Spk2SRImgNet: Super-Resolve Dynamic Scene from Spike Stream via Motion Aligned Collaborative Filtering

Yuanlin Wang[1,2], Yiyang Zhang[1,2], Ruiqin Xiong[1,2,*], Jing Zhao[1,2],
Jian Zhang[3], Xiaopeng Fan[4], Tiejun Huang[1,2]
[1]School of Computer Science, Peking University
[2]National Key Laboratory for Multimedia Information Processing, Peking University
[3]School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University
[4]School of Computer Science and Technology, Harbin Institute of Technology
wangyuanlin@stu.pku.edu.cn
{yiyangz, rqxiong, jzhaopku, zhangjian.sz, tjhuang}@pku.edu.cn fxp@hit.edu.cn

Abstract

Spike camera is a kind of neuromorphic camera that records dynamic scenes by firing a stream of binary spikes with extremely high temporal resolution. It demonstrates great potential for vision tasks in high-speed scenarios. One limitation in its current implementation is the relatively low spatial resolution. This paper develops a network called Spk2SRImgNet to super-resolve high resolution images from low resolution spike stream. However, fluctuations in spike stream hinder the performance of spike camera super resolution. To address this issue, we propose a motion aligned collaborative filtering (MACF) module, which is motivated by key ideas in classic image restoration schemes to mitigate fluctuations in spike data. MACF leverages the temporal similarity of spike stream to acquire similar features from neighboring moments via motion alignment. To separate disturbances from features, MACF filters these similar features jointly in transform domain to exploit representation sparsity, and generates refinement features that will be used to update initial fluctuated features. Specifically, MACF designs an inverse motion alignment operation to map these refinement features back to their original positions. The initial features are aggregated with the repositioned refinement features to enhance reliability. Experimental results demonstrate that the proposed method achieves state-of-the-art performance compared with existing methods.

1. Introduction

High-speed imaging attracts increasing attention in emerging computer vision applications such as autonomous driving [15, 17] and robotics [16, 29]. Conventional camera employs an exposure-based imaging model, which limits its ability to capture high-speed scenes. Specifically, conventional camera accumulates photoelectric signals over an exposure time window to form a snapshot image, which requires the scene to be still during the exposure period. Otherwise, a single point on a moving object may be projected onto different sensor pixels, resulting in motion blur.

Figure 1. Illustration of spike camera super resolution (SCSR) and visual comparison (×4) with existing SCSR methods: VidarSR [42] and SpikeSR-Net [50]. Please zoom in for better comparison.

Neuromorphic camera is an emerging camera that abandons the concept of exposure window and mimics the working mechanism of biological retina. It records intensity information with high temporal resolution, showing great potential for high speed vision tasks. One kind of neuromorphic camera is event camera [2, 25, 26, 31]. It detects light intensity changes of each pixel in the logarithmic domain and fires an event when the change exceeds a specified threshold. Another kind of neuromorphic camera is spike camera [8, 9, 19]. Each pixel of spike camera independently and continuously accumulates photons, and fires a spike when the accumulation exceeds a predefined threshold. Unlike event camera that only records *relative* changes of light intensity, spike camera captures *absolute* intensity by firing a stream of binary spikes, providing clearer information for reconstructing image details.

Benefiting from its extremely high temporal resolution

*Corresponding author.

(*e.g*. 40kHz), spike camera could sample a large amount of intensity information in a short period. Besides, these samples cover different locations of the scene due to the relative motion between camera and scene. Therefore, there is promising potential to reconstruct high resolution (HR) images from spike stream by converting the dense sampling in temporal domain to spatial domain.

This paper aims to super-resolve high resolution images from low resolution spike stream. However, due to the Poisson effect of photon arrival, the quantization effect of spike readout, and the thermal noises introduced by circuits, spikes exhibit fluctuations, i.e., the length of spike accumulation period varies over time, even when light intensity remains unchanged. Thus, the features extracted from fluctuated spikes tend to be noisy and temporally incoherent. These unreliable features provide low-quality information source for super resolution (SR) reconstruction. Moreover, they affect the alignment accuracy when leveraging temporal information, further degrading the quality of SR reconstruction.

