR quality for compared methods. As recorded in Table 2, NeuroZoom-sp achieves the best performance for both 1× and 2× SR.

5 Applications

Event cameras have shown promising capability in auxiliary dealing with high-level as well as low-level computer vision tasks, we show two applications of the proposed NeuroZoom-ev to benefit event-based visual object tracking Sec. 5.1 and SR image reconstruction Sec. 5.2.

5.1 Event-based visual object tracking

We evaluate the performance improvement that our NeuroZoom-ev can bring to the task of event-based visual object tracking. STNet [70] is chosen as the benchmark tracker, which is an efficient event-based tracking method that uses a spiking transformer architecture for single object tracking from event frames. We respectively input raw events, 2× and 4× enhanced by NeuroZoom-ev into STNet [70], and compare the object tracking accuracy to evaluate the effectiveness of NeuroZoom-ev. The event-based object tracking dataset VisEvent [69] allows us to benchmark each type of input. The test dataset of VisEvent consists of 172 samples, each of which includes a real-captured video, a synchronized event stream, and corresponding ground truth bounding boxes. The captured scenes cover diverse scenes such as low illumination, high

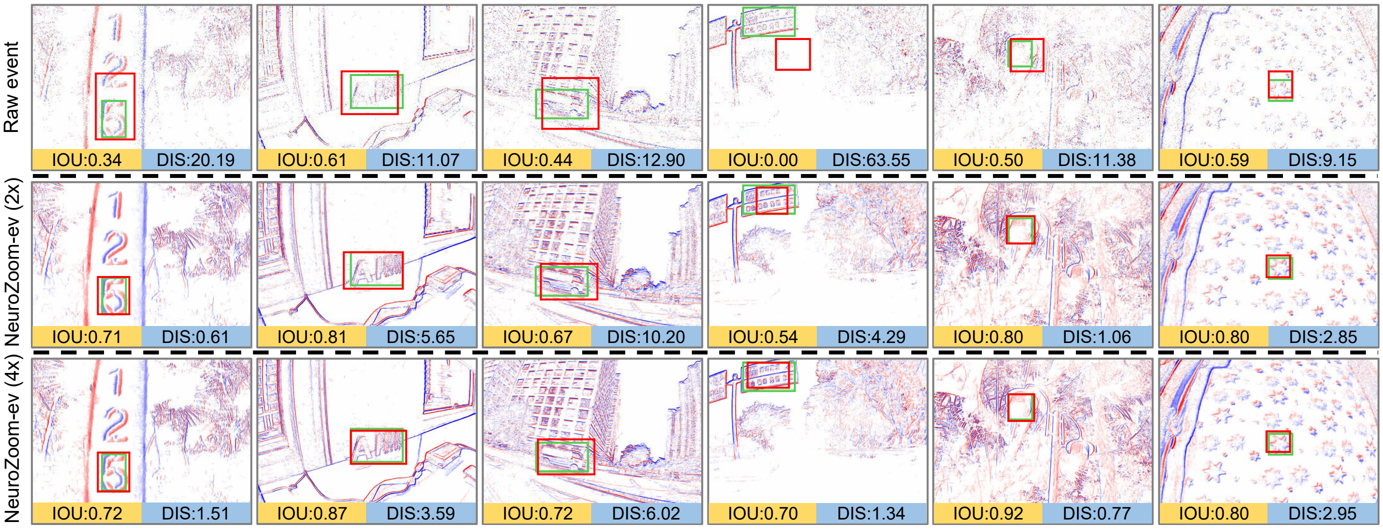


Figure 11: Examples for the tracking results. Red/Green bounding boxes represent the prediction/ground truth. Larger IoU and smaller DIS correspond to better performance. Additional results are included in the supplementary video.

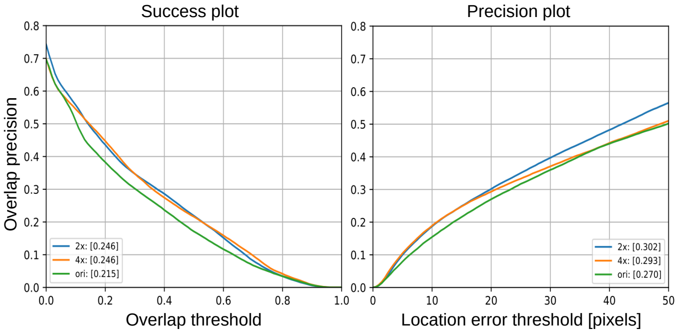


Figure 12: Success plot (left) and precision plot on the VisEvent [69] test datasets. We compare three situations: tracking from raw event data, and from processed event data enhanced by 2× and 4× NeuroZoom-ev.

speed, and background clutter. Similar to the preprocessing process described in STNet [70], we remove samples that miss events or bounding boxes misaligned with timestamps, and finally choose 82 samples as our test dataset.

Each sequence is tested on three resolution scales. The 1× scale represents a resolution of 346 × 260, the 2× represents 692 × 520, and the 4× is 1384 × 1040. We show six comparison examples with the tracking results in Fig. 11, and record intersection over union (IoU) and center point distance (DIS) values below each results. IoU and DIS are used as evaluation metrics between predicted bounding boxes and the ground truth, where IoU represents the overlap ratio of the two boxes and DIS records the distance between the center points of the two boxes. We normalize the result values to 1× scale to ensure fair comparisons. It can be seen that, for these examples, with the increased resolution through our method, the event frame becomes clearer and the predicted bounding box gets closer to coincide with the ground truth, which means that the object is tracked more accurately. Values of IoU and DIS also verify that our event enhancement method obtains tracking improvements.

In order to comprehensively show the performance of the proposed method on all 82 test samples, we record all tracking accuracy results and show the success plot and precision plot in Fig. 12. Curves of the success plot denote the proportion of samples with IoU greater than the x-coordinate value, and curves of the precision plot denote the proportion of samples whose DIS value is less than the x-coordinate value. It can be seen that event samples enhanced through NeuroZoom-ev outperform the raw event data on the object tracking task.

5.2 Event-based SR image reconstruction

We use NeuroZoom-ev for event-based SR image reconstruction. The E2VID [61] is chosen as the benchmark event-to-image reconstruction algorithm [18], [61]. For 2× SR image reconstruction, we compare with 1) 1× E2VID + image SR using DCLS [71], 2) E2SRI [41], one of the state-of-the-art algorithms that performs super resolved image reconstruction directly from raw 1× event data, 3) EventZoom [9], 4) our proposed NeuroZoom-ev (2×). For 3) and 4), we first feed raw 1× events into two methods respectively to restore SR event data, and then employ E2VID [61] to reconstruct 2× images. E2VID [18], EventZoom 2× [9], and E2SRI 2×/4× [41] are all retrained with 32 samples of the training data of Mutli-E dataset. The results are shown in Fig. 13. As can be seen in the figure, NeuroZoom-ev achieves the best image reconstruction quality at 2×. For 4× SR, we also compare with first 1× E2VID then 4× DCLS [71] and direct 4× E2SRI [41]. The 4× results in Fig. 13 show our strategy effectively recovers text and texture of images without introducing obvious artifacts, which with only 1× event frames as input. For quantitative analysis, we benchmark the reconstruction performance on 13 test samples of the Multi-E dataset and calculate several measures including LPIPS, MSE, SSIM, and PSNR between reconstructed images and corresponding APS frames in Table 3. Our results show that NeuroZoom-ev outperforms E2SRI [41] and DCLS [71] across all metrics on average.

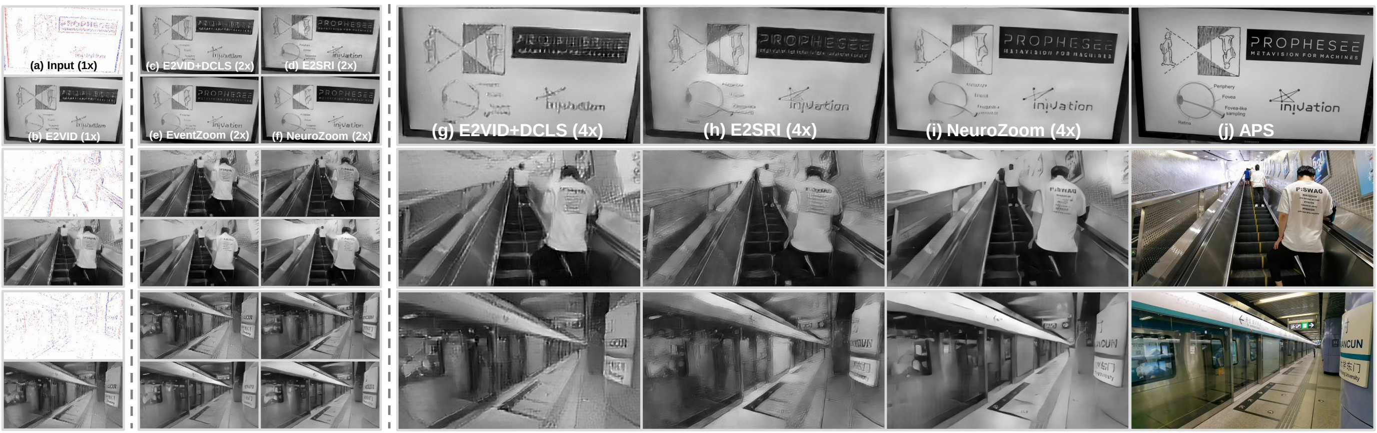


Figure 13: Comparison of event-based image reconstruction on our Multi-E dataset. The caption of each subfigure is labeled on the first sample and applies to all samples: (a) Input event frame. (b) Reconstruct 1× image with E2VID [61]. (c) Reconstruct 1× image with E2VID [61] and then 2× upsample image with DCLS [71]. (d) Reconstruct 2× image directly with E2SRI [41]. (e) Reconstruct 2× event with EventZoom [9] and then reconstruct 2× image with E2VID [61]. (f) 2× NeuroZoom-ev + E2VID [61]. (g) E2VID [61] + 4× DCLS [71]. (h) Reconstruct 4× image directly with E2SRI [41]. (i) 4× NeuroZoom-ev + E2VID [61]. (j) An APS frame.

Table 3: Image reconstruction performance.

	LPIPS (↓)	MSE (↓)	SSIM (↑)	PSNR (↑)
DCLS (2x) [71]	0.343	0.023	0.709	16.760
E2SRI (2x) [41]	0.254	0.020	0.746	18.333
EventZoom (2x) [9]	0.253	0.013	0.750	19.828
Ours (2x)	0.228	0.012	0.760	20.048
DCLS (4x) [71]	0.456	0.023	0.750	16.781
E2SRI (4x) [41]	0.335	0.025	0.747	16.938
Ours (4x)	0.320	0.016	0.792	18.380

6 Conclusion

This paper presents a novel neural framework to address NDSR for both event and spike camera, referred as NeuroZoom. NeuroZoom uses a 3D U-Net as the backbone architecture with a pyramid architecture and cross-scale feature fusion that enable to implement both 2× and 4× SR together. NeuroZoom-ev and NeuroZoom-sp share same architecture to perform event/spike denoising and super resolution. In order to learn the mapping from LR to HR neuromorphic data, we propose a display-camera system for multi-resolution event and spike data collection. The system is used to convert the high framerate RGB videos to an event version (Mutli-E) at four scales and a spike version (Mutli-S) at two scales. By training with the provided noise-corrupted HR-LR pairs, the network is able to effectively perform NDSR up to 4× SR. We also collect a full-reference test dataset for events by a controllable camera system, and benchmark the restoration performance for existing event-based NDSR methods. NeuroZoom achieves state-of-the-art results with improved time efficiency, and the enhanced events also contribute to improved visual task performance.

There are several limitations for this work. The dataset quality is compromised by the display, which has relatively low refresh rate and dynamic range. Interestingly, we do not find much generalization issue for the trained models after testing on external datasets, which means our method is compatible with real-captured event and spike data.

Acknowledgment

This work was supported by National Key R&D Program of China (2021ZD0109800) and National Natural Science Foundation of China under Grant No. 62088102, 62136001.

References

[1] G. Gallego, T. Delbrück, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza, "Event-based vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 154–180, 2022.

[2] T. Huang, Y. Zheng, Z. Yu, R. Chen, Y. Li, R. Xiong, L. Ma, J. Zhao, S. Dong, L. Zhu, J. Li, S. Jia, Y. Fu, B. Shi, S. Wu, and Y. Tian, "1000× faster camera and machine vision with ordinary devices," *Engineering*, 2022.

[3] S. Dong, T. Huang, and Y. Tian, "Spike camera and its coding methods," in *Proc. of Data Compression Conference*, 2017.

[4] C. D. Schuman, S. R. Kulkarni, M. Parsa, J. P. Mitchell, P. Date, and B. Kay, "Opportunities for neuromorphic computing algorithms and applications," *Nature Computational Science*, vol. 2, no. 1, pp. 10–19, 2022.

[5] 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.

[6] G. Taverni, D. P. Moeys, C. Li, C. Cavaco, V. Motsnyi, D. S. S. Bello, and T. Delbruck, "Front and back illuminated dynamic and active pixel vision sensors comparison," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 5, pp. 677–681, 2018.

