Ultralight Polarity-Split Neuromorphic SNN for Event-Stream Super-Resolution

Chuanzhi Xu[*†], **Haoxian Zhou**[*], **Langyi Chen, Yuk Ying Chung, Qiang Qu**

School of Computer Science, The University of Sydney, NSW, Australia
chuanzhi.xu@sydney.edu.au

Abstract

Event cameras offer unparalleled advantages such as high temporal resolution, low latency, and high dynamic range. However, their limited spatial resolution poses challenges for fine-grained perception tasks. In this work, we propose an ultra-lightweight, stream-based event-to-event super-resolution method based on Spiking Neural Networks (SNNs), designed for real-time deployment on resource-constrained devices. To further reduce model size, we introduce a novel Dual-Forward Polarity-Split Event Encoding strategy that decouples positive and negative events into separate forward paths through a shared SNN. Furthermore, we propose a Learnable Spatio-temporal Polarity-aware Loss (LearnSTPLoss) that adaptively balances temporal, spatial, and polarity consistency using learnable uncertainty-based weights. Experimental results demonstrate that our method achieves competitive super-resolution performance on multiple datasets while significantly reducing model size and inference time. The lightweight design enables embedding the module into event cameras or using it as an efficient front-end preprocessing for downstream vision tasks.

1 Introduction

Event cameras, also known as neuromorphic cameras or dynamic vision sensors (DVS), are asynchronous sensors that respond to changes in scene brightness. When the brightness change at a given pixel exceeds a certain threshold, an event is triggered and recorded as a tuple (x_k, y_k, t_k, p_k), where (x_k, y_k) denotes the spatial coordinates of the pixel, t_k is the precise timestamp, and $p_k \in \{+1, -1\}$ represents the polarity of brightness change (Gallego et al. 2020; Xu et al. 2025b). The continuous sequence of N events forms an event stream, which can be represented as:

$$EventStream = \{(t_k, x_k, y_k, p_k)\}_{k=1}^{N}. \quad (1)$$

Due to this asynchronous and sparse sensing mechanism, event streams exhibit several unique advantages over traditional frame-based RGB data, including high temporal resolution, high dynamic range (HDR), low latency, and negligible motion blur (Gallego et al. 2020; Posch et al. 2014).

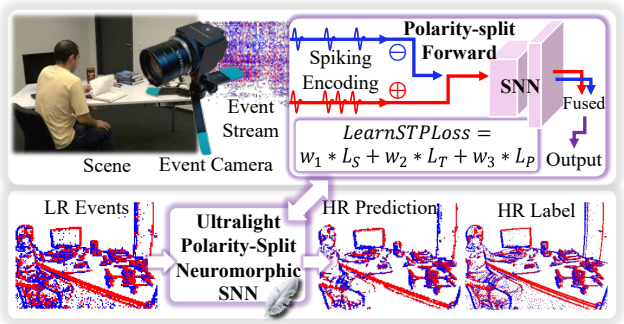

Figure 1: An overview of event stream super-resolution with Ultralight Polarity-Split Neuromorphic SNN.

These properties make them especially promising for applications in 3D reconstruction (Xu et al. 2025a), high-speed robotics (Iaboni et al. 2021), AR/VR (Dong et al. 2024), and autonomous driving (Gehrig and Scaramuzza 2024).

The spatial resolution of most commercially available event cameras ($\leq 640 \times 480$) remains significantly lower than that of frame-based cameras (Chakravarthi et al. 2024). Although high-resolution event sensors such as the Sony IMX646 (1280×720) have been developed (Chakravarthi et al. 2024), achieving higher resolution at the hardware level introduces increased power consumption and cost (Weng, Zhang, and Xiong 2022). Moreover, recent studies suggest that the motivation for developing higher-resolution event cameras may be limited (Gehrig and Scaramuzza 2022). In extreme conditions such as low-light scenes or high-speed motion, high-resolution event cameras may perform worse, since they elevate per-pixel event rates, which in turn amplify temporal noise (Gehrig and Scaramuzza 2022). On the other hand, more studies show that under standard conditions, low-resolution event data limits fine-grained perception and downstream performance, while inputting high-resolution events can improve tasks like object recognition and image/video reconstruction (Li et al. 2021; Weng, Zhang, and Xiong 2022; Huang et al. 2024; Liang et al. 2024).

Event stream super-resolution (EventSR) emerges as the only technological direction that reconciles both perspectives: it aims to reconstruct high-resolution (HR) events from

[*]These authors contributed equally.
[†]Corresponding author.

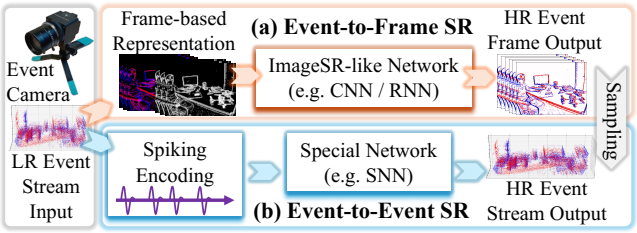

Figure 2: Event-to-frame SR compresses the temporal dimension in exchange for spatial awareness and typically relies on heavier image SR models. Event-to-event SR requires specialized networks to directly generate asynchronous event streams, maintaining the nature of events.

low-resolution (LR) input, as shown in Figure 1. Without the need to develop costly high-resolution event cameras, researchers can super-resolve event data in appropriate scenarios to achieve improved downstream task performance.

Existing event super-resolution methods can be broadly categorized into two types, referring to Figure 2 and Section 2. Event-to-frame super-resolution converts event streams into frame-based representations such as event stacks or event count maps (Li, Li, and Shi 2019; Duan et al. 2021; Liang et al. 2024), and then applies reconstruction algorithms or networks similar to those used in image super-resolution. The frame-based output is then uniformly or randomly sampled on the temporal dimension to produce high-resolution event streams. As many downstream vision tasks still require frame-based inputs to interface with processing modules like CNNs, the loss of temporal precision caused during this process is often considered acceptable. The second type is event-to-event stream-based super-resolution, which directly reconstructs high-resolution event streams without any temporal sampling (Li et al. 2021). It preserves the asynchronous and temporally precise nature of event data, while recovering such temporally asynchronous events requires specialized networks such as Spiking Neural Networks (SNNs) (Li et al. 2021).

However, for the motivation of embedding a controllable super-resolution module on lightweight event cameras, only the event-to-event super-resolution is suitable. An event super-resolution module should not assume the processing of subsequent vision tasks. It is necessary to recover high-temporal-resolution event stream data in this case.