In this paper, we propose a spike-to-super-resolution-image network Spk2SRImgNet. To handle spike fluctuations, we take inspiration from several classic image restoration methods, such as BM3D [6], BAS [43], LSSC [27], and WNNM [18]. These methods incorporate the idea of leveraging non-local similarity within natural images to reduce the unknown random disturbance. We embed these concepts into the design of Spk2SRImgNet and develop a motion aligned collaborative filtering (MACF) module to reduce disturbances in spike information. Firstly, MACF obtains similar features from neighboring moments via motion alignment, exploiting the temporal similarity of spike stream. To separate disturbances from features, MACF performs collaborative filtering on grouped similar features in transform domain for leveraging representation sparsity, and generates refinement features for each moment. These refinement features will be utilized to update initial fluctuated features. Specifically, MACF introduces an inverse motion alignment operation to map these multi-moment refinement features to their original unaligned positions. The initial features are integrated with the repositioned refinement features to improve reliability. Experimental results show that the proposed Spk2SRImgNet achieves state-of-the-art performance on synthesized and real-world spike data.

2. Related Work

Video Super Resolution. Video super resolution (VSR) produces high resolution images from low resolution (LR) image sequences. Temporal alignment is a crucial step to model temporal correspondences of misaligned frames. Early learning-based methods [3, 21] employ traditional optical flow estimation [40] for motion compensation. With the diverse development of network structures, TDAN [35] and EDVR [37] utilize deformable convolution [7, 56] for feature alignment. BasicVSR++ [4] leverages optical flow to guide deformable alignment for stable training. VRT [24] proposes a temporal mutual attention block to facilitate motion estimation and feature alignment.

Event Camera Reconstruction and Super Resolution. Event camera monitors relative changes in light intensity with high temporal resolution. Reconstructing intensity images from events has been studied in the past decade. Kim et al. [22] propose to use the extended Kalman filter and Poisson reconstruction for event reconstruction. Bardow et al. [1] estimate both intensity and optical flow using the primal-dual algorithm. With the development of deep learning, E2VID [32], FireNet [20], ET-Net [41], EVSNN [55] have been proposed successively with good performance. To reconstruct HR images from LR events, Wang et al. [36] propose an unsupervised GAN EventSR. Mostafavi et al. [28] propose an end-to-end recurrent network E2SRI.

Spike Camera Reconstruction and Super Resolution. Image reconstruction is a popular topic for spike camera [38, 48, 51–54]. TFI [53] reconstructs images based on the interval between two temporally neighboring spikes. TFP [53] utilizes the number of spikes within a temporal window to estimate the intensity. For better performance, CNN-based [48, 52] and spike neural network (SNN)-based [51, 54] frameworks have been proposed successively. To reconstruct HR images from LR spike stream, Zhao et al. [47] derive the constraint relationship between pixel intensities and each spike period, and obtain HR images based on gradient descent algorithm. Xiang et al. [42] design a multi-level feature learning network to achieve spike camera super resolution. Zhao et al. [50] propose SpikeSR-Net based on the observation model of spike camera. Furthermore, Dong et al. [11] propose a super resolution network for the newly developed color spike camera [10, 12–14].