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

[8] L. Wang, T.-K. Kim, and K.-J. Yoon, "EventSR: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[9] P. Duan, Z. Wang, X. Zhou, Y. Ma, and B. Shi, "EventZoom: Learning to denoise and super resolve neuromorphic events," in *Proc. of Computer Vision and Pattern Recognition*, 2021.

[10] S. Chen and M. Guo, "Live demonstration: Celex-v: a 1m pixel multi-mode event-based sensor," in *Proc. of Computer Vision and Pattern Recognition Workshops*, 2019.

[11] S. Dong, L. Zhu, D. Xu, Y. Tian, and T. Huang, "An efficient coding method for spike camera using inter-spike intervals," in *Proc. of Data Compression Conference*, 2019.

[12] S. Tulyakov, D. Gehrig, S. Georgoulis, J. Erbach, M. Gehrig, Y. Li, and D. Scaramuzza, "Time Lens: Event-based video frame interpolation," in *Proc. of Computer Vision and Pattern Recognition*, 2021.

[13] X. Zhou, P. Duan, Y. Ma, and B. Shi, "EvUnroll: Neuromorphic events based rolling shutter image correction," in *Proc. of Computer Vision and Pattern Recognition*, 2022.

[14] Y. Zheng, L. Zheng, Z. Yu, B. Shi, Y. Tian, and T. Huang, "High-speed image reconstruction through short-term plasticity for spiking cameras," in *Proc. of Computer Vision and Pattern Recognition*, 2021.

[15] J. Han, C. Zhou, P. Duan, Y. Tang, C. Xu, C. Xu, T. Huang, and B. Shi, "Neuromorphic camera guided high dynamic range imaging," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[16] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Ultimate SLAM? combining events, images, and IMU for robust visual SLAM in HDR and high-speed scenarios," *IEEE Robot. and Auto. Lett.*, vol. 3, no. 2, pp. 994–1001, 2018.

[17] U. M. Nunes and Y. Demiris, "Robust event-based vision model estimation by dispersion minimisation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 9561–9573, 2021.

[18] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "High speed and high dynamic range video with an event camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 1964–1980, 2021.

[19] P. Bardow, A. J. Davison, and S. Leutenegger, "Simultaneous optical flow and intensity estimation from an event camera," in *Proc. of Computer Vision and Pattern Recognition*, 2016.

[20] L. Zhu, S. Dong, J. Li, T. Huang, and Y. Tian, "Retina-like visual image reconstruction via spiking neural model," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[21] D. Gehrig and D. Scaramuzza, "Are high-resolution event cameras really needed?," *ArXiv*, vol. abs/2203.14672, 2022.

[22] R. Baldwin, M. Almatrafi, V. Asari, and K. Hirakawa, "Event probability mask (EPM) and event denoising convolutional neural network (EDnCNN) for neuromorphic cameras," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[23] Z. W. Wang, P. Duan, O. Cossairt, A. Katsaggelos, T. Huang, and B. Shi, "Joint filtering of intensity images and neuromorphic events for high-resolution noise-robust imaging," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[24] P. Duan, Z. Wang, B. Shi, O. Cossairt, T. Huang, and A. Katsaggelos, "Guided Event Filtering: Synergy between intensity images and neuromorphic events for high performance imaging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8261–8275, 2021.

[25] J. Barrios-Avilés, A. Rosado-Muñoz, L. D. Medus, M. Bataller-Mompeán, and J. F. Guerrero-Martínez, "Less data same information for event-based sensors: A bioinspired filtering and data reduction algorithm," *Sensors*, vol. 18, no. 12, p. 4122, 2018.

[26] A. Khodamoradi and R. Kastner, "O (n)-space spatiotemporal filter for reducing noise in neuromorphic vision sensors," *IEEE Trans. on Emerg. Topics in Comput.*, vol. 9, no. 1, pp. 15–23, 2018.

[27] H. Liu, C. Brandli, C. Li, S.-C. Liu, and T. Delbruck, "Design of a spatiotemporal correlation filter for event-based sensors," in *Proc. of IEEE Inter. Sym. on Circ. and Sys.*, 2015.

[28] L. Zhu, S. Dong, T. Huang, and Y. Tian, "A retina-inspired sampling method for visual texture reconstruction," in *Proc. of International Conference on Multimedia Expo*, 2019.

[29] J. Zhao, R. Xiong, H. Liu, J. Zhang, and T. Huang, "Spk2ImgNet: Learning to reconstruct dynamic scene from continuous spike stream," in *Proc. of Computer Vision and Pattern Recognition*, 2021.

[30] L. Zhu, J. Li, X. Wang, T. Huang, and Y. Tian, "NeuSpike-Net: High speed video reconstruction via bio-inspired neuromorphic cameras," in *Proc. of International Conference on Computer Vision*, 2021.

[31] J. Zhao, J. Xie, R. Xiong, J. Zhang, Z. Yu, and T. Huang, "Super resolve dynamic scene from continuous spike streams," in *Proc. of International Conference on Computer Vision*, 2021.

[32] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning image restoration without clean data," in *Proc. of International Conference on Machine Learning*, 2018.

[33] X. Zhang, Q. Chen, R. Ng, and V. Koltun, "Zoom to learn, learn to zoom," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[34] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: learning dense volumetric segmentation from sparse annotation," in *Proc. of Medical Image Computing and Computer-Assisted Intervention*, 2016.

[35] E. Perot, P. de Tournemire, D. Nitti, J. Masci, and A. Sironi, "Learning to detect objects with a 1 megapixel event camera," in *Advances in Neural Information Processing Systems*, 2020.

[36] Y. Wang, B. Du, Y. Shen, K. Wu, G. Zhao, J. Sun, and H. Wen, "EV-Gait: Event-based robust gait recognition using dynamic vision sensors," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[37] G. Gallego, H. Rebecq, and D. Scaramuzza, "A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation," in *Proc. of Computer Vision and Pattern Recognition*, 2018.

[38] T. Stoffregen, G. Gallego, T. Drummond, L. Kleeman, and D. Scaramuzza, "Event-based motion segmentation by motion compensation," in *Proc. of International Conference on Computer Vision*, 2019.

[39] D. Liu, A. Parra, and T.-J. Chin, "Globally optimal contrast maximisation for event-based motion estimation," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[40] X. Peng, Y. Wang, L. Gao, and L. Kneip, "Globally-optimal event camera motion estimation," in *Proc. of European Conference on Computer Vision*, 2020.

[41] S. M. Mostafavi I., J. Choi, and K.-J. Yoon, "Learning to super resolve intensity images from events," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[42] R. Henri, G. Daniel, and S. Davide, "ESIM: an open event camera simulator," in *Proc. of Conference on Robot Learning*, 2018.

[43] Y. Hu, S.-C. Liu, and T. Delbruck, "V2E: From video frames to realistic dvs events," in *Proc. of Computer Vision and Pattern Recognition Workshops*, 2021.

[44] D. Gehrig, M. Gehrig, J. Hidalgo-Carrió, and D. Scaramuzza, "Video to events: Bringing modern computer vision closer to event cameras," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[45] T. Stoffregen, C. Scheerlinck, D. Scaramuzza, T. Drummond, N. Barnes, L. Kleeman, and R. Mahony, "Reducing the sim-to-real gap for event cameras," in *Proc. of European Conference on Computer Vision*, 2020.

[46] A. Z. Zhu, D. Thakur, T. Özaslan, B. Pfrommer, V. Kumar, and K. Daniilidis, "The multivehicle stereo event camera dataset: An event camera dataset for 3D perception," *IEEE Robot. and Auto. Lett.*, vol. 3, no. 3, pp. 2032–2039, 2018.

[47] Y. Hu, H. Liu, M. Pfeiffer, and T. Delbruck, "DVS benchmark datasets for object tracking, action recognition, and object recognition," *Front. in neuro.*, vol. 10, p. 405, 2016.

[48] F. Paredes-Vallés, 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.*, vol. 42, no. 8, pp. 2051–2064, 2019.

[49] D. Neil, M. Pfeiffer, and S.-C. Liu, "Phased LSTM: Accelerating recurrent network training for long or event-based sequences," in *Advances in Neural Information Processing Systems*, 2016.

[50] M. Cannici, M. Ciccone, A. Romanoni, and M. Matteucci, "A differentiable recurrent surface for asynchronous event-based data," in *Proc. of European Conference on Computer Vision*, 2020.

[51] Y. Sekikawa, K. Hara, and H. Saito, "EventNet: Asynchronous recursive event processing," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[52] S. Tulyakov, F. Fleuret, M. Kiefel, P. Gehler, and M. Hirsch, "Learning an event sequence embedding for dense event-based deep stereo," in *Proc. of International Conference on Computer Vision*, 2019.

[53] Z. W. Wang, W. Jiang, K. He, B. Shi, A. Katsaggelos, and O. Cossairt, "Event-driven video frame synthesis," in *Proc. of Computer Vision and Pattern Recognition Workshops*, 2019.

[54] L. Wang, Y.-S. Ho, K.-J. Yoon, *et al.*, "Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[55] Z. Jiang, Y. Zhang, D. Zou, J. Ren, J. Lv, and Y. Liu, "Learning event-based motion deblurring," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[56] N. Messikommer, D. Gehrig, A. Loquercio, and D. Scaramuzza, "Event-based asynchronous sparse convolutional networks," in *Proc. of European Conference on Computer Vision*, 2020.

[57] D. Gehrig, A. Loquercio, K. G. Derpanis, and D. Scaramuzza, "End-to-end learning of representations for asynchronous event-based data," in *Proc. of International Conference on Computer Vision*, 2019.

[58] Y. Huang, J. Li, Y. Hu, X. Gao, and H. Huang, "Transitional learning: Exploring the transition states of degradation for blind super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 6495–6510, 2023.

[59] G. Hamed Kiani, F. Ashton, H. Chen, R. Deva, and L. Simon, "Need for speed: A benchmark for higher frame rate object tracking," in *Proc. of International Conference on Computer Vision*, 2017.

[60] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *Proc. of Computer Vision and Pattern Recognition*, 2017.

[61] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[62] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Fast and accurate image super-resolution with deep laplacian pyramid networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2599–2613, 2018.

[63] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proc. of Computer Vision and Pattern Recognition*, 2015.

[64] T. Delbruck, "Frame-free dynamic digital vision," in *Proc. of the International Symposium on Secure-Life Electronics, Advanced Electronics for Quality Life and Society*, 2008.

[65] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, 2011.

[66] X. Yu, C. G. Bampis, P. Gupta, and A. C. Bovik, "Predicting the quality of images compressed after distortion in two steps," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5757–5770, 2019.

[67] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a "completely blind" image quality analyzer," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 209–212, 2013.

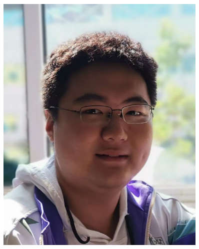
[68] A. S. Abutaleb, "Automatic thresholding of gray-level pictures using two-dimensional entropy," *Comput. Vis. Graph. Image Process.*, vol. 47, no. 1, pp. 22–32, 1989.

[69] X. Wang, J. Li, L. Zhu, Z. Zhang, Z. Chen, X. Li, Y. Wang, Y. Tian, and F. Wu, "VisEvent: Reliable object tracking via collaboration of frame and event flows," *ArXiv*, vol. abs/2108.05015, 2021.

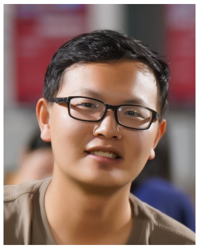
[70] J. Zhang, B. Dong, H. Zhang, J. Ding, F. Heide, B. Yin, and X. Yang, "Spiking transformers for event-based single object tracking," in *Proc. of Computer Vision and Pattern Recognition*, 2022.

[71] Z. Luo, H. Huang, L. Yu, Y. Li, H. Fan, and S. Liu, "Deep constrained least squares for blind image super-resolution," in *Proc. of Computer Vision and Pattern Recognition*, 2022.

Peiqi Duan is currently a Boya Postdoctoral Fellow at the School of Computer Science, Peking University. He received the PhD degree from Peking University in 2023. His research interests span event-based imaging and vision, single-image super-resolution, and HDR image reconstruction. He has served as a reviewer/program committee member for IJCV, TCSVT, CVPR, ICCV, ECCV, NeurIPS, etc.



Yi Ma received the B.S. degree from Peking University in 2021. He is currently a graduate student at Peking University. His research interests are centered around event-based vision signal processing.



Xinyu Zhou is a Ph.D. student in the School of Intelligence Science and Technology of Peking University. He received the B.S. degree from Peking University in 2022. His research interests include computational photography and event-based vision.

Xinyu Shi is a Ph.D. student in the School of Electronic Engineering and Computer Science of Peking University. His research interests include event-based vision signal processing and spiking neuron network.