More importantly, an ideal event super-resolution module should be **lightweight**, **energy-efficient**, and capable of **real-time** processing. It is impractical to allocate a high-end chip solely for running a heavy model on an event camera, and it is also undesirable to dedicate substantial computational resources to a super-resolution module within a vision processing pipeline as a step of preprocessing. However, recent methods have made this module still heavy (Huang et al. 2024; Liang et al. 2024).

Therefore, we propose an ultra-lightweight, real-time, stream-based event-to-event super-resolution network based on SNN. Figure 1 provides an overview of our method. It not only improves super-resolution accuracy but also significantly reduces model parameters, further accelerating inference. This makes it feasible to embed lightweight event super-resolution modules into event cameras or use them as energy-efficient visual preprocessing units.

Our contributions can be summarized as follows:

- We propose an ultra-lightweight, SNN-based event-to-event super-resolution network that achieves higher super-resolution accuracy while enabling real-time deployment on resource-constrained devices.
- We introduce a neuromorphic forward propagation strategy named Dual-Forward Polarity-Split Event Encoding, which further reduces the model size by half and improves the spatio-temporal precision.
- By integrating our proposed Learnable Spatio-temporal Polarity-aware Loss (LearnSTPLoss), our method achieves superior reconstruction accuracy across multiple datasets, and the super-resolved event streams further enhance downstream tasks of object recognition and image reconstruction.

2 Related Work

2.1 Event-to-Frame Super-Resolution

Before 2020, event-to-frame super-resolution was mainly based on mathematical modeling. Li et al. were the first to address this task (Li, Li, and Shi 2019). They modeled the event stream at each pixel as a non-homogeneous Poisson process, using event count maps to recover the number of events in the spatial domain and spatiotemporal filters to estimate the event rate function in the temporal domain. Later, Wang et al. proposed the Guided Event Filtering method (Wang et al. 2020), which jointly filtered frame images and event data to improve spatial denoising in event super-resolution.

Subsequently, deep learning-based methods were developed. In 2021, Duan et al. introduced EventZoom (Duan et al. 2021), a method based on a 3D U-Net architecture with an event-to-image module to leverage high-resolution image features, achieving better results than previous non-learning approaches. In 2022, Weng et al. proposed RecEvSR (Weng, Zhang, and Xiong 2022), which used an RNN with a spatial attention mechanism to enhance detail in key areas, showing strong performance for high upsampling rates.

More recent works have explored separating polarities and feeding them into different branches of a network, with interaction modules to exchange information. In 2024, Huang et al. proposed BMCNet (Huang et al. 2024), a bilateral mining and complementary fusion framework that extracts positive and negative polarity features separately, then fuses them to improve detail recovery. Liang et al. proposed RMFNet (Liang et al. 2024), which combines a dual-branch architecture with a recursive structure to enhance temporal modeling across frames and further improve performance.

2.2 Event-to-Event Super-Resolution

Only a few methods are event-to-event super-resolution that preserve the asynchronous and temporally precise nature of the original event stream.

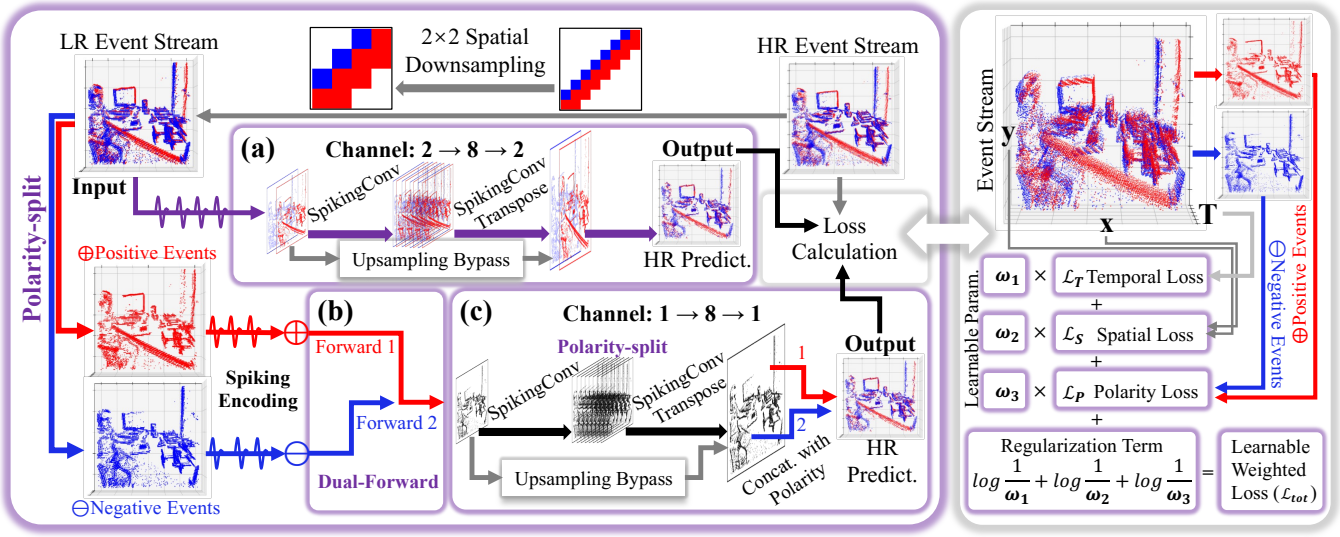

Figure 3: Architecture of key modules in our proposed event super-resolution method. Left: Main architecture of our method. Right: Composition of the Learnable Spatio-temporal Polarity-aware Loss function. (a) Dual-layer EventSR Network. (b) Dual-Forward Polarity-Split Event Encoding Strategy. (c) Ultra-lightweight Polarity-Split EventSR Network.

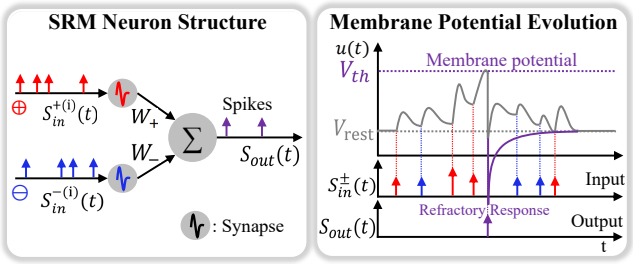

Figure 4: SRM-based spiking neuron structure and spike triggering mechanism.