3. Spike Camera

Working Mechanism. Spike camera contains a pixel array of size $H \times W$ and each pixel works independently. Each pixel comprises three main components: photon receptor, integrator, and comparator, as shown in Fig. 2. The photoreceptor captures incident photons from external scenes and converts them into electric charges through photoelectric conversion. The integrator accumulates the electric charges continuously. Meanwhile, the comparator checks the accumulated voltage persistently. Once the voltage reaches a predefined threshold θ, the pixel sets up a spike flag and resets the integrator, immediately restarting a new "accumulation-and-set" cycle. Spike readout circuit periodically checks the spike flag at a frequency of 40kHz. When detecting the flag, the circuit outputs a spike "1", otherwise, it outputs "0". The voltage accumulated by the in-

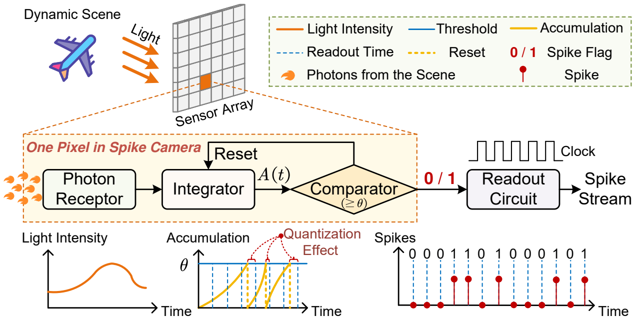

Figure 2. Working mechanism of spike camera.

tegrator at an arbitrary time t can be formulated as [49]:

$$A(x, y, t) = \int_0^t \eta I(x, y, u) du \mod \theta, \tag{1}$$

where $I(x, y, u)$ denotes the arrival rate of photons received by pixel (x, y) at time u, η denotes the photoelectric conversion rate. The output of spike camera is a 3D binary spike matrix $\{S(u)\}_{u=1}^{T}$ with size $H \times W \times T$, where T is the temporal length, $S(u)$ is a spike frame at time u.

Spike Interval Based Reconstruction. According to the working mechanism of spike camera, the stronger the light intensity, the shorter the interval between two neighboring spikes, i.e., the spike interval is inversely proportional to the light intensity. This property is exploited by spike-interval-based reconstruction method (SIR) [53]. Specifically, the spike interval of two temporally neighboring spikes that cover the position (x, y, u) can be formulated as:

$$\begin{aligned}\Phi(x, y, u) = & \, min\{t|S(x, y, t) = 1, t \geq u\} \\ & - max\{t|S(x, y, t) = 1, t < u\}.\end{aligned} \tag{2}$$

SIR estimates the light intensity of pixel (x, y) at time u by

$$Y(x, y, u) = \frac{\delta}{\Phi(x, y, u)}, \tag{3}$$

where δ denotes the maximum dynamic range of reconstruction.

Spike Fluctuations. The imaging of spike camera is affected by multiple kinds of noise. First, the Poisson process of photon arrival causes the time taken for the number of photons to reach the threshold to exhibit randomness, which is denoted as Poisson noise. Second, the spike readout circuit is controlled by a periodic clock signal, i.e., it only outputs spikes at discrete times, which results in a time difference between spike flag set and spike readout (see Fig. 2). This difference leads to quantization noise. Third, the thermal noise introduced by circuits brings randomness to spike fire. For the above reasons, the intervals of neighboring spikes exhibit fluctuations and struggle to accurately reflect light intensity, leading to noisy reconstruction results from SIR.

4. Motivation and Idea

The key problem we aim to address is how to super-resolve the original dynamic scene from captured binary spike data to obtain high quality visual reconstruction with clean image details. According to the working mechanism of spike camera, each individual spike does not reveal the light intensity directly. Instead, it only tells the time point at which a spike firing occurs. Essentially, the instantaneous light intensity can be inferred only by considering the length of intervals between consecutive spikes. Unfortunately, for the reasons discussed above, the recorded spike intervals are not stable, and they fluctuate around an expectation value determined by actual light intensity.

To handle such fluctuations, we draw inspiration from several classic image restoration methods, such as BM3D [6], BAS [43], LSSC [27], and WNNM [18]. Although they are quite diversified in algorithmic design, all these methods incorporate the idea of leveraging non-local similarity within natural image signals to reduce the unknown random disturbance. They utilize a group of similar patches and establish a prior model for them so that a regularization term is formed to separate the true signal from the noise.