Zihao Wang (publishes under Zihao W. Wang) received the Diploma degree in Optics with Honors from Chu Ko-chen Honors College in Zhejiang University in 2015, and the PhD degree from Northwestern University in 2020. His research interests include computer vision and machine learning. He has served as a reviewer/program committee member in CVPR, ECCV, NeurIPS, ICLR, AAAI, etc. He is a Computer Vision Engineer at Apple Inc.

Tiejun Huang received the B.Sc. and M.Sc. degrees in computer science from Wuhan University of Technology, China in 1992 and 1995, respectively, and the Ph.D. degree in pattern recognition and image analysis from Huazhong (Central China) University of Science and Technology in 1998. He is currently a professor with the School of Computer Science, Peking University, and the Director of the Beijing Academy for Artificial Intelligence. His research areas include visual information processing and neuromorphic computing. He is a Fellow of CAAI, CCF, CSIG and vice chair of the China National General Group on AI Standardization. He published 300+ peer-reviewed papers on leading journals and conferences, and also co-editor of 4 ISO/IEC standards, 5 National standards and 4 IEEE standards. He holds 100+ granted patents. Professor Huang received National Award for Science and Technology of China (Tier-2) for three times (2010, 2012, 2017).

Boxin Shi received the BE degree from the Beijing University of Posts and Telecommunications, the ME degree from Peking University, and the PhD degree from the University of Tokyo, in 2007, 2010, and 2013. He is currently a Boya Young Fellow Assistant Professor and Research Professor at Peking University, where he leads the Camera Intelligence Lab. Before joining PKU, he did research with MIT Media Lab, Singapore University of Technology and Design, Nanyang Technological University, National Institute of Advanced Industrial Science and Technology, from 2013 to 2017. His papers were awarded as Best Paper Runner-Up at ICCP 2015 and selected as Best Papers from ICCV 2015 for IJCV Special Issue. He has served as an associate editor of TPAMI/IJCV and an area chair of CVPR/ICCV. He is a senior member of IEEE.

Supplemental Material: NeuroZoom: Denoising and Super Resolving Neuromorphic Events and Spikes

Peiqi Duan, Yi Ma, Xinyu Zhou, Xinyu Shi, Zihao W. Wang,
Tiejun Huang, *Senior Member, IEEE*, Boxin Shi[‡], *Senior Member, IEEE*
[‡]Corresponding author.

7 Relationship between events and spikes

We use the real-captured spike-event calibrated data [1] to test whether spike-to-event transformation can be implemented in real data. Figure 14 shows the transformation results and the spike-transformed events tend to be the same as the real events. The corresponding videos are also added to the video of the supplementary material.

8 NeuroZoom-ev SR processing results on DAVIS240 dataset

To further illustrate the compatibility of the proposed algorithm for different series of event cameras, we show the processing results on the dataset [2] captured by a DAVIS240 event camera in Fig. 15. The corresponding videos are also added to the video of the supplementary material.

9 Ablation study on the event stream-dividing strategy

There are two event stream dividing strategies, *i.e.*, fixing the time interval of each channel and fixing the event number of each channel, we conduct an ablation study to find the appropriate dividing strategy. Event denoising is chosen as the basic task for this ablation. We experiment with different strategies and choose the optimal one based on denoising performance comparisons.

In the training stage, there are six cases: (1) The fixing-event-number strategy divides every 3000 events of LR data into one channel, with the corresponding HR events from the same period also stacked into that channel. Each channel is further divided into two sub-channels corresponding to event polarities. (2-6) For the fixing-time-duration strategy, each 3D tensor covers a fixed duration of $1/90s$, and we set five different channel numbers of 4, 8, 16, 32, and 64, respectively. Each channel further divides events at equal time intervals, which implies that the smaller the number of channels, the more temporal information is lost for events, and vice versa. We use the above six cases to train the network with an equal number of epochs.

In the test stage, we apply the fixing-time-duration strategy with the same 5 channel numbers as used in the

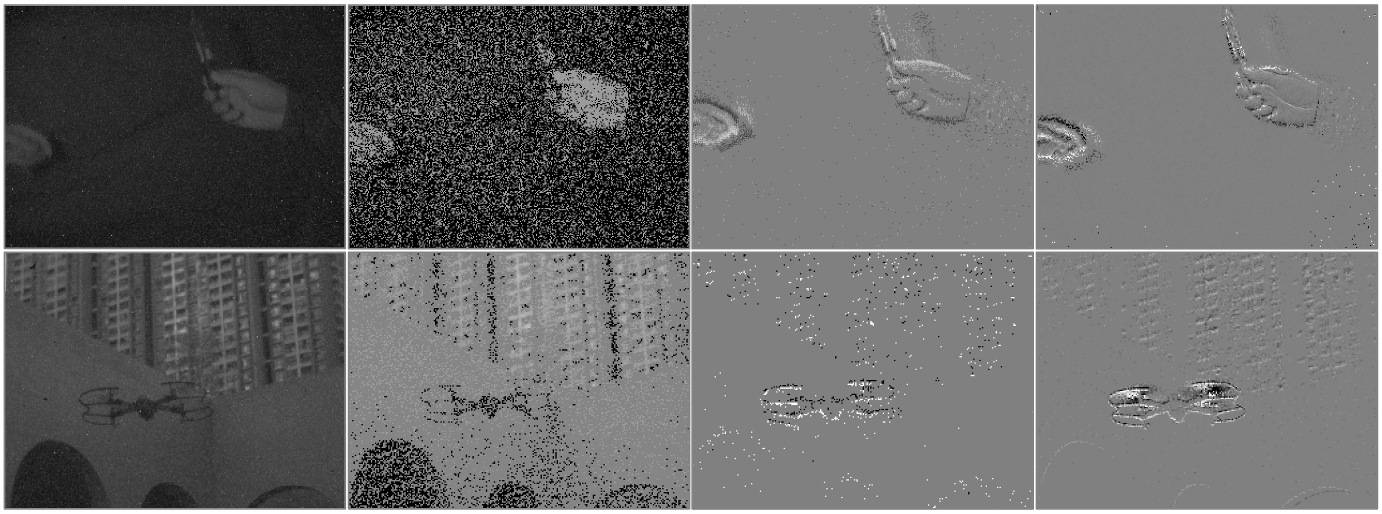

(a) Shooting scene (b) Real captured spike frame (c) Real captured event frame (d) Event frame transformed from spikes

Figure 14: Two examples to show the relationship between event and spike.

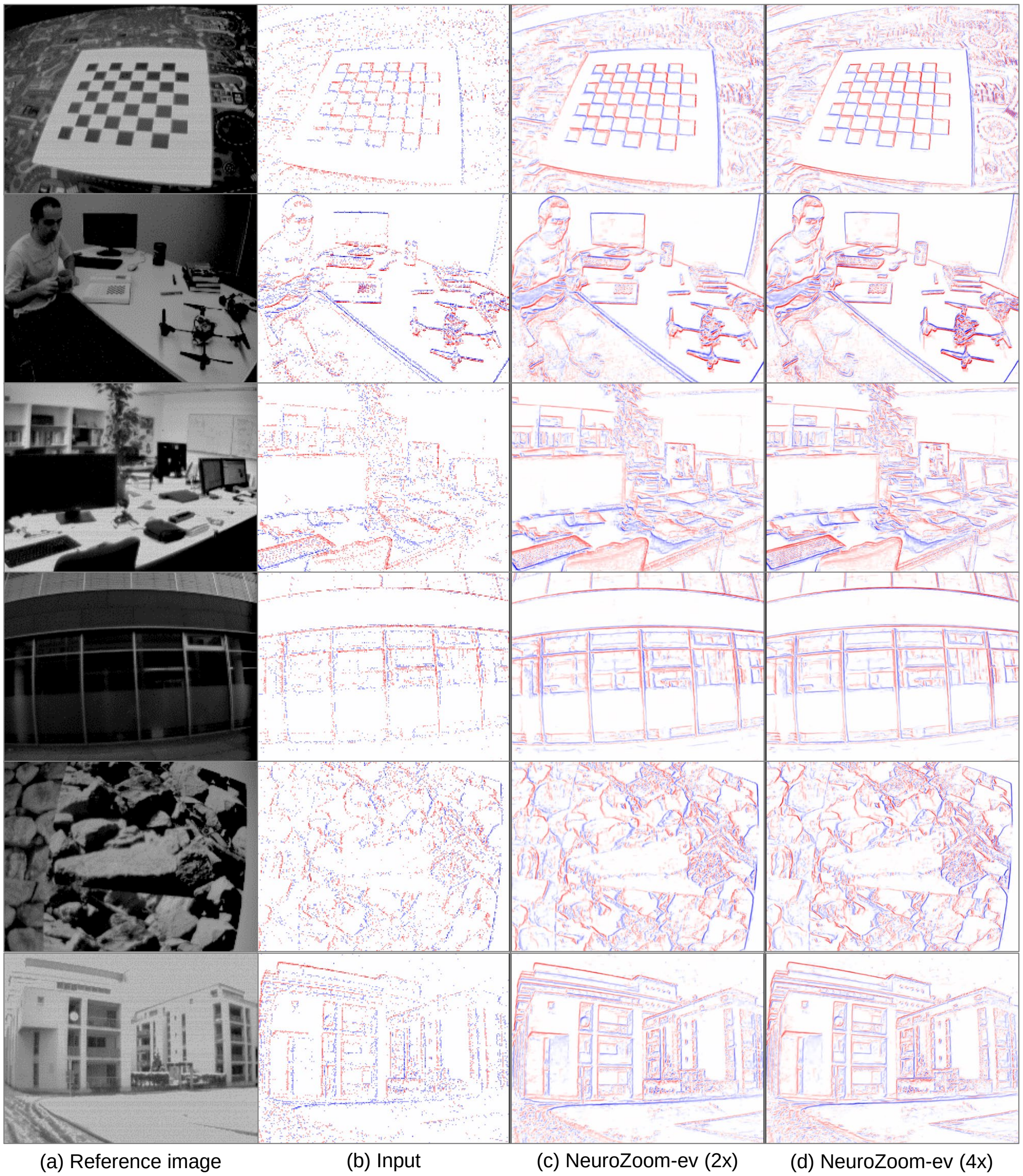


Figure 15: 2× and 4× event SR processing on DAVIS240 dataset [2].

training stage. We test different cases on our collected real event data which are captured by a Prophesee Gen 4.0. Several representative results are shown in Fig. 16 to Fig. 20. Specifically, Fig. 16 and Fig. 17 show two indoor scenes with slow and fast motion, respectively; while Fig. 18, Fig. 19, and Fig. 20 show the same outdoor scene at slow, moderate, and fast movement speeds. To ensure a fair comparison, all event frames in the figures cover the same time interval. Specifically, the event frames extracted from the result corresponding to a 4-channel case cover one channel, those from an 8-channel result are stacked from two channels, and so on.

From the results shown in Fig. 16 to Fig. 20, we observe that the choice of testing strategy has a more significant impact on the denoising performance compared to the training strategy. In the test stage, using a smaller number of channels may lead to blurred edges in the denoised event frame due to the loss of more temporal information, while a larger number of channels may not be able to effectively learn the denoising mapping because there are too few events allocated in each channel. In the training stage, the performance of the fixing-event-number strategy is more stable than others. Even if the extreme 4-channel and 64-channel cases are selected during the test, or in the face of various scene motion speeds, this strategy can reconstruct better results than other cases. Therefore, we suggest fixing the event number in the training stage, and $cn = 16$ in the test stage.