In 2021, Li et al. introduced an innovative approach using SNNs for the event super-resolution task (Li et al. 2021). SNNs can directly process raw event streams and output high-resolution asynchronous event streams, maintaining the original temporal characteristics. In addition, they designed spatiotemporal constraints to enable the network to learn both spatial and temporal distributions of events. This method serves as the primary baseline for our work.

3 Methods

3.1 SRM-based Spiking Neuron Encoding

We adopt a Spiking Neural Network (SNN) based on the Spike Response Model (SRM) (Gerstner and Kistler 2002) to address the task of event stream super-resolution. Unlike conventional artificial neurons that rely on continuous and differentiable activation functions, SRM-based spiking neurons encode and transmit information through temporal spike trains, simulating biological neural behavior, as shown in Figure 4.

Let $s_{\text{in}}^{(i)}(t)$ and $s_{\text{out}}(t)$ denote the input and output neuron spike trains, respectively. Each neuron maintains an internal membrane potential $u(t)$, which integrates incoming spikes via synaptic weights. The contribution from each input spike train is first processed through a spike response kernel $\epsilon(t)$ to generate a Post Synaptic Potential (PSP), representing the temporal influence of a spike. This PSP is then scaled by a synaptic weight $w^{(i)}$ before being accumulated into the membrane potential. The PSP is computed as a temporal convolution of $s_{\text{in}}^{(i)}(t)$ with $\epsilon(t)$. When $u(t)$ surpasses a threshold V_{th}, the neuron emits an output spike and triggers a refractory mechanism that suppresses but does not reset the membrane potential. This mechanism is controlled by a time constant τ_{ref} and suppression coefficient λ, which together determine the duration and strength of the inhibition. The membrane potential dynamics are given by:

$$u(t) = \sum_i w^{(i)} \cdot (\epsilon(t) * s_{\text{in}}^{(i)}(t)) + (\gamma(t) * s_{\text{out}}(t)), \quad (2)$$

where $*$ denotes convolution, and $\gamma(t)$ is the refractory response kernel.

Considering a feed-forward SNN consists of L layers. The membrane potential and spiking output at layer $l+1$ are:

$$u^{(l+1)}(t) = \mathbf{W}^{(l)} \cdot \left[\epsilon(t) * s^{(l)}(t)\right] + \gamma(t) * s^{(l+1)}(t), \quad (3)$$

$$s^{(l+1)}(t) = \sum_{t_k \in \{t | u^{(l+1)}(t) = V_{\text{th}}\}} \delta(t - t_k), \quad (4)$$

where $\mathbf{W}^{(l)}$ denotes the synaptic weight matrix, and δ is the Dirac function representing spike times.

The specific forms of the response kernels are:

$$\epsilon(t) = \frac{t}{\tau_s} \exp\left(1 - \frac{t}{\tau_s}\right) \cdot \Theta(t), \quad (5)$$

$$\gamma(t) = -\lambda \cdot \exp\left(-\frac{t}{\tau_r}\right) \cdot \Theta(t), \quad (6)$$

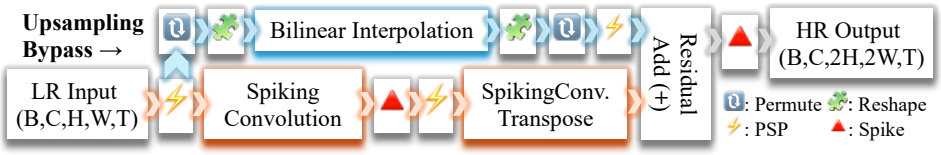

Figure 5: Illustration of the main network path and the PSP-based residual upsampling bypass.

where τ_s and τ_r are time constants for the spike and refractory kernels, respectively, λ is the configurable suppression coefficient, and $\Theta(t)$ is the Heaviside step function.

3.2 Dual-layer EventSR SNN

Most previous works tend to increase the complexity of neural architectures for event stream super-resolution to achieve better performance (Huang et al. 2024; Liang et al. 2024). However, the baseline work has shown that for SNN-based event stream super-resolution, deeper networks may actually degrade performance (Li et al. 2021), which supports our motivation to further simplify the network design.

Layer	Channels	Kernel Size	Stride	Padding
SpikingConv.	$2 \Rightarrow 8$	5×5	1	2
SpikingConv. Transpose	$8 \Rightarrow 2$	2×2	2	0

Table 1: Configuration of Dual-layer EventSR Network.

We propose a compact yet effective SNN-based event stream super-resolution network, simply named **Dual-layer SNN** in this paper. As shown in Figure 3 and Table 1, this network consists of two primary spiking convolution layers. The input and output of the network contain two channels, representing positive and negative event streams. The first layer extracts local spatiotemporal spike patterns and generates intermediate spike responses. The second layer upsamples the output of the first layer back to the original resolution. As shown in Figure 5, it also incorporates an upsampling bilinear-interpolated PSP bypass to preserve fine details from the low-resolution input for final reconstruction.

Layer	V_{th}	τ_s	τ_r	λ	τ_ρ	ρ
SpikingConv.	30	1	1	1	1	10
SpikingConv. Transpose	100	4	4	1	10	100

Table 2: Neuron parameter configuration. τ_ρ and ρ control the time constant and scaling factor of the surrogate gradient used during backpropagation.

For neuron parameters, we adopt a configuration (see Table 2) that enables rapid response to sparse inputs in early layers and stronger integration in later layers. This helps the network effectively learn the mapping for temporal density restoration and spatial upsampling, without relying on deep layer stacking.

3.3 Dual-Forward Polarity-split Event Encoding

Recent studies have proposed using dual-branch networks to separately process positive and negative events (Huang et al. 2024; Liang et al. 2024), demonstrating that handling polarities separately leads to improved performance on event super-resolution. However, the additional network branches significantly increase the model's weight.