Suppose $\mathbf{y} = \mathbf{x} + \mathbf{n}$ is an image corrupted by noise, where $\mathbf{x}$ is the clean image and $\mathbf{n}$ is the noise. The idea of these methods can be summarized as follows:

(1) For each pixel location $\mathbf{p}$, search for a group of similar patches from the noisy image $\mathbf{y}$:

$$\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N = \Lambda(\mathbf{y}|\mathbf{p}), \tag{4}$$

where N is the number of similar patches, $\Lambda(\cdot|\cdot)$ is the operation of searching for and extracting similar patches. Note that the notation $\mathbf{y}_i$ should depend on $\mathbf{p}$ but we omit it for conciseness.

(2) Exploit the correlation within the patch group $\{\mathbf{y}_i\}_{i=1}^{N}$ to generate an estimate for each patch:

$$\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, ..., \tilde{\mathbf{x}}_N = \Pi(\{\mathbf{y}_i\}_{i=1}^{N}), \tag{5}$$

where $\tilde{\mathbf{x}}_i$ is the estimate of the true signal $\mathbf{x}_i$ corresponding to $\mathbf{y}_i$. The actual operation of $\Pi(\cdot)$ depends on the prior model adopted to characterize the consistency between patches $\{\mathbf{y}_i\}_{i=1}^{N}$. Generally, $\Pi(\cdot)$ can be regarded as an operator to solve the following optimization objective:

$$\{\tilde{\mathbf{x}}_i\} = \arg\min \sum_{i=1}^{N} ||\mathbf{x}_i - \mathbf{y}_i||_2^2 + \Phi\left(\{\mathbf{x}_i\}_{i=1}^{N}\right) \tag{6}$$

BM3D [6] solves Eq. (6) by applying a 3D transform to the stacked similar patches and constructing a sparsity regularization term in the transform domain. It utilizes hard-thresholding, soft-thresholding, and Wiener filtering in different steps, corresponding to l_0, l_1, and l_2

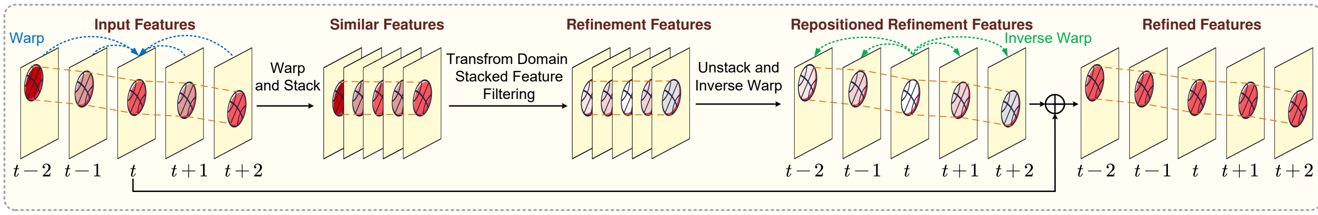

Figure 3. The main idea of motion aligned collaborative filtering. First, we collect temporal similar features via motion alignment. Then, we stack these aligned features and perform transform-domain collaborative filtering to generate refinement features for each moment. Subsequently, the refinement features are unstacked and mapped back to their original positions via inverse warping. These repositioned refinement features are aggregated with input features to obtain refined features with improved consistency.

sparsity norm, respectively. BAS [43] applies a content adaptive 2D decorrelation transform to the similar patches via principle component analysis (PCA), and builds a separate distribution model and regularization term for each transform band, using the transform coefficients of these patches as samples. It leads to a PCA-domain adaptive soft-thresholding to solve Eq. (6). LLSC [27] adopts another prior model that emphasizes that similar patches tend to share the location pattern of non-zero dictionary elements in their sparse decomposition. WNNM [18], on the other hand, utilizes the low-rank property of similar patches so that the problem (6) reduces to singular value decomposition (SVD) and soft-thresholding on the singular values.