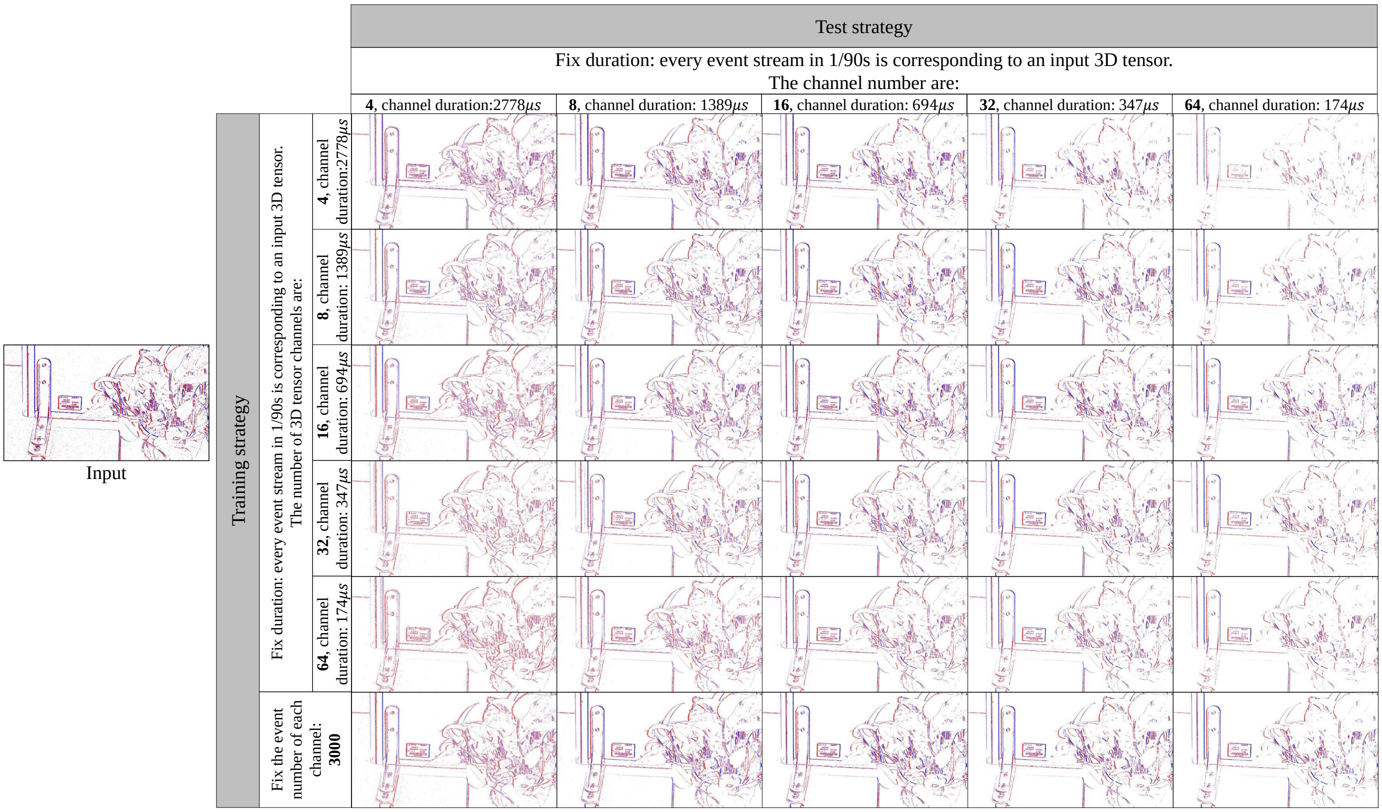

Test strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The channel number are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Input
Training strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor. The number of 3D tensor channels are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Fix the event number of each channel: 3000

Figure 16: Ablation study on different channel division strategies. (Indoor slow motion)

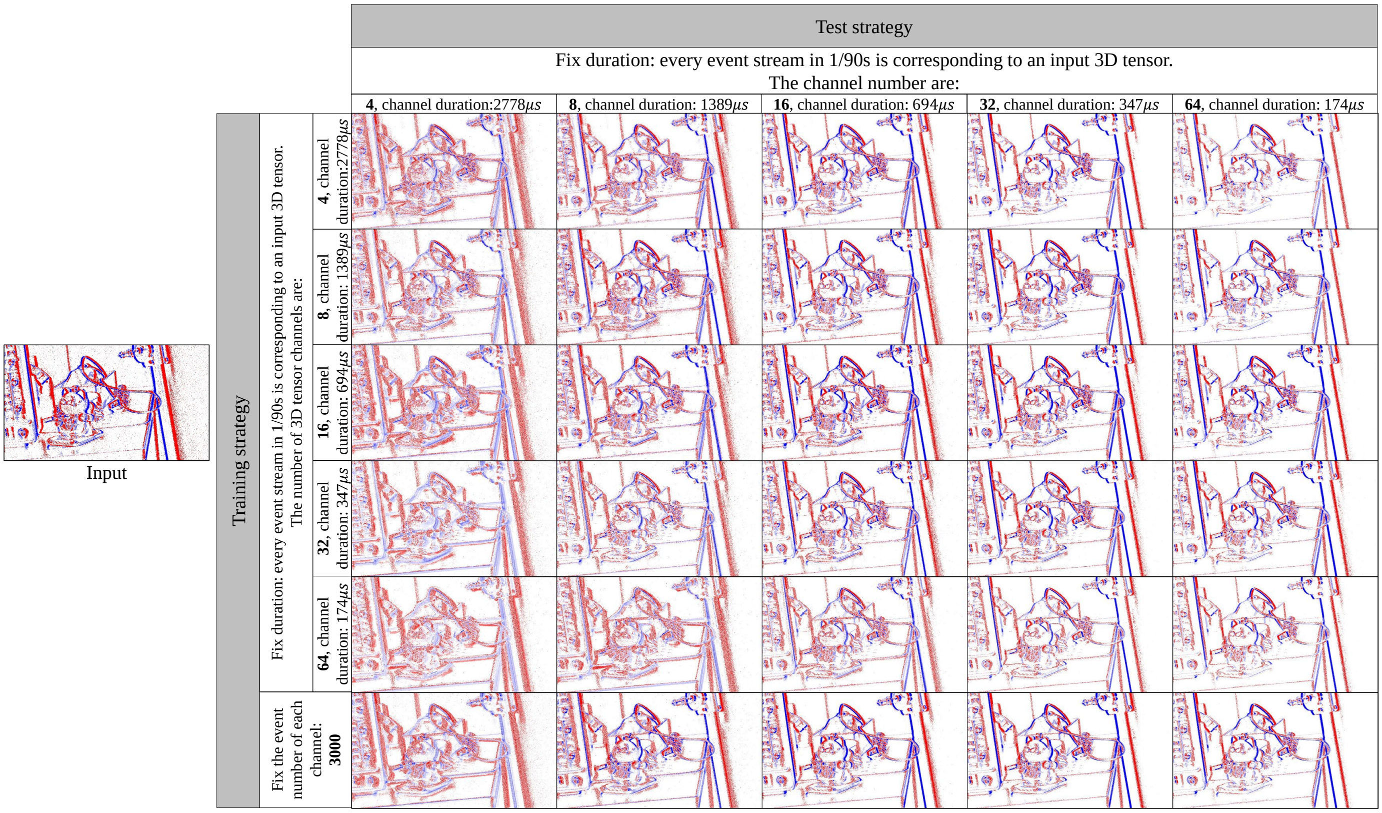

Test strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The channel number are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Input
Training strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor. The number of 3D tensor channels are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Fix the event number of each channel: 3000

Figure 17: Ablation study on different channel division strategies. (Indoor fast motion)

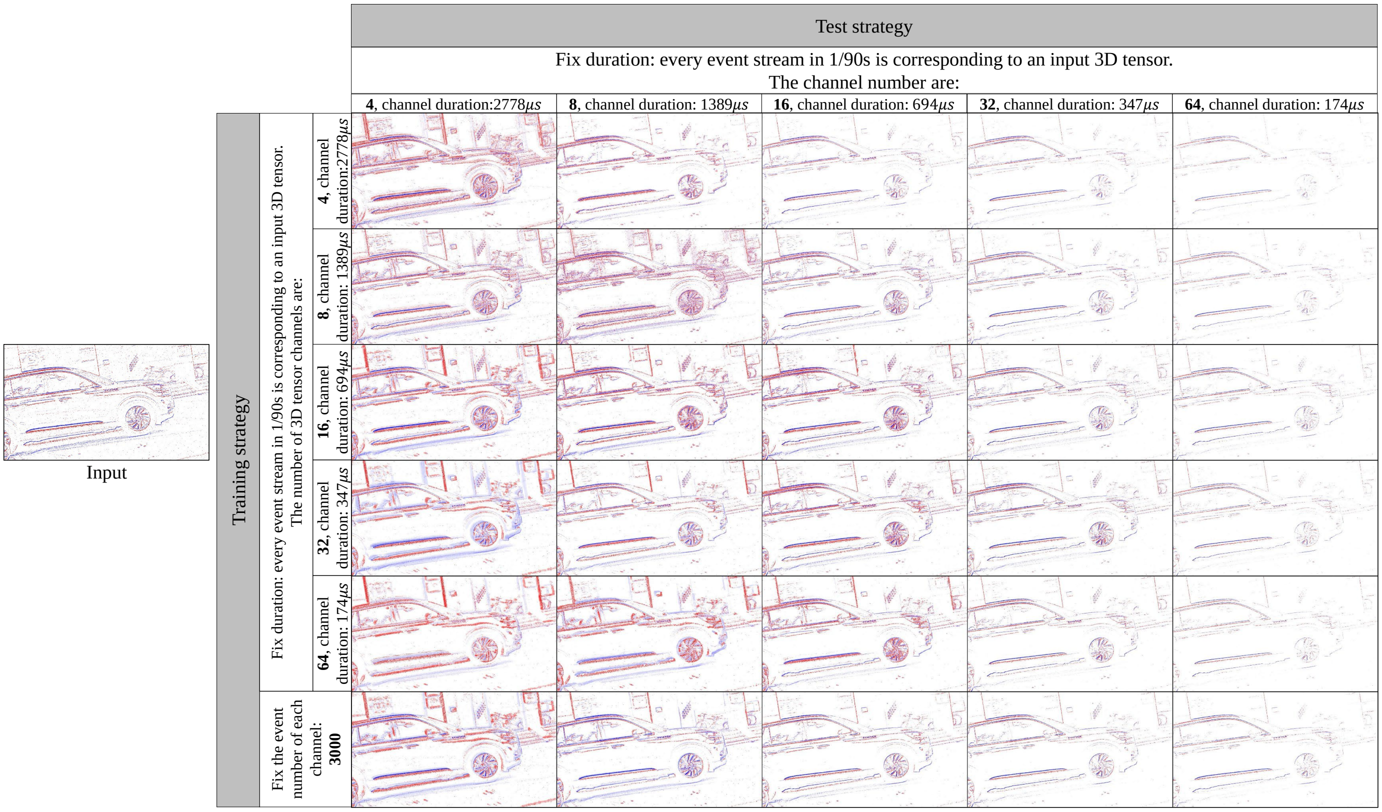

Test strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The channel number are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Training strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The number of 3D tensor channels are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Fix the event number of each channel: **3000**
Input

Figure 18: Ablation study on different channel division strategies. (Outdoor slow motion)

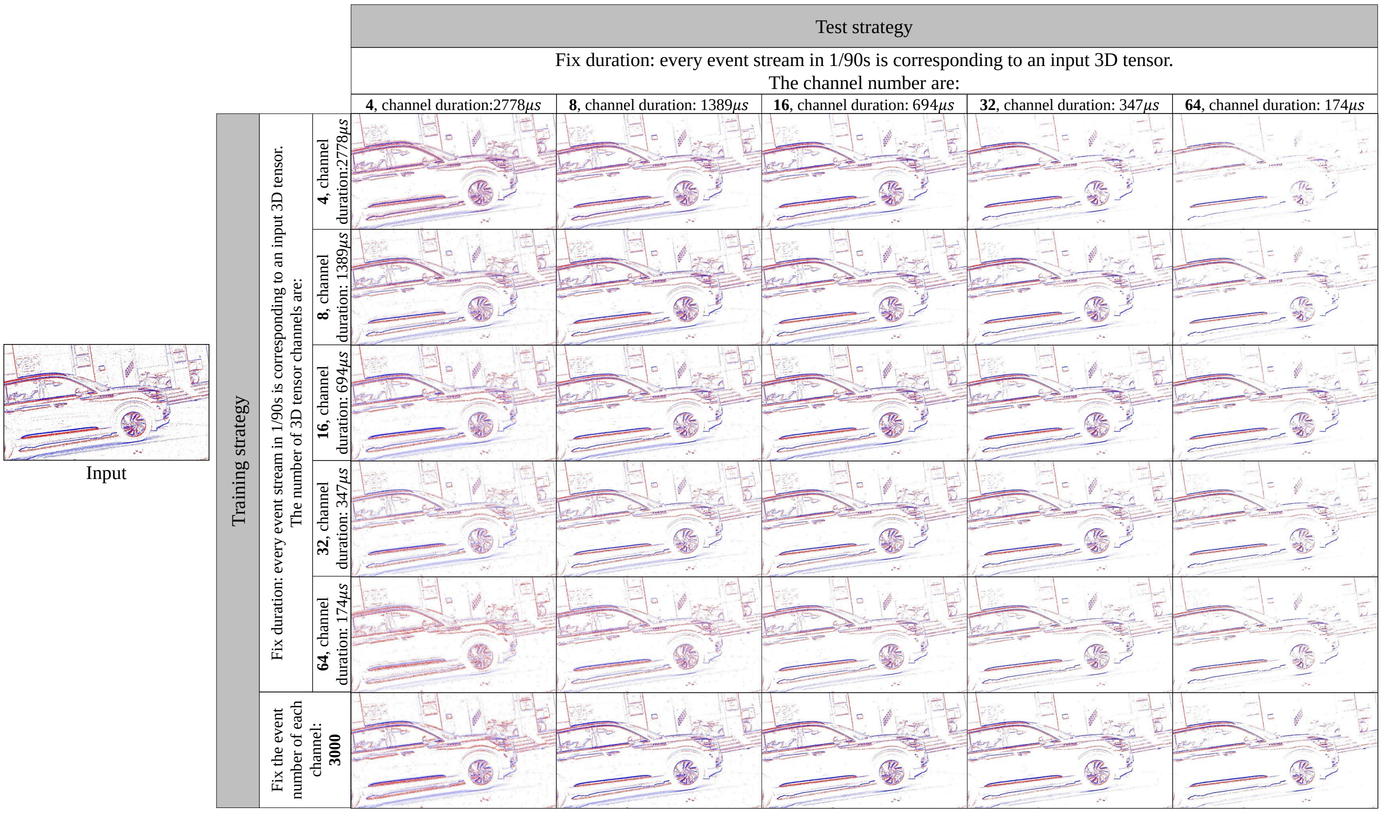

Test strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The channel number are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Training strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The number of 3D tensor channels are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Fix the event number of each channel: **3000**
Input