Algorithm 1: Dual-Forward Polarity-Split Event Encoding

Require: Input LR event stream $e^{(\text{in})}$

1: Split into polarity channels: $e^+ \leftarrow e^{(\text{in})}_{0,:,:,:}$, $e^- \leftarrow e^{(\text{in})}_{1,:,:,:}$
2: Forward pass separately/concurrently:
3: $\quad 1st: \hat{e}^+ \leftarrow \mathcal{F}(e^+;\theta)$
4: $\quad 2nd: \hat{e}^- \leftarrow \mathcal{F}(e^-;\theta)$
5: Concatenate: $\hat{e}^{(\text{out})} \leftarrow \text{Concat}(\hat{e}^+, \hat{e}^-)$
6: Compute total loss: $\mathcal{L}_{\text{total}} \leftarrow \text{LearnSTPLoss}(e_{\text{out}}, e_{\text{gt}})$
7: Backpropagate: $\theta \leftarrow \theta - \nabla_\theta \mathcal{L}_{\text{total}}$

We novelly propose a polarity-aware event data forward strategy - Dual-Forward Polarity-Split Event Encoding (See Figure 3b and Alg. 1), simply named **Dual-Forward strategy** in this paper. Instead of treating the polarity channels as a single input, we decouple the input event stream tensors $e^{(\text{in})} \in \mathbb{R}^{2\times H\times W\times T}$ into positive and negative streams:

$$e^+ = e^{(\text{in})}_{0,:,:,:}, \quad e^- = e^{(\text{in})}_{1,:,:,:}, \tag{7}$$

where e^+ and e^- denote the event tensors corresponding to positive and negative polarities, respectively.

These are then forwarded independently through the shared SNN-based network $\mathcal{F}(\cdot;\theta)$:

$$\hat{e}^+ = \mathcal{F}(e^+;\theta), \quad \hat{e}^- = \mathcal{F}(e^-;\theta). \tag{8}$$

Then, integrating the two outputs as the final output:

$$\hat{e}^{(\text{out})} = \text{Concat}(\hat{e}^+, \hat{e}^-) \in \mathbb{R}^{2\times H'\times W'\times T}. \tag{9}$$

This integrated output is used to calculate the loss, followed by a joint backpropagation.

This strategy offers many advantages. It removes the need for double-channel event handling within each SNN layer, significantly reducing model size and parameter count. In the experimental phase, running these two forward propagations concurrently also speeds up inference. Moreover, it preserves polarity-specific spatiotemporal dynamics via dedicated forward paths while maintaining unified training through shared losses.

3.4 Ultra-lightweight EventSR SNN

We integrate the Dual-Forward Polarity-Split Event Encoding strategy into our dual-layer SNN event stream super-resolution network, simply named **Ultralight SNN** in this

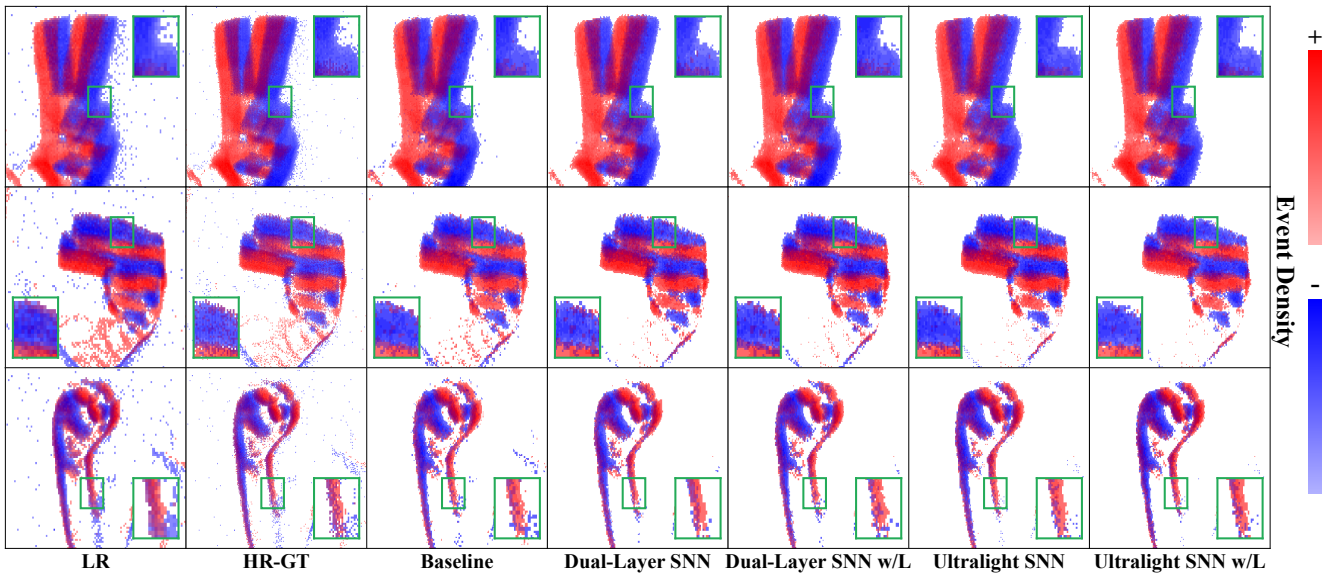

Figure 6: Visualizations on the ASL-DVS. Positive (red) and negative (blue) events are accumulated on each pixel.

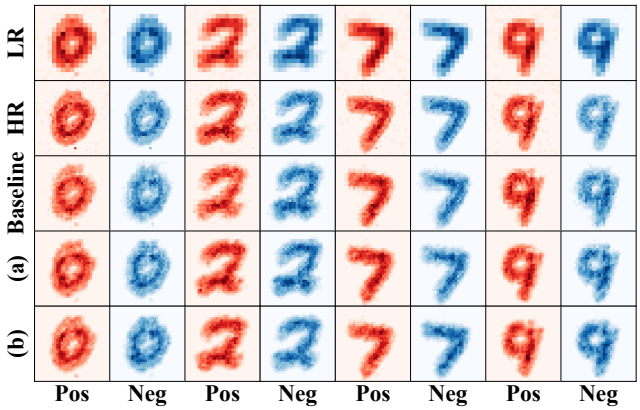

Figure 7: Visualizations on the N-MNIST, split by polarity. (a) Dual-Layer SNN w/Loss. (b) Ultralight SNN w/Loss.

paper. As shown in Figure 3c, since events are forward-propagated separately by polarity, the model only requires a single input and output channel (reduced from 2 to 1), with polarity information encoded within independent propagation groups. The model becomes significantly more lightweight. Moreover, this design helps spatio-temporal perception while introducing only a minor impact on polarity accuracy, as shown in the experimental results.