(3) Place each estimate $\tilde{\mathbf{x}}_i$ back to the position where $\mathbf{y}_i$ is taken from, and aggregate the overlapping estimates from all position $\mathbf{p}$ to form a whole estimate for $\mathbf{x}$:

$$\tilde{\mathbf{x}} = \Lambda^{-1}\left(\{\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, ..., \tilde{\mathbf{x}}_N\}_{\mathbf{p}}\right). \tag{7}$$

Here $\Lambda^{-1}(\cdot)$ denotes patch reposition and aggregation operation. The reposition of $\tilde{\mathbf{x}}_i$ to $\tilde{\mathbf{x}}$ is the inverse operation of taking $\mathbf{y}_i$ from $\mathbf{y}$ in stage (1).

Taking these ideas into account, we propose a novel framework tailored for spike data to reduce disturbances in spike information. The main idea is illustrated in Fig. 3. The similarity exploited in this framework is temporal similarity, and it may be extended to include spatial non-local similarity in the future. First, we align the spike features at neighboring moments with the spike feature at the reconstruction moment to obtain a set of similar features. Then, we stack the aligned features and perform a neural-network-based transform-domain collaborative filtering, to separate the disturbances from clean features by exploiting representation sparsity and generate refinement features for each moment. These refinement features are used to update the initial input features. Specifically, the refinement features are unstacked and mapped back to their original positions via inverse warping, and then aggregated with input features. The purpose of this framework is to improve the consistency of multi-moment features along the motion trajectory.

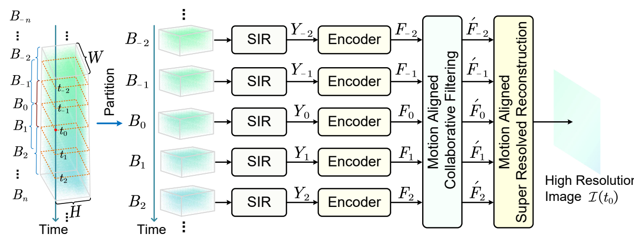

Figure 4. Overall architecture of the proposed spike camera super resolution network Spk2SRImgNet.

5. Method

5.1. Overall Architecture

To reconstruct HR dynamic scenes from LR binary spike stream, we develop an end-to-end trainable spike camera super resolution network, Spk2SRImgNet. The overall architecture is shown in Fig. 4. To reconstruct an HR image $\mathcal{I}(t_0) \in \mathbb{R}^{rH\times rW}$ at time t_0, we take a set of consecutive spike frames $S(t_0+i), i = 0, \pm1, \pm2, ...$ around $S(t_0)$ as input, where $S(t_0+i) \in \mathbb{B}^{H\times W\times 1}$, r is SR scale, $\mathbb{B}$ represents the binary domain.

Firstly, the input spike stream is partitioned into K overlapping short-term spike blocks $\{B_i\}_{i=-n}^{n}(K = 2n+1)$, where $B_i \in \mathbb{B}^{H\times W\times L}$, L is the temporal length of each block, B_0 is the key block centered at time t_0, $\{B_i\}, i = \pm1, \pm2, \ldots$ are neighboring blocks distributed symmetrically on both sides of B_0, B_i is centered at time t_i. To bridge the domain gap between binary spikes and real pixel values, we first convert spike frames in $B_i = \{S(u)\}_{u=t_i-w_h}^{t_i+w_h}$ to intensity frames in $Y_i = \{Y(u)\}_{u=t_i-w_h}^{t_i+w_h} \in \mathbb{R}^{H\times W\times L}$ based on the spike interval based reconstruction (SIR) method [53], where $L = 2w_h+1$. Then, we extract features $\{F_i\}_{i=-n}^{n}$ from $\{Y_i\}_{i=-n}^{n}$ through an encoder, which consists of two convolutional layers and three residual blocks. To reduce fluctuations, $\{F_i\}_{i=-n}^{n}$ are input to motion aligned collaborative filtering (MACF) module, which generates improved features $\{\acute{F}_i\}_{i=-n}^{n}$. Finally, the multi-moment features $\{\acute{F}_i\}_{i=-n}^{n}$ are fed into motion aligned super resolved reconstruction module to generate an HR image $\mathcal{I}(t_0)$.