Figure 19: Ablation study on different channel division strategies. (Outdoor moderate motion)

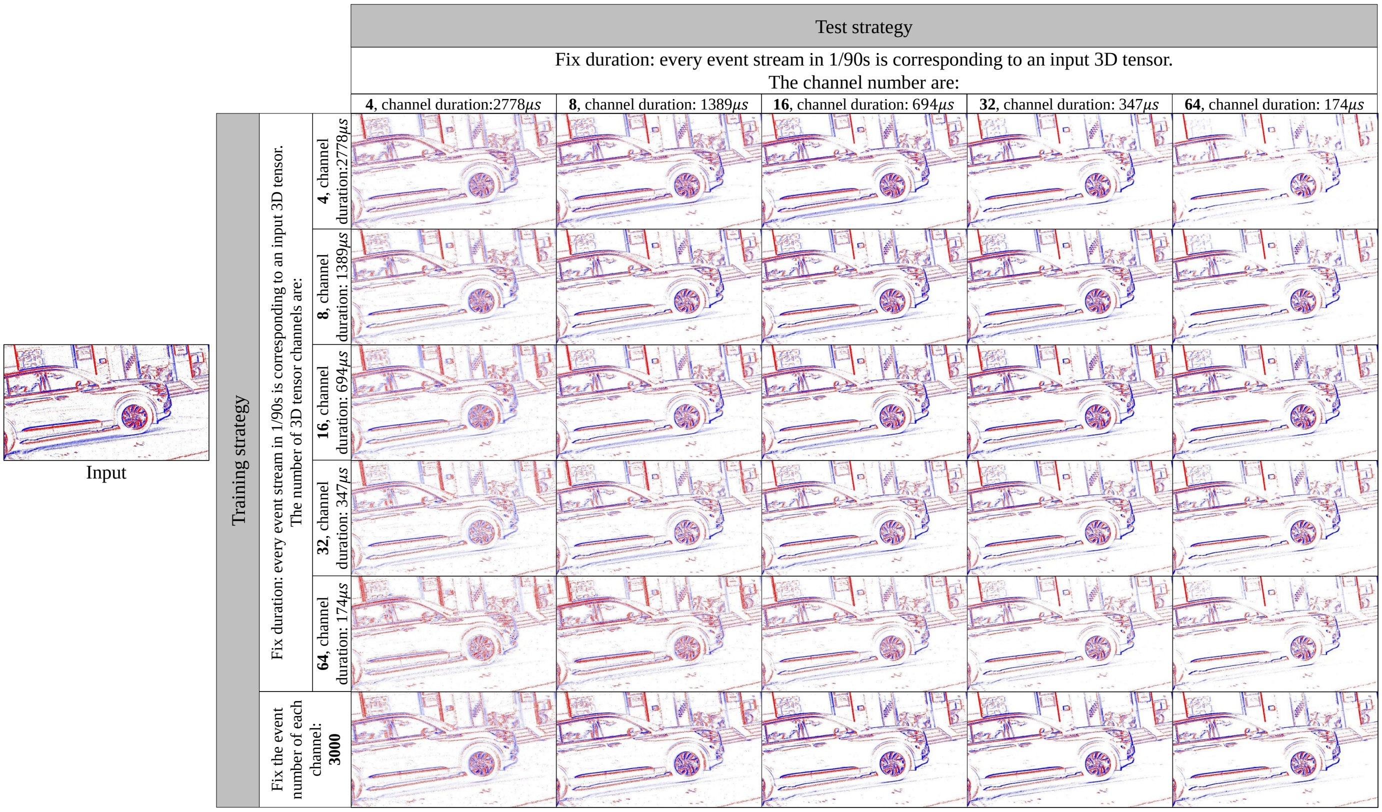

Test strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The channel number are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Training strategy
Fix duration: every event stream in 1/90s is corresponding to an input 3D tensor.
The number of 3D tensor channels are:
4, channel duration:2778μs
8, channel duration: 1389μs
16, channel duration: 694μs
32, channel duration: 347μs
64, channel duration: 174μs
Fix the event number of each channel: 3000
Input

Figure 20: Ablation study on different channel division strategies. (Outdoor fast motion)

10 Analysis of why NeuroZoom fills in missing information on edges

Event signals carry rich time information but limited spatial information (most areas are blank), besides, the bandwidth limitation of the event sensor (*i.e.*, bus congestion [3]) and the response delay of event pixels (*i.e.*, refractory period [4]) will lead to some events be discarded or ignored [4], [5]. Fortunately, due to the low-latency response characteristics of the event camera, even if some log brightness changes don't trigger any events at a certain moment or have been discarded before storage, these log brightness changes trigger events in their time neighborhood. We use this temporal information redundancy to recover discarded or ignored events. Figure 21 shows six adjacent event channels extracted from a single event 3D tensor and their corresponding results after NeuroZoom-ev processing. The raw event data are obtained from EDnCNN [6] and captured by a DAVIS346 camera. In the input line, an obvious event missing can be seen in channels 2, 3, and 5, but not in channels 1, 4, and 6, probably caused by the bus congestion or the refractory period of event sensors. The 3D U-Net [7] backbone of NeuroZoom-ev enables the network to extract not only spatial domain features but also temporal domain features, thereby exploiting temporal information redundancy to complement the missing edges of event frames. The output of NeuroZoom-ev shows that the event edges of the chessboard have been successfully reconstructed, with proper event polarity.

We conduct a test to show the performance when NeuroZoom-ev handles dense or specialized textures. As shown in Fig. 22 (a), we design two special pictures and paste them on the plane: Picture 1 (Fig. 22 top row) contains multiple disconnected line segments, with disconnections occurring gradually and clearly, and an elephant-shaped toy is pasted to occlude line segment texture; Picture 2 (Fig. 22 bottom row) is a specially designed checkboard, in which some black blocks are deleted (as red blocks marked). We use a DAVIS346 camera to capture these stationary pictures on the plane and shake the camera to trigger events. The collected events are shown in Fig. 22 (b). The SR processing results from NeuroZoom-ev are shown in Fig. 22 (c). As can be seen, the disconnected line segments in Picture 1 are properly connected and the original messy dense textures are restored to clear textures, thanks to the redundant information provided by the temporal neighbors. Similarly, the textures in Picture 2 are correctly restored, and there is no error to complement the area without black blocks. Note that there are no samples in our training dataset that contain a checkboard.

11 Ablation study of event re-distribution

To further validate the effectiveness of event re-distribution, we conduct an ablation study with different time precisions. Specifically, we simulate the event re-distribution process by first clipping the raw event stream and stacking each clip into one event frame. Then we randomly distribute the clips to the corresponding time periods. E2VID [8] is chosen as a downstream task to verify the effectiveness of event re-distribution. As shown in Fig. 23, we choose $[10, 30, 50, 70, 90, 110, 130, 150, 170]$ as the candidate clip

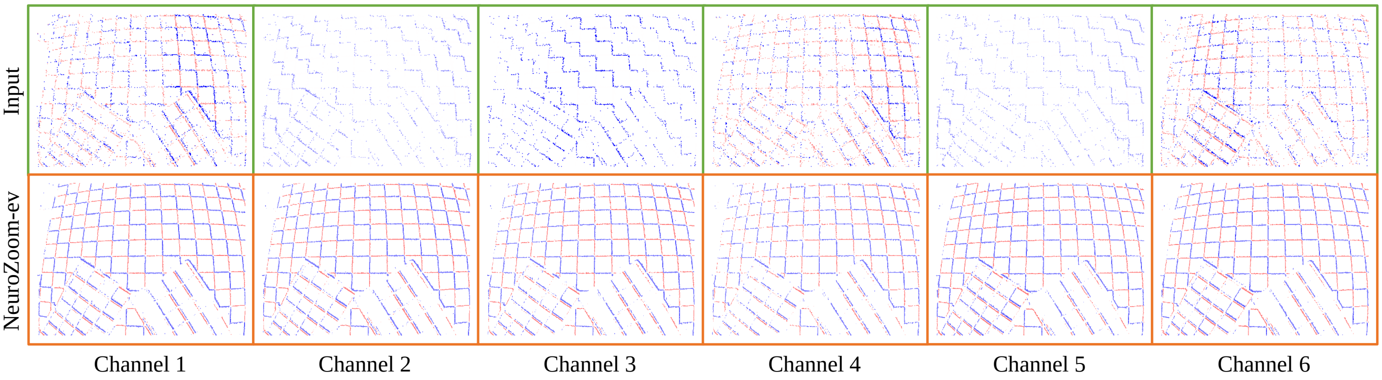
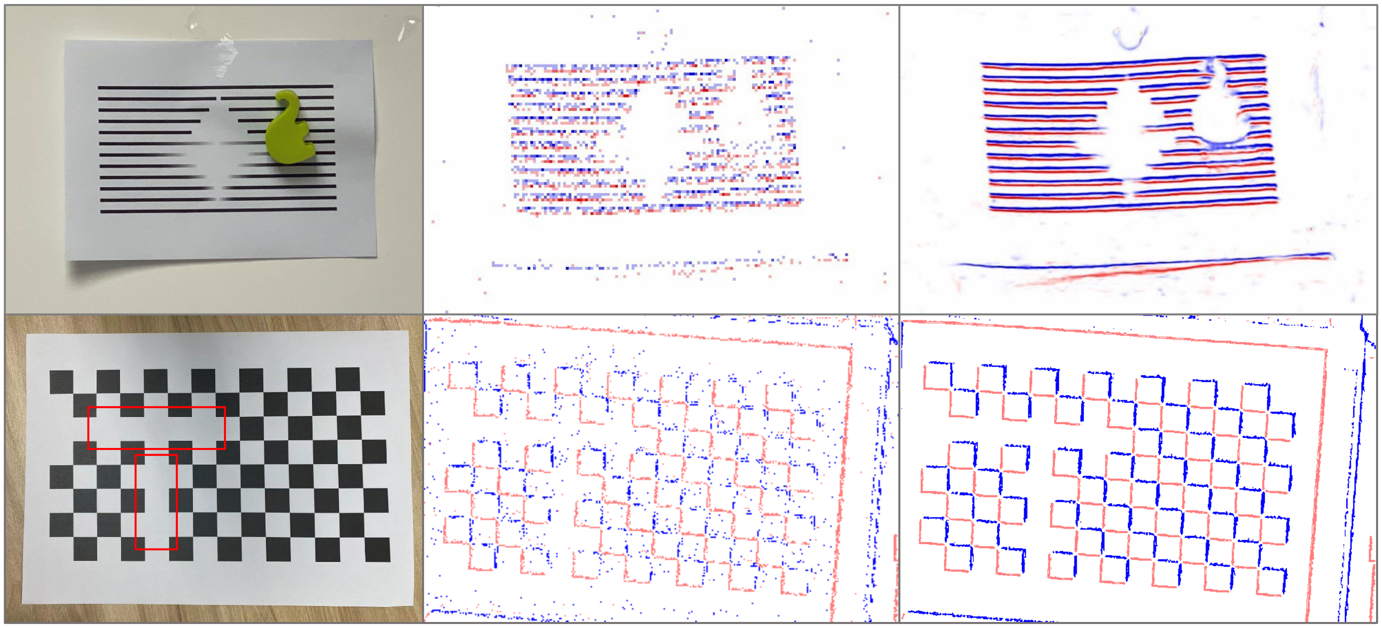


Figure 21: A set of raw event frames of adjacent channels in one event 3D tensor and the corresponding results after NeuroZoom-ev processing.



(a) Shooting scene (b) Raw events captured by a DAVIS 346 (c) SR results from NeuroZoom-ev

Figure 22: Examples of SR restoration results when NeuroZoom-ev handles dense or specialized textures.

rates for event streams and execute E2VID [8] to reconstruct videos at each rate. The $1\times$ and $2\times$ test datasets of our Mutli-E are used to evaluate the reconstruction quality, and five samples are shown in Fig. 23.

The qualitative comparison shows that as the clip rate increases, the quality of image reconstruction tends to be stable, and artifacts are significantly reduced. This result is consistent with our intuition, as the larger the clip rate, the less time information is lost, and the better the reconstruction result is. We also report the quantitative result in Table 4 and present the corresponding boxplot figures in Fig. 24. The comparison results on all metrics (LPIPS, MSE, SSIM, and PSNR) indicate that when the clip rate is larger than 30, the numerical results tend to be stable, and the difference is negligible, which means at least for this task, the event re-distribution with an appropriate time duration does not significantly lead to performance degradation.