3.5 Learnable Spatio-temporal Polarity-aware Loss

Inspired by previous work that performs the spatio-temporal learning (Li et al. 2021) and weighted uncertainty loss (Kendall, Gal, and Cipolla 2018), we extend the spatio-temporal only loss with polarity awareness and learnable adaptive weighting, namely **Learn**able **S**patio-**t**emporal **P**olarity-aware Loss (**LearnSTPLoss**), simply marked as "w/L" in this paper. Given the predicted spike output e_{out} and the ground truth spike stream e_{gt}, we have:

Temporal Loss. $\mathcal{L}_{\text{T}}$ maintains temporal accuracy by dividing events into T time windows and comparing them frame by frame.

$$\mathcal{L}_{\text{T}} = \frac{1}{T}\sum_{t=1}^{T} \|e_{\text{out}}(:,:,:,:,t) - e_{\text{gt}}(:,:,:,:,t)\|_2^2 . \quad (10)$$

Spatial Loss. $\mathcal{L}_{\text{S}}$ accumulates spike values over B temporal bins $\mathcal{T}_b$ to measure deviation in spatial projections.

$$\mathcal{L}_{\text{S}} = \sum_{b=1}^{B} \left\| \sum_{t\in\mathcal{T}_b} e_{\text{out}}(:,:,:,:,t) - \sum_{t\in\mathcal{T}_b} e_{\text{gt}}(:,:,:,:,t) \right\|_2^2 \quad (11)$$

Polarity-aware Loss. To enhance polarity fidelity, we introduce a polarity-aware loss that evaluates the discrepancy across positive and negative event channels independently:

$$\mathcal{L}_{\text{P}} = \left\| e_{\text{out}}^{(+)} - e_{\text{gt}}^{(+)} \right\|_2^2 + \left\| e_{\text{out}}^{(-)} - e_{\text{gt}}^{(-)} \right\|_2^2 . \quad (12)$$

Learnable Weighted Total Loss. Preserving time, space, and polarity information is essential to the event stream super-resolution task. When using a combined loss function, previous methods often rely on manually set hyperparameters to weight each component (Li et al. 2021). However, such manual tuning lacks adaptability across different datasets or model architectures. To address this issue, we introduce a learnable weighting mechanism based on the logarithm of variance, enabling the weights of each loss term to be jointly balanced and optimized with the network parameters during training. This learnable loss formulation is expected to improve super-resolution accuracy without compromising the model's lightweight design.

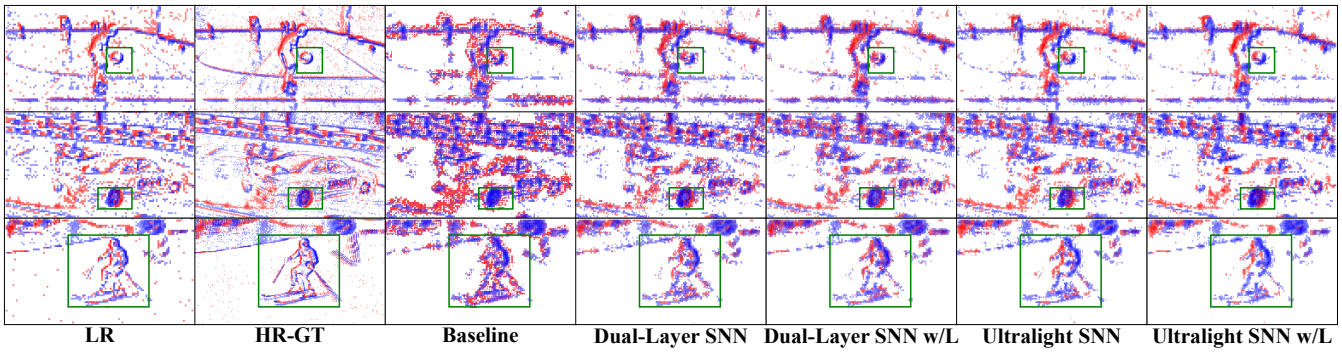

Figure 8: Visualizations on the EventNFS-real. (Zoom up to see the performance on edges.)

The final total loss can be calculated as:

$$\mathcal{L}_{\text{total}} = w_1 \cdot \mathcal{L}_{\text{T}} + w_2 \cdot \mathcal{L}_{\text{S}} + w_3 \cdot \mathcal{L}_{\text{P}} + \log \frac{1}{w_1} + \log \frac{1}{w_2} + \log \frac{1}{w_3}, \tag{13}$$

where each weight w_i represents a learnable parameter with uncertainty proxy σ_i^2:

$$w_i = \exp(-\log \sigma_i^2). \tag{14}$$

This ensures each weight remains positive, and the regularization term avoids collapsing weights to zero.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate our method on five datasets, as summarized in Table 3. The HR event streams correspond to the original event data in these datasets. To generate the LR event streams, we perform spatial downsampling by merging events within each 2×2 region using a stride of 2, which is shown in Figure 3.

Implementation Details. We implement our SNN architecture using SlayerSNN (Shrestha and Orchard 2018), with a discretized simulation step size of 1 millisecond to update membrane potentials and determine spike firing. We train our model for 30 epochs using the Adam optimizer with an initial learning rate of 0.1. Table 3 provides the batch size for training each of the datasets. All experiments are conducted on an NVIDIA RTX 5090 GPU.

We conduct 2× spatial super-resolution experiments (i.e., doubling both Height (H) and Width (W)), which is sufficient to cover practical use cases for embedding super-resolution modules in event cameras.

4.2 Evaluation Metrics

To quantitatively evaluate the performance of event stream super-resolution, referring to our baseline (Li et al. 2021), we implement a Root Mean Squared Error (RMSE) metric that considers both temporal (MSE_T) and spatial (MSE_S) precision. The predicted and ground-truth event streams are represented as four-column arrays (t, x, y, p) and discretized into 4D voxel tensors of shape $(C{=}2, H, W, T)$, where the two channels correspond to positive and negative polarities. In the formulations below, (i, j) index pixel locations, t denotes the temporal coordinate within the interval $[T_0, T_1]$, N_p is the number of active pixels, k indexes the temporal bins, and N_b is the total number of bins.

Dataset	Train/Test	Sensor	bs
N-MNIST (Orchard et al. 2015)	60k / 10k	ATIS	64
CIFAR10-DVS (Li et al. 2017)	8.5k / 1.5k	DVS128	8
ASL-DVS (Bi et al. 2019)	80.6k / 20.2k	DAVIS240C	32
EventCameraD. (Mueggler et al. 2017)	10.1k / 1.6k	DAVIS240C	64
EventNFS-real (Duan et al. 2021)	74.2k / 10.5k	DAVIS346	64

Table 3: Datasets and Batch Size (bs) information.