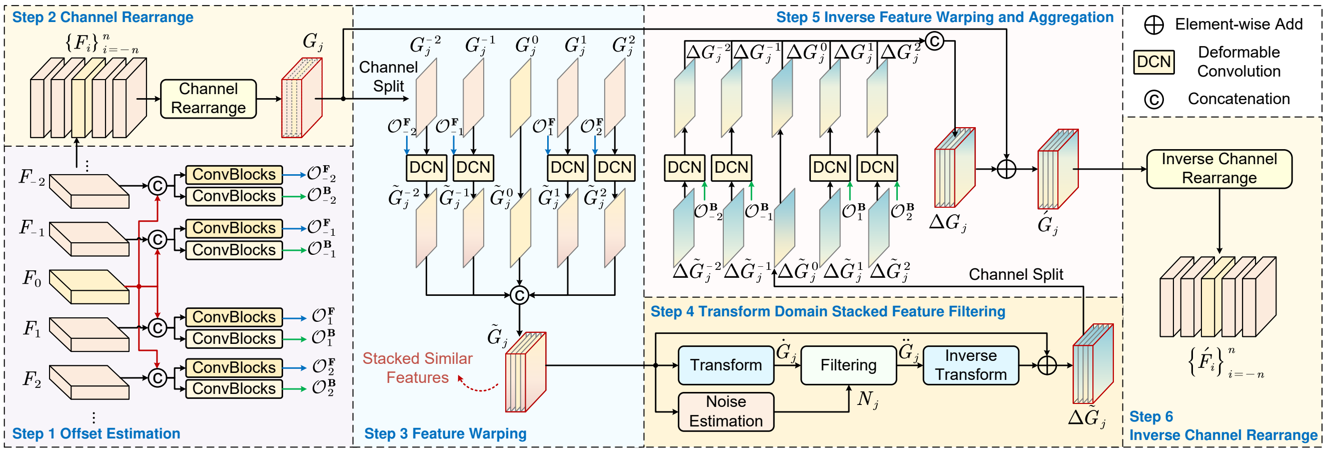

Figure 5. The overall framework of motion aligned collaborative filtering (MACF) module.

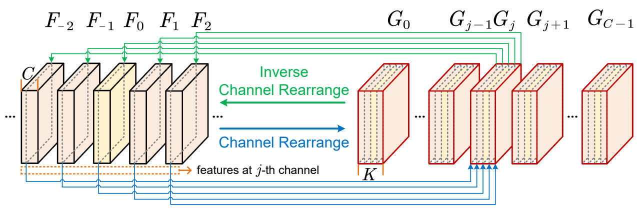

Figure 6. Illustration of channel rearrangement (CR) and inverse channel rearrangement (ICR). The left part is K features $\{F_i\}_{i=-n}^{n}$ at K different moments, where $F_i \in \mathbb{R}^{H\times W\times C}$. The right part is C features $\{G_j\}_{j=0}^{C-1}$, where $G_j \in \mathbb{R}^{H\times W\times K}$, each G_j contains features at K different moments.

5.2. Motion Aligned Collaborative Filtering

We propose a motion aligned collaborative filtering (MACF) module to implement the idea proposed in Sec. 4. The overall framework of MACF is shown in Fig. 5, which includes six steps. We provide a detailed description of each step below.

Step 1: Offset Estimation. This step generates the forward ($F_i \rightarrow F_0$) and backward ($F_0 \rightarrow F_i$) motion offsets between neighboring features $\{F_i\}, i = \pm 1, ..., \pm n$ and key feature F_0. These offsets will be utilized in the subsequent feature warping (step 3) and inverse feature warping (step 5), respectively. Specifically, the forward and backward motion offsets : $\mathcal{