12 Comparison of the SR results of direct SR strategy and doise+SR strategy.

In the Sec. 3.1, we formulate the basic image degradation model as follows:

$$\hat{I}^{\text{LR}} = (I^{\text{HR}} * k) \downarrow_s + n_{\text{image}}, \tag{11}$$

where the LR image $\hat{I}^{\text{LR}}$ is assumed the result of a downscaling operation from a degraded HR image I^{HR} added by noise. k denotes a degradation for isotropic Gaussian blur [9], $\downarrow_s$ is a downscaling operation with a scale factor of s, and n_{image} represents the additive image noise. Note that when we use this degradation to formulate events and spikes, the blur kernel k can be ignored due to the low latency requirement of neuromorphic signals. Hence, the degradation model is simplified to $\hat{I}^{\text{LR}} = (I^{\text{HR}}) \downarrow_s + n_{\text{image}}$. Let's first analyze the correlation between image denoising and SR. If we first denoise and then SR for $\hat{I}^{\text{LR}}$ by an SR function $F_{SR}(\cdot)$, and output an estimated HR image $\hat{I}^{\text{HR}}$, *i.e.*, $\hat{I}^{\text{HR}} = F_{SR}(\hat{I}^{\text{LR}} - n_{\text{image}})$, and let n_{loss} represents the

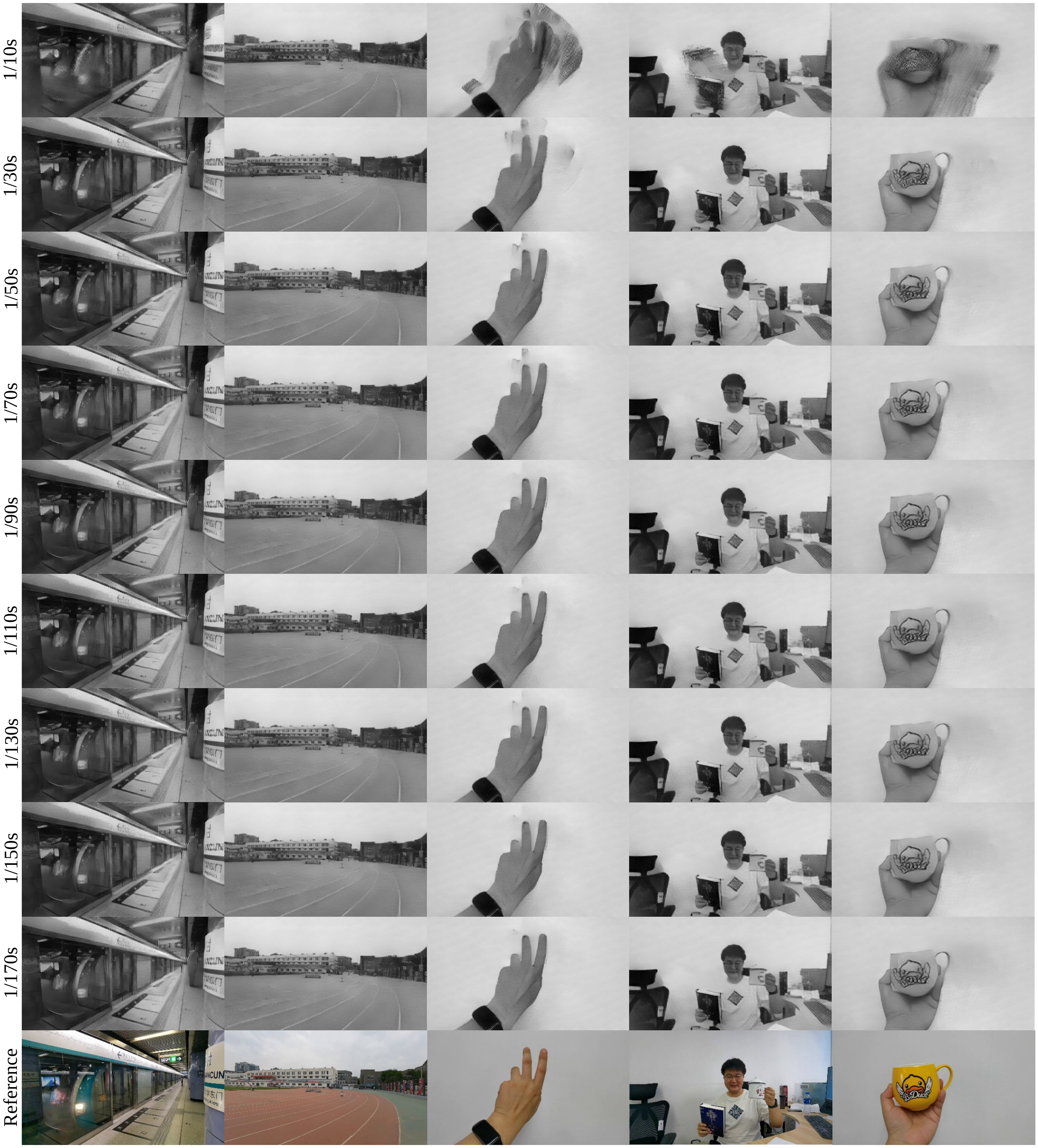


Figure 23: Event re-distribution effectiveness verification: Image reconstruction qualitative comparison for different clip rates of event streams.

Table 4: Event re-distribution effectiveness verification: Image reconstruction quantitative comparison for different clip rates of event streams.

		LPIPS (↓)	MSE (↓)	SSIM (↑)	PSNR (↑)			LPIPS (↓)	MSE (↓)	SSIM (↑)	PSNR (↑)
1×	10	0.2621	0.0221	0.6901	16.9302	2×	10	0.3320	0.0198	0.7428	17.2811
	30	0.2245	0.0231	0.6967	16.8177		30	0.2263	0.0198	0.7588	17.4683
	50	0.2233	0.0231	0.6964	16.8091		50	0.2189	0.0204	0.7581	17.4271
	70	0.2227	0.0232	0.6962	16.7999		70	0.2166	0.0203	0.7580	17.4415
	90	0.2217	0.0233	0.6960	16.7905		90	0.2126	0.0202	0.7579	17.4495
	110	0.2220	0.0233	0.6960	16.7900		110	0.2147	0.0205	0.7574	17.4120
	130	0.2220	0.0233	0.6960	16.7895		130	0.2138	0.0200	0.7583	17.4603
	150	0.2218	0.0233	0.6960	16.7877		150	0.2140	0.0204	0.7576	17.4259
	170	0.2218	0.0233	0.6960	16.7888		170	0.2137	0.0202	0.7579	17.4443

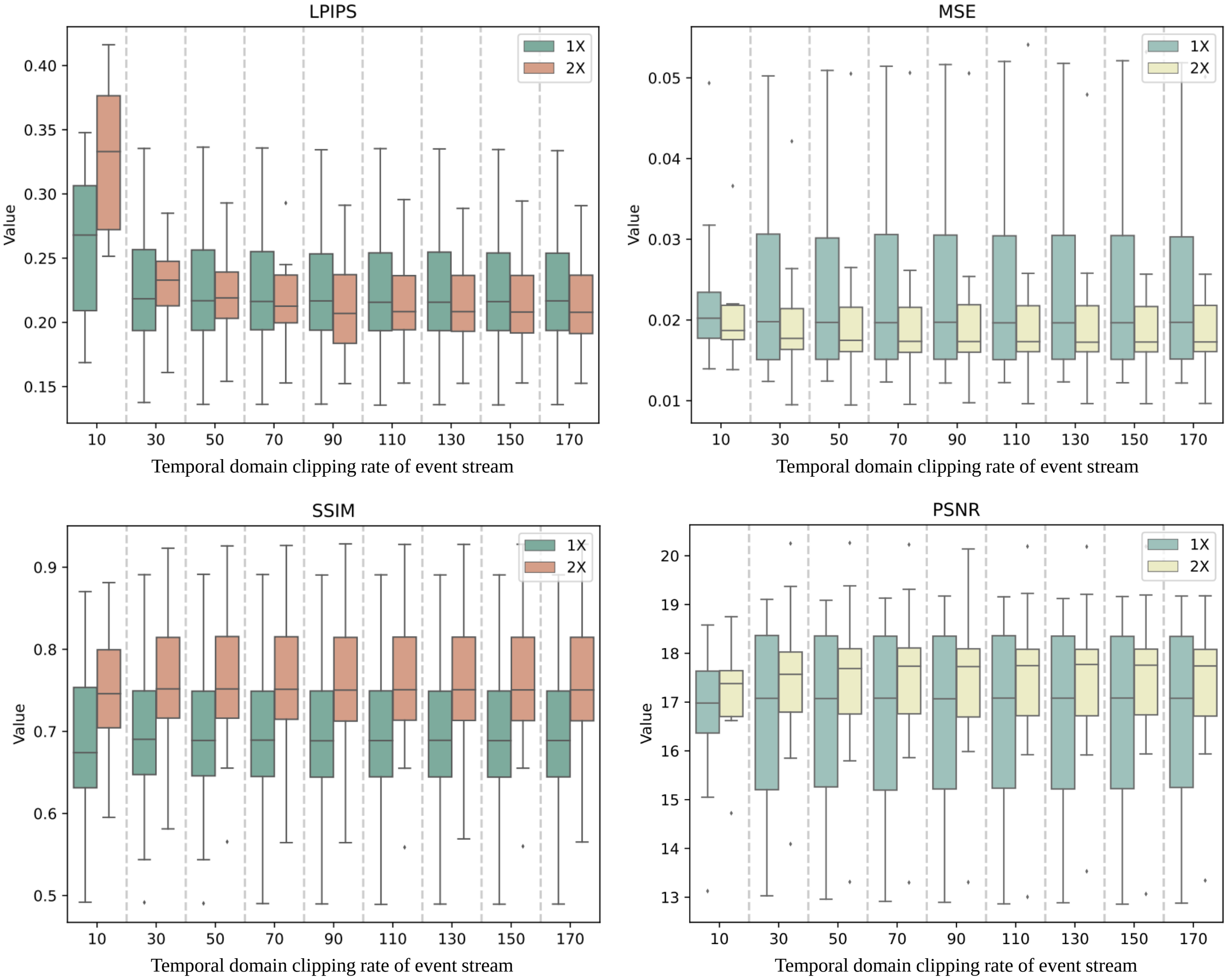


Figure 24: A boxplot to show the comparison results of the event re-distribution effectiveness verification

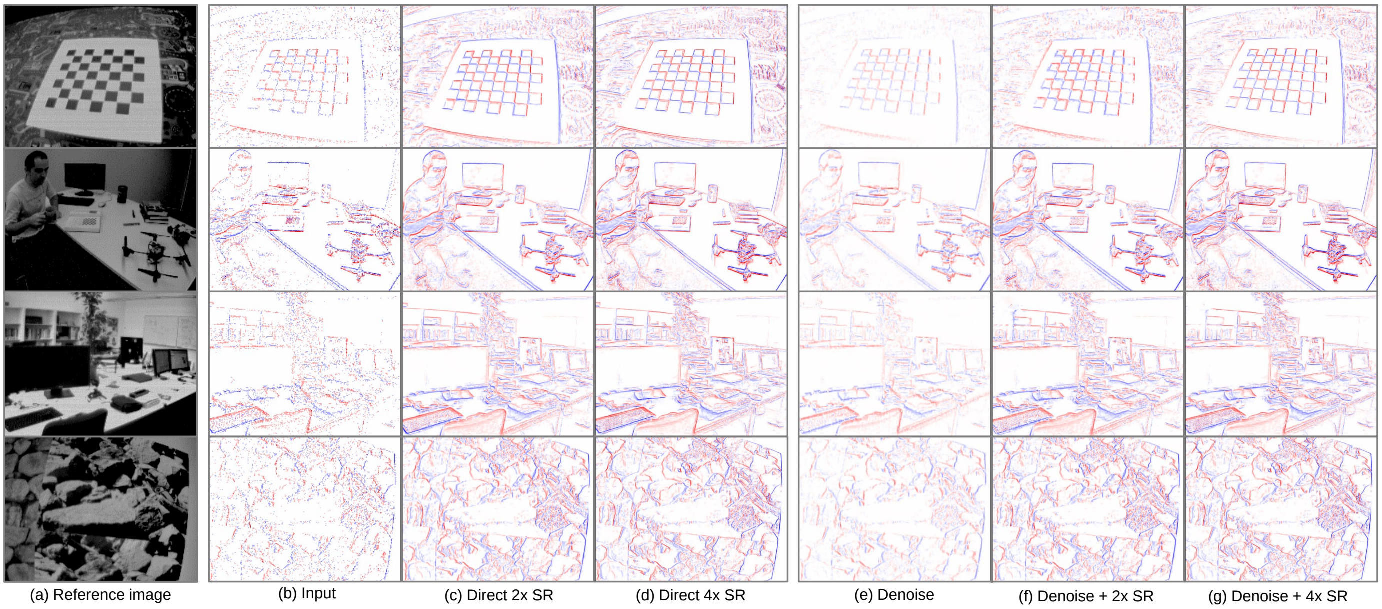


Figure 25: Comparison of the SR results of direct SR strategy and denoise+SR strategy.

error between $\hat{I}^{\text{HR}}$ and the ground truth I^{HR}, then,

$$\begin{aligned} I^{\text{HR}} &= F_{SR}(\hat{I}^{\text{LR}} - n_{\text{image}}) + n_{\text{loss}} \\ &= F_{SR}(\hat{I}^{\text{LR}}) - \widetilde{n}_{\text{image}} + n_{\text{loss}} \\ &= F_{SR}(\hat{I}^{\text{LR}}) + \widetilde{n}_{\text{loss}}, \end{aligned} \tag{12}$$

where the $\widetilde{n}_{\text{image}}$ denotes the new additive noise after the SR process, and the $\widetilde{n}_{\text{loss}} = n_{\text{loss}} - \widetilde{n}_{\text{image}}$. It's interesting to note that the $F_{SR}(\hat{I}^{\text{LR}})$ means the direct SR process for $\hat{I}^{\text{LR}}$ and the $\widetilde{n}_{\text{loss}}$ is additive noise that can be eliminated by a denoising mapping. In other words, the strategy of "input → denoise → SR → output" can also be conducted as the "input → SR → denoise → output" processing. In fact, image SR can be viewed as a denoising process of the initially upsampled image until it approaches the HR ground truth. It has become a consensus among researchers in this field to combine additive noise removal with SR as a single process [9], [10]. The classic image SR method IBP [11] and its upgraded version [12] achieve image SR and denoising synchronously by iterative back-projection. FSRCNN [13] simplifies the network in SRCNN [14], which initially upsamples and then denoises the image, into a unified network and achieves better performance. The merging of SR and denoise has also been adopted by the field of super-resolution microscopy [15], [16].