Temporal Consistency Error (MSE_T). The temporal error is computed by summing the squared differences of the voxel tensors across all spatiotemporal locations (Li et al. 2021). Following our detailed definition, it is formally expressed as:

$$\text{MSE}_T = \frac{1}{N_p} \sum_{i,j} \int_{T_0}^{T_1} \left(\text{Spike}_{i,j}^{h}(t) - \text{Spike}_{i,j}^{gt}(t) \right)^2 dt. \tag{15}$$

Spatial Consistency Error (MSE_S). To evaluate local event distribution over time, the event stream is divided into non-overlapping time blocks. For each block, a Peri-Stimulus Time Histogram (PSTH) is computed by summing voxel counts along the time dimension. MSE_S is defined as:

$$\text{MSE}_S = \frac{1}{N_p} \sum_{k=1}^{N_b} \sum_{i,j} \left\| \text{PSTH}_{i,j}^{h}(k) - \text{PSTH}_{i,j}^{gt}(k) \right\|_2^2. \tag{16}$$

Root Mean Squared Error (RMSE_{ST}). The final RMSE score is defined as:

$$\text{RMSE}_{\text{ST}} = \sqrt{\frac{1}{(T_1 - T_0) \cdot N_p} (\text{MSE}_T + \text{MSE}_S)}. \tag{17}$$

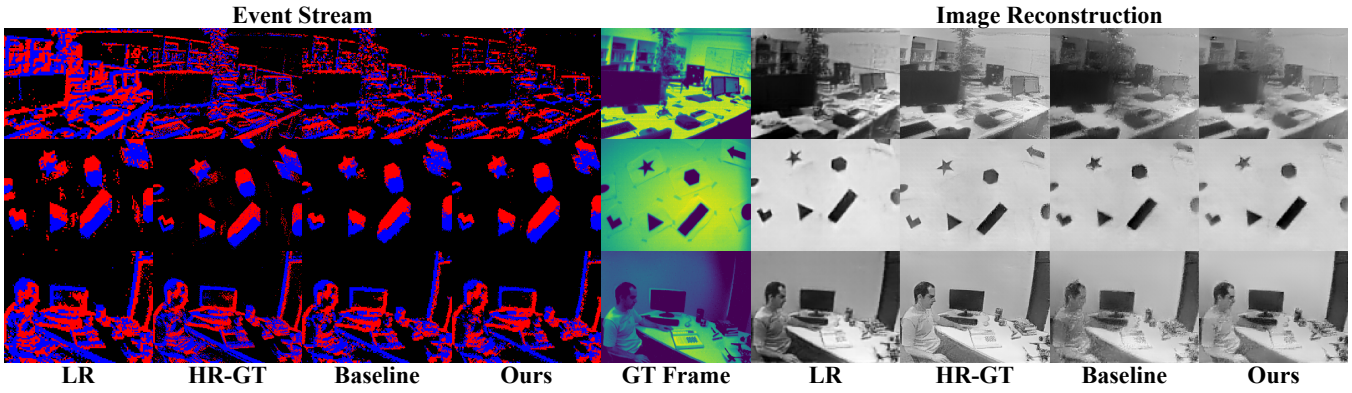

Figure 9: Visualizations of Image Reconstruction on Event Camera Dataset. Our method shown here is Ultralight SNN w/LearnSTPLoss, and it performs better on edges. (Zoom up to see the performance on edges.)

Method	N-MNIST				CIFAR10-DVS				ASL-DVS				Event Camera Dataset				EventNFS-real			
	$RMSE_{ST}$	MSE_S	MSE_T	PA (%)	$RMSE_{ST}$	MSE_S	MSE_T	PA (%)	$RMSE_{ST}$	MSE_S	MSE_T	PA (%)	$RMSE_{ST}$	MSE_S	MSE_T	PA (%)	$RMSE_{ST}$	MSE_S	MSE_T	PA (%)
Baseline	.2720	.1440	.2301	96.36	.1792	.1011	.1479	69.42	.2290	.1055	.2008	98.80	.3250	.2753	.1723	83.82	.3448	.2414	.2460	76.69
D-L	.2621	.1292	.2280	97.34	.1781	.1002	.1473	69.92	.2250	.1048	.1975	99.39	.3165	.2593	.1697	89.06	.3180	.2160	.2329	87.46
D-L w/L	**.2607**	**.1277**	**.2272**	**97.87**	.1772	.1017	.1451	**70.17**	.2250	**.1038**	.1982	**99.42**	.3130	.2626	.1675	89.82	.3105	.2148	.2237	87.65
U	.2614	.1289	.2274	95.63	.1754	**.0992**	.1446	68.28	.2247	.1069	.1961	98.87	.3128	.2626	**.1674**	**89.85**	.3113	.2123	.2270	89.35
U w/L	.2611	.1283	.2274	95.74	**.1747**	.0994	**.1437**	68.78	**.2236**	.1067	**.1948**	98.93	**.3117**	**.2592**	.1699	89.03	**.3061**	**.2118**	**.2202**	**89.96**

Table 4: Comparison of methods and baseline. Dual-Layer SNN and Ultralight SNN are marked as "D-L" and "U". LearnSTPLoss is marked as "w/L". (**The best**; The second best.)

Polarity Accuracy (PA). We also compute a new metric, the polarity accuracy (PA), defined as:

$$\mathrm{PA} = \frac{\left|\left\{(x,y,t) \mid p^{\mathrm{out}}_{x,y,t} = p^{\mathrm{gt}}_{x,y,t}\right\} \cap \Omega\right|}{|\Omega|}, \qquad (18)$$

where Ω denotes the set of shared spatiotemporal coordinates that are present in both the predicted and ground truth event streams. This metric reflects the model's ability to preserve event polarity information, while, to the best of our knowledge, previous event super-resolution methods have not explicitly incorporated this metric.

4.3 Comparative Results

Results of Super-resolved Event Stream. Table 4 shows the comparison results between our methods and the baseline across five datasets. All four variants of our methods outperform the baseline on all event stream evaluation metrics, demonstrating the clear superiority of our dual-layer SNN architecture. Figure 6 and Figure 7 show visualizations on the ASL-DVS and N-MNIST datasets, where our methods produce sharper details along dense edges and exhibit better suppression of edge-related noise. Figure 8 demonstrates the performance of our method on the EventNFS-real dataset, offering visual comparability with other event-to-frame super-resolution methods.

Results of Model Lightweight. According to Table 5, our method achieves extreme lightweight efficiency. Compared to the baseline, the Ultralight SNN reduces the parameter count by 78%. A trained Ultralight SNN on EventNFS-real occupies only 4.0KB of storage. Meanwhile, our inference speed surpasses all other methods, enabling the super-resolved event stream to appear on the monitor with virtually no perceptible delay from the scene.