For neuromorphic signals, we also choose to combine SR and denoising for unified processing because the spikes and events are obtained by integrating or logarithmic domain differencing the potential image sequences. The integration and difference in the time domain will not affect the properties of additive noise, making it possible to apply the above analysis in the image field to the neuromorphic signals field. For intance, eSL-NET++ [17] plugs dual sparse learning module into SR network to remove noise and artifacts, while E2SRI [18] utilizes optical flow to enhance the noisy event stacks in the SR processing. Furthermore, the difficulty of obtaining noise-free HR events forces us to deal with noise simultaneously when learning SR mapping. NeuroZoom uses the noise-to-noise [19] fashion to deal with the absence of noise-free ground truth. This strategy has also been used in fluorescence image SR [20] and OCT image SR [21].

To experimentally compare the performance of "denoise+SR" and "direct SR" approaches, we compare the 2× and 4× SR on the DAVIS240 dataset [2], and record the results in Fig. 25. The corresponding videos are also added to the video of the supplementary material. As can be observed, despite the additional computing power and time consumption required, the "denoise+SR" approach does not achieve significantly improved performance compared to the "direct SR" approach, and the results of the two are almost the same. This experimental result provides evidence for the effectiveness of the "direct SR" strategy.

13 How close is the dataset captured in this paper to real world

To compare the difference between the events captured by the real camera, display-camera system, and simulators, we set up an RGB-Event hybrid imaging system to collect three types of event data from the same scene. We built an RGB-Event hybrid camera system (Fig. 26) consisting of an event camera (Prophesee Gen4.1, with a resolution of 1280×720), a machine vision camera (MV-CA016-10UC, with a resolution of 1440×1080 at a frame rate of 150 FPS), and a beam splitter (Thorlabs CCM1-BS013) mounted in front of the two cameras, providing 50% optical splitting. To perform spatial calibration, we place a screen displaying a flashing checkboard directly in front of the imaging system, and the two cameras then capture the flashing checkboard and calculate the homography between their respective views in order to calibrate them. As for temporal synchronization, we use a signal generator to send a stable square wave signal to both cameras at the same time, ensuring their synchronous capture. The parameters of the two lenses remain consistent.

Table 5: Image reconstruction quantitative comparison for different types of training dataset. The green block marks the better value of V2E [22] and ours.

	V2E [22]				Our			
Sequence	**LPIPS**	**MSE**	**SSIM**	**PSNR**	**LPIPS**	**MSE**	**SSIM**	**PSNR**
bridge_lake_01	0.572	0.055	0.583	12.641	0.464	0.048	0.686	13.261
bridge_lake_03	0.614	0.048	0.583	13.176	0.463	0.088	0.686	10.596
candle	0.845	0.191	0.246	7.283	0.823	0.170	0.265	7.978
fountain_schaffhauserplatz_02	0.641	0.144	0.362	8.420	0.651	0.139	0.380	8.592
kornhausbruecke_letten_random_04	0.666	0.149	0.501	8.351	0.631	0.129	0.522	9.010
lake_01	0.623	0.056	0.570	12.499	0.505	0.056	0.704	12.517
lake_03	0.572	0.065	0.548	11.874	0.566	0.060	0.615	12.228
sihl_03	0.613	0.115	0.501	9.512	0.616	0.133	0.507	8.910
Average	0.643	0.103	0.487	10.470	0.590	0.103	0.546	10.386

We use this system to collect a single sample, which concludes a **real-captured event stream** and a corresponding 150FPS video. Next, we play this video on the display (ASUS PG259QNR, 1920 × 1080) with a refresh rate of $360Hz$ and collect a **display-camera event stream** using the method in Sec. 3.2. Then, we select V2E [22], one of the most widely used event simulators, to generate simulated events. In the first step of using V2E [22], we interpolate the video to a frame rate of $360FPS$ to match the refresh rate of the display, and then generate the **V2E-simulated event stream** with a threshold setting of 0.2. We select clips of these three event data over the same time period ($0.01s$) and show their different 3D views in Fig. 27.

The comparison results clearly indicate that although display-camera events can capture the periodic flow of events caused by the display refreshing, they are more similar to the event distribution of the real-captured events compared to the V2E-simulated events, which exhibit obvious slicing characteristics. In the front view, the V2E-simulated and display-camera events have fewer texture details than the real-captured events, which is due to the low dynamic range and spatial resolution of the video source. However, this does not affect the impact of our proposed dataset for the NDSR tasks, as both LR and HR event data are captured from the same video source playing on the same display.

We further use image reconstruction application E2VID [23] to qualitatively and quantitatively compare whether our dataset is closer to real-captured data than simulated data. We collect two training datasets: our 2× Mutli-E dataset and the simulation dataset generated from V2E [22] using the video sources of Mutli-E. We train the E2VID [23] with the same number of epochs and test its performance on the HS-ERGB dataset [24], which includes real-captured events and the corresponding calibrated ground truth high frame-rate videos. The visual results are shown in Fig. 28 and the corresponding videos are also added to the video of the supplementary material. The visual comparison between the results shows that the model trained by V2E [22] data reconstructs obvious artifacts, while the model trained with our dataset recovers clear details that are closer to the reference image. For quantitative comparison, we select samples from the HS-ERGB dataset [24] to calculate performance metrics, which are recorded in Table 5. In order to avoid invalid evaluations, we exclude the scenes with only local motion since the background regions with no event triggered can cause failure in reconstructing background textures, such as the two examples on the left of Fig. 28. The results demonstrate that models trained on our display-camera data perform better on most samples and metrics, indicating that our data better match the characteristics of the real events and better facilitate NDSR tasks.

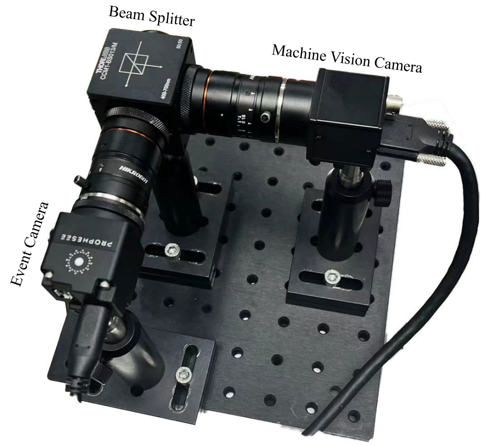


Figure 26: RGB-Event hybrid imaging system.

14 Additional Results

We show some additional results of event denoising, spike denoising and SR, Event-based SR image reconstruction in Fig. 29, Fig. 30, Fig. 31 respectively.

References

[1] L. Zhu, J. Li, X. Wang, T. Huang, and Y. Tian, "NeuSpike-Net: High speed video reconstruction via bio-inspired neuromorphic cameras," in *Proc. of International Conference on Computer Vision*, 2021.

[2] R. Henri, G. Daniel, and S. Davide, "ESIM: an open event camera simulator," in *Proc. of Conference on Robot Learning*, 2018.

[3] M. Yang, S.-C. Liu, and T. Delbruck, "Analysis of encoding degradation in spiking sensors due to spike delay variation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, pp. 145–155, 2017.

[4] G. Gallego, T. Delbrück, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza, "Event-based vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 154–180, 2022.

[5] D. Gehrig and D. Scaramuzza, "Are high-resolution event cameras really needed?," *ArXiv*, vol. abs/2203.14672, 2022.

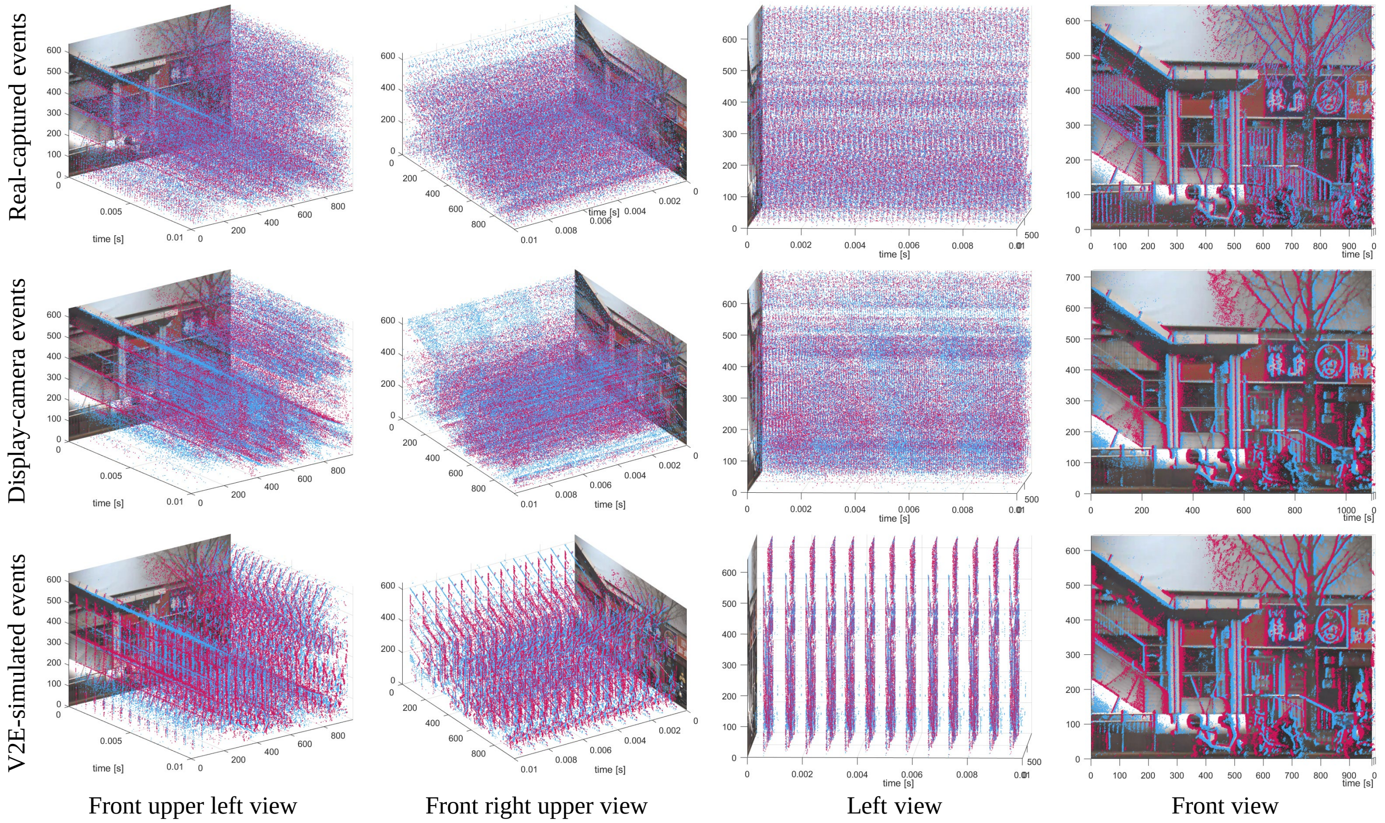


Figure 27: Different views of real-captured events, display-camera events, and V2E-simulated events. The images are captured in sync with the machine vision camera, and we resize them to match the resolution of events.