Method	Params	FLOPs	Time
SNN-based Event SR Methods			
Dual-Layer SNN w/L (ours)	464 (0.464K)	**0.53G**	**1.49ms**
Ultralight SNN w/L (ours)	**232 (0.232K)**	0.58G	1.55†ms
Baseline (Li et al. 2021)	1040 (1.04K)	1.10G	1.62ms
Traditional Image/Video SR Methods			
SRFBN-esr (Li et al. 2019)	2.1M (2100K)	39.5G	37.3ms
RLSP-esr (Fuoli, Gu, and Timofte 2019)	1.2M (1200K)	23.1G	–
RSTT-esr (Geng et al. 2022)	3.8M (3800K)	22.3G	61.4ms
Event-to-Frame Event SR Methods			
EventZoom (Duan et al. 2021)	11.5M (11500K)	65.3G	17.4ms
RecEvSR (Weng, Zhang, and Xiong 2022)	1.8M (1800K)	2.80G	13.2ms
BMCNET (Huang et al. 2024)	2.6M (2600K)	35.35G	–
RMFNET (Liang et al. 2024)	3.0M (3000K)	8.73G	7.0ms

Table 5: Comparison of model size, computational cost, and inference time on EventNFS dataset. †: Ultralight SNN uses concurrent dual forwards. Parts of the data are from (Huang et al. 2024).

Results of Downstream Application. Referring Table 6, we evaluate our method on the image reconstruction task using the Event Camera Dataset and E2VID (Rebecq et al. 2019). The results show that our method improves SSIM by 7% and reduces MSE by 4.48% compared to the baseline. Figure 9 visualizes the reconstruction, where zoomed-up regions reveal that our method preserves clearer edge details.

Additionally, we conduct classification tasks on N-MNIST, ASL-DVS, and CIFAR10-DVS datasets. Our method consistently improves the performance (See Ap-

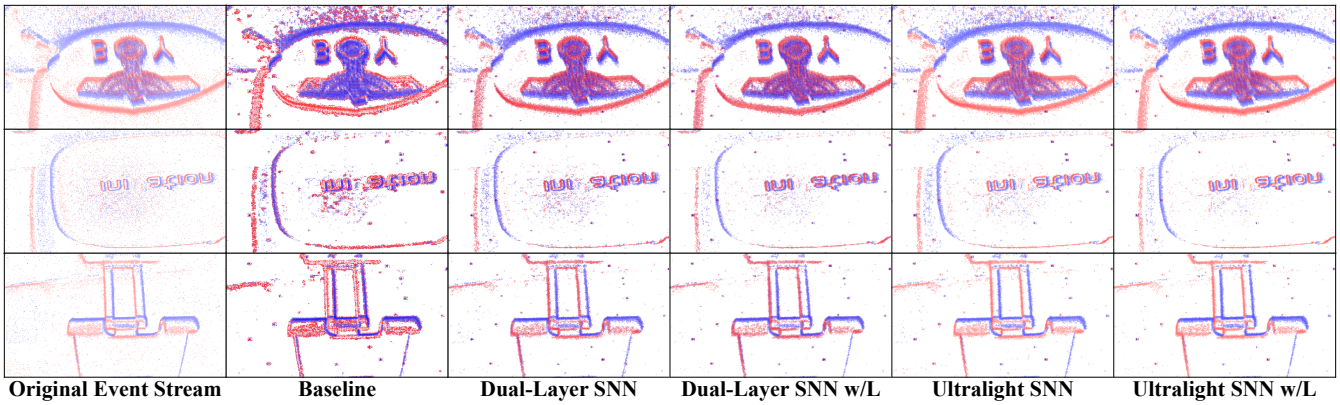

Figure 10: Visualization Results on Real-world Deployment.

Metric	LR	HR-GT	(a)	Baseline	(b)	(c)
$RMSE_{ST}$	–	–	0.837	0.325	0.313	**0.312**
SSIM↑	0.460	0.624	0.497	0.530	0.563	**0.567**
MSE↓	0.084	0.057	0.074	0.067	0.066	**0.064**

Table 6: Results of Image Reconstruction on Event Camera Dataset. (a) (Li, Li, and Shi 2019) (b) Dual-Layer SNN w/L (c) Ultralight SNN w/L

Method	$RMSE_{ST}$	MSE_S	MSE_T	PA (%)
Baseline	0.2720	0.1440	0.2301	96.36
Baseline + DualForward	0.2669	**0.1389**	0.2278	94.34
Baseline + w/L	0.2696	0.1415	0.2295	**97.14**
Baseline + DualForward + w/L	**0.2661**	0.1404	**0.2260**	96.12

Table 7: Ablation study of Dual-Forward Strategy and LearnSTPLoss on N-MNIST with baseline method.

pendix (Xu et al. 2025c)). These results also demonstrate that higher-resolution event data leads to better performance on these tasks, further validating the significance of super-resolving event data.

Extended Validation. Since event-to-frame methods evaluate only the spatial quality of their frame-based HR outputs, our method is not directly comparable at the event stream level. In the appendix, we provide further analytical comparisons in terms of spatiotemporal quality. We also deploy a DVXplorer-Lite event camera in a mobile scenario to collect real-world data and evaluate our method, demonstrating its robustness in performing 2× super-resolution on raw event streams. The visualization results can be found in Figure 10. (See Appendix (Xu et al. 2025c) for more details)

4.4 Ablation Study

According to Table 4, for the three spatio-temporal accuracy metrics, the Ultralight SNN outperforms the Dual-layer SNN on all datasets except N-MNIST. However, it achieves lower PA in three datasets. This suggests that the Dual-Forward strategy, by separating the learning and inference of positive and negative polarities, effectively mitigates spatio-temporal interference between polarities, resulting in better spatio-temporal super-resolution awareness, though at the cost of polarity estimation stability. In addition, when comparing SNNs trained with only spatial and temporal losses, the proposed LearnSTPLoss consistently improves nearly all evaluation metrics.

We also evaluated the Dual-Forward strategy and LearnSTPLoss individually on the baseline three-layer SNN using the N-MNIST dataset. As shown in Table 7, both components independently enhance the baseline performance, and their combination further improves the results.