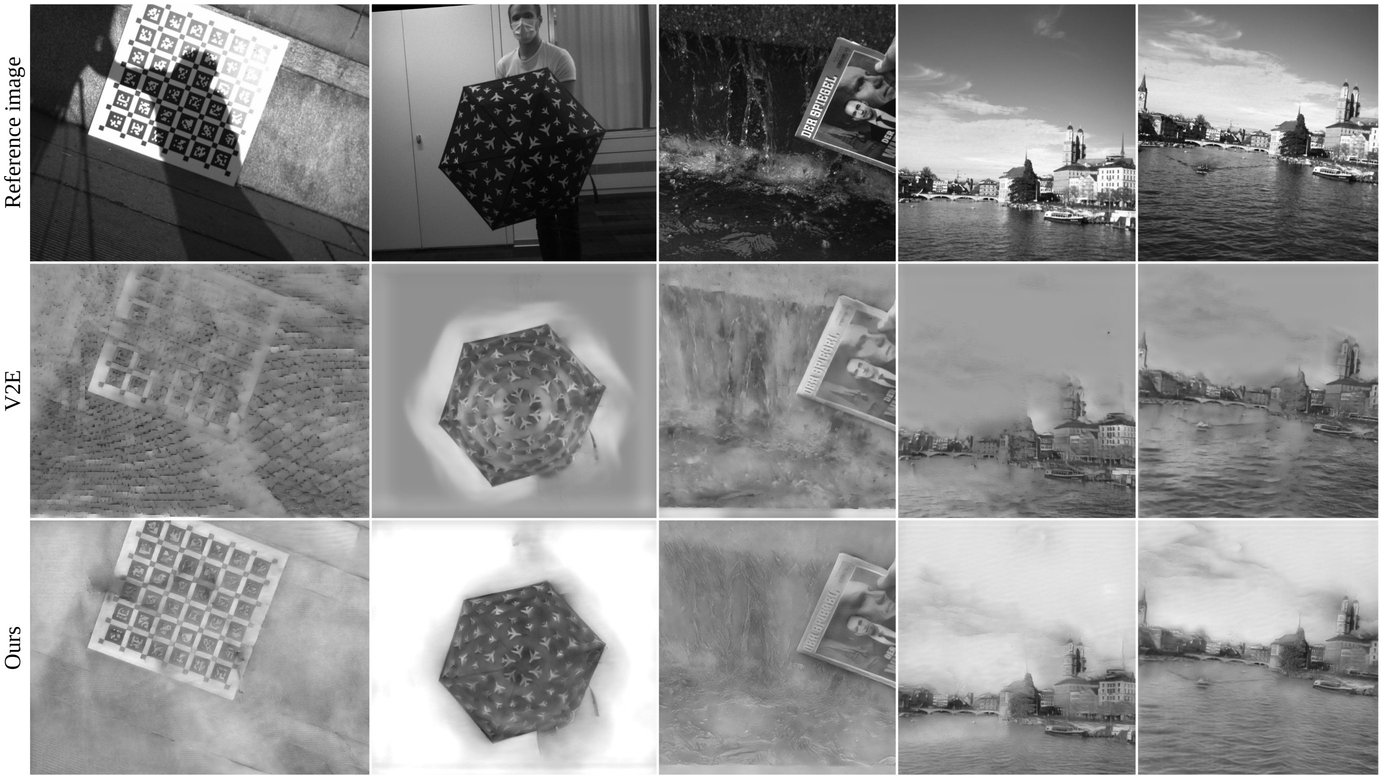


Figure 28: Comparing the image reconstruction performance of E2VID [25] trained with different types of datasets.

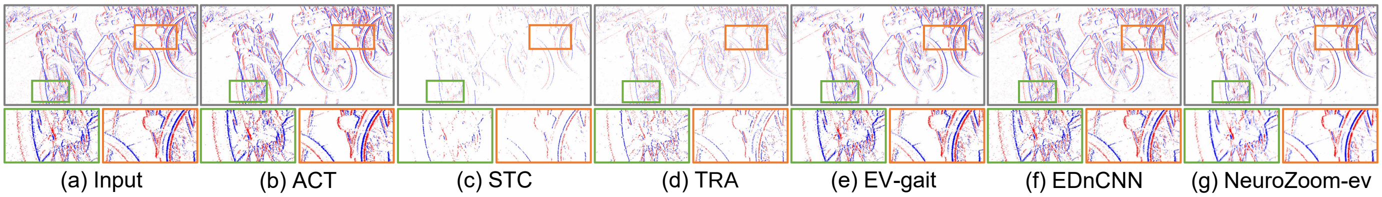

(a) Input (b) ACT (c) STC (d) TRA (e) EV-gait (f) EDnCNN (g) NeuroZoom-ev

Figure 29: Same-resolution denoising comparison results on the real-captured dataset. (a) Event frames clipped from four raw event streams which are captured by a Prophesee Gen 4.0 camera at a spatial resolution of 1280 × 720. (b)-(d) Denoising results of (a), processed by three basic noise filters provided by Prophesee [26]. (e)-(g) Denoising results of (a), processed by EV-gait [27], EDnCNN [6] and the proposed NeuroZoom-ev respectively. Closed-up views of green and orange boxes are shown below the results. Additional denoising results are included in the supplementary video.

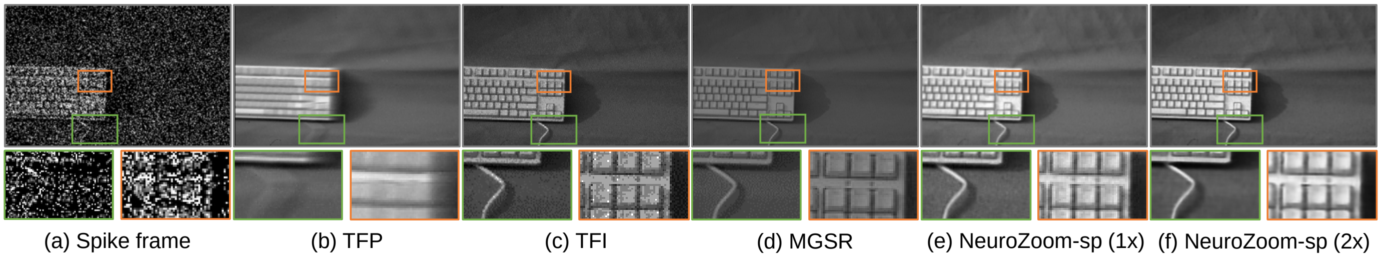

(a) Spike frame (b) TFP (c) TFI (d) MGSR (e) NeuroZoom-sp (1x) (f) NeuroZoom-sp (2x)

Figure 30: Spike denoising and SR (spike-based NDSR) comparison results. Closed-up views of green and blue boxes are shown below the results. Additional results are included in the supplementary video.

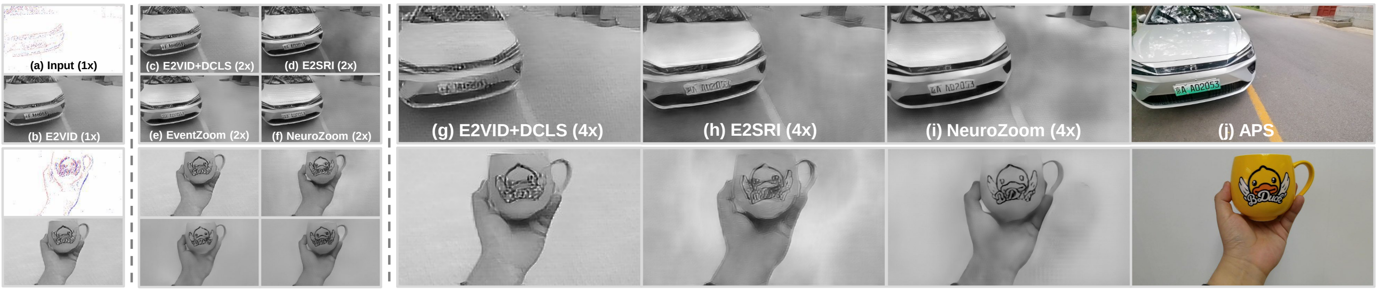


Figure 31: Comparison of event-based image reconstruction on our Multi-E dataset. The caption of each subfigure is labeled on the first sample and applies to all samples: (a) Input event frame. (b) Reconstruct 1× image with E2VID [8]. (c) Reconstruct 1× image with E2VID [8] and then 2× upsample image with DCLS [28]. (d) Reconstruct 2× image directly with E2SRI [29]. (e) Reconstruct 2× event with EventZoom [30] and then reconstruct 2× image with E2VID [8]. (f) 2× NeuroZoom-ev + E2VID [8]. (g) E2VID [8] + 4× DCLS [28]. (h) Reconstruct 4× image directly with E2SRI [29]. (i) 4× NeuroZoom-ev + E2VID [8]. (j) An APS frame.

[6] R. Baldwin, M. Almatrafi, V. Asari, and K. Hirakawa, "Event probability mask (EPM) and event denoising convolutional neural network (EDnCNN) for neuromorphic cameras," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[7] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: learning dense volumetric segmentation from sparse annotation," in *Proc. of Medical Image Computing and Computer-Assisted Intervention*, 2016.

[8] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[9] Y. Huang, J. Li, Y. Hu, X. Gao, and H. Huang, "Transitional learning: Exploring the transition states of degradation for blind super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 6495–6510, 2023.

[10] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, "Feedback network for image super-resolution," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[11] M. Irani and S. Peleg, "Improving resolution by image registration," *Graphical Models and Image Processing*, vol. 53, no. 3, pp. 231–239, 1991.

[12] J.-S. Yoo and J.-O. Kim, "Noise-robust iterative back-projection," *IEEE Transactions on Image Processing*, vol. 29, pp. 1219–1232, 2020.

[13] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. of European Conference on Computer Vision*, 2016.

[14] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, 2016.

[15] C. Qiao, D. Li, Y. Liu, S. Zhang, K. Liu, C. Liu, Y. Guo, T. Jiang, C. Fang, N. Li, Y. Zeng, K. He, X. Zhu, J. Lippincott-Schwartz, Q. Dai, and D. Li, "Rationalized deep learning super-resolution microscopy for sustained live imaging of rapid subcellular processes," *Nature Biotechnology*, vol. 41, pp. 1–11, 2022.

[16] R. Zhou, M. El Helou, D. Sage, T. Laroche, A. Seitz, and S. Süsstrunk, "W2S: microscopy data with joint denoising and super-resolution for widefield to SIM mapping," in *Proc. of European Conference on Computer Vision Workshops*, 2020.

[17] L. Yu, B. Wang, X. Zhang, H. Zhang, W. Yang, J. Liu, and G.-S. Xia, "Learning to super-resolve blurry images with events," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–17, 2023.

[18] M. Mostafavi, Y. Nam, J. Choi, and K.-J. Yoon, "E2SRI: Learning to super-resolve intensity images from events," *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, vol. 44, no. 10, pp. 6890–6909, 2022.

[19] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning image restoration without clean data," in *Proc. of International Conference on Machine Learning*, 2018.

[20] X. Tian, Q. Wu, H. Wei, and Y. Zhang, "Noise2SR: Learning to denoise from super-resolved single noisy fluorescence image," in *Proc. of Medical Image Computing and Computer-Assisted Intervention*, 2022.

[21] B. Qiu, Y. You, Z. Huang, X. Meng, Z. Jiang, C. Zhou, G. Liu, K. Yang, Q. Ren, and Y. Lu, "N2NSR-OCT : Simultaneous denoising and super-resolution in optical coherence tomography images using semisupervised deep learning," *Journal of Biophotonics*, vol. 14, no. 1, p. e202000282, 2021.

[22] Y. Hu, S.-C. Liu, and T. Delbruck, "V2E: From video frames to realistic dvs events," in *Proc. of Computer Vision and Pattern Recognition Workshops*, 2021.

[23] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "High speed and high dynamic range video with an event camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 1964–1980, 2021.

[24] S. Tulyakov, D. Gehrig, S. Georgoulis, J. Erbach, M. Gehrig, Y. Li, and D. Scaramuzza, "Time Lens: Event-based video frame interpolation," in *Proc. of Computer Vision and Pattern Recognition*, 2021.

[25] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proc. of Computer Vision and Pattern Recognition*, pp. 3857–3866, 2019.

[26] E. Perot, P. de Tournemire, D. Nitti, J. Masci, and A. Sironi, "Learning to detect objects with a 1 megapixel event camera," in *Advances in Neural Information Processing Systems*, 2020.

[27] Y. Wang, B. Du, Y. Shen, K. Wu, G. Zhao, J. Sun, and H. Wen, "EV-Gait: Event-based robust gait recognition using dynamic vision sensors," in *Proc. of Computer Vision and Pattern Recognition*, 2019.

[28] Z. Luo, H. Huang, L. Yu, Y. Li, H. Fan, and S. Liu, "Deep constrained least squares for blind image super-resolution," in *Proc. of Computer Vision and Pattern Recognition*, 2022.

[29] S. M. Mostafavi I., J. Choi, and K.-J. Yoon, "Learning to super resolve intensity images from events," in *Proc. of Computer Vision and Pattern Recognition*, 2020.

[30] P. Duan, Z. Wang, X. Zhou, Y. Ma, and B. Shi, "EventZoom: Learning to denoise and super resolve neuromorphic events," in *Proc. of Computer Vision and Pattern Recognition*, 2021.