5 Conclusion & Future Work

In conclusion, we present an ultra-lightweight, event-to-event stream-based super-resolution approach using SNNs. The proposed Dual-layer SNN structure and Dual-Forward Polarity-Split Event Encoding strategy enable separate processing of positive and negative events, reducing model size and computational cost while enhancing spatio-temporal consistency. Furthermore, the Learnable Spatio-temporal Polarity-aware Loss (LearnSTPLoss) adaptively balances spatial, temporal, and polarity fidelity, leading to improved reconstruction accuracy across diverse datasets. Experimental results demonstrate that our approach surpasses the baseline on five benchmark datasets and downstream tasks such as image reconstruction and object classification. Our method is the most lightweight and fastest in inference among all mainstream event-based super-resolution methods, making real-time, on-device deployment feasible.

In the future, we plan to deploy our model on real SNN hardware chips, such as Intel Loihi, to increase the temporal precision of spiking. We expect that this can further improve the results and speed, and will be more energy-saving.

References

Bi, Y.; Chadha, A.; Abbas, A.; Bourtsoulatze, E.; and Andreopoulos, Y. 2019. Graph-based object classification for neuromorphic vision sensing. In *Proceedings of the*

IEEE/CVF international conference on computer vision, 491–501.

Chakravarthi, B.; Verma, A. A.; Daniilidis, K.; Fermuller, C.; and Yang, Y. 2024. Recent event camera innovations: A survey. In *European Conference on Computer Vision*, 342–376. Springer.

Dong, Y.; Chen, Z.; He, X.; Li, L.; Shu, Z.; Cao, Y.; Feng, J.; Liu, S.; Li, C.; and Wang, J. 2024. SEVAR: a stereo event camera dataset for virtual and augmented reality. *Frontiers of Information Technology & Electronic Engineering*, 25(5): 755–762.

Duan, P.; Wang, Z. W.; Zhou, X.; Ma, Y.; and Shi, B. 2021. EventZoom: Learning to denoise and super resolve neuromorphic events. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12824–12833.

Fuoli, D.; Gu, S.; and Timofte, R. 2019. Efficient video super-resolution through recurrent latent space propagation. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 3476–3485. IEEE.

Gallego, G.; Delbrück, T.; Orchard, G.; Bartolozzi, C.; Taba, B.; Censi, A.; Leutenegger, S.; Davison, A. J.; Conradt, J.; Daniilidis, K.; et al. 2020. Event-based vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(1): 154–180.

Gehrig, D.; and Scaramuzza, D. 2022. Are high-resolution event cameras really needed? *arXiv preprint arXiv:2203.14672*.

Gehrig, D.; and Scaramuzza, D. 2024. Low-latency automotive vision with event cameras. *Nature*, 629(8014): 1034–1040.

Geng, Z.; Liang, L.; Ding, T.; and Zharkov, I. 2022. Rstt: Real-time spatial temporal transformer for space-time video super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17441–17451.

Gerstner, W.; and Kistler, W. M. 2002. *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press.

Huang, Z.; Liang, Q.; Yu, Y.; Qin, C.; Zheng, X.; Huang, K.; Zhou, Z.; and Yang, W. 2024. Bilateral event mining and complementary for event stream super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 34–43.

Iaboni, C.; Patel, H.; Lobo, D.; Choi, J.-W.; and Abichandani, P. 2021. Event camera based real-time detection and tracking of indoor ground robots. *IEEE Access*, 9: 166588–166602.

Kendall, A.; Gal, Y.; and Cipolla, R. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7482–7491.

Li, H.; Li, G.; and Shi, L. 2019. Super-resolution of spatiotemporal event-stream image. *Neurocomputing*, 335: 206–214.

Li, H.; Liu, H.; Ji, X.; Li, G.; and Shi, L. 2017. Cifar10-dvs: an event-stream dataset for object classification. *Frontiers in neuroscience*, 11: 244131.

Li, S.; Feng, Y.; Li, Y.; Jiang, Y.; Zou, C.; and Gao, Y. 2021. Event stream super-resolution via spatiotemporal constraint learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4480–4489.

Li, Z.; Yang, J.; Liu, Z.; Yang, X.; Jeon, G.; and Wu, W. 2019. Feedback network for image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3867–3876.

Liang, Q.; Huang, Z.; Zheng, X.; Yang, F.; Peng, J.; Huang, K.; and Tian, Y. 2024. Efficient event stream super-resolution with recursive multi-branch fusion. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*.

Mueggler, E.; Rebecq, H.; Gallego, G.; Delbruck, T.; and Scaramuzza, D. 2017. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM. *The International journal of robotics research*, 36(2): 142–149.

Orchard, G.; Jayawant, A.; Cohen, G. K.; and Thakor, N. 2015. Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in neuroscience*, 9: 437.

Posch, C.; Serrano-Gotarredona, T.; Linares-Barranco, B.; and Delbruck, T. 2014. Retinomorphic event-based vision sensors: bioinspired cameras with spiking output. *Proceedings of the IEEE*, 102(10): 1470–1484.

Rebecq, H.; Ranftl, R.; Koltun, V.; and Scaramuzza, D. 2019. High speed and high dynamic range video with an event camera. *IEEE transactions on pattern analysis and machine intelligence*, 43(6): 1964–1980.

Shrestha, S. B.; and Orchard, G. 2018. Slayer: Spike layer error reassignment in time. *Advances in neural information processing systems*, 31.

Wang, Z. W.; Duan, P.; Cossairt, O.; Katsaggelos, A.; Huang, T.; and Shi, B. 2020. Joint filtering of intensity images and neuromorphic events for high-resolution noise-robust imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1609–1619.

Weng, W.; Zhang, Y.; and Xiong, Z. 2022. Boosting event stream super-resolution with a recurrent neural network. In *European Conference on Computer Vision*, 470–488. Springer.

Xu, C.; Chen, L.; Chen, H.; Chung, V.; and Qu, Q. 2025a. Towards End-to-End Neuromorphic Voxel-based 3D Object Reconstruction Without Physical Priors. *arXiv preprint arXiv:2501.00741*.

Xu, C.; Zhou, H.; Chen, L.; Chen, H.; Zhou, Y.; Chung, V.; Qu, Q.; and Cai, W. 2025b. A Survey of 3D Reconstruction with Event Cameras. *arXiv preprint arXiv:2505.08438*.

Xu, C.; Zhou, H.; Chen, L.; Chung, Y. Y.; and Qu, Q. 2025c. Ultralight Polarity-Split Neuromorphic SNN for Event-Stream Super-Resolution. *arXiv preprint arXiv:2508.03244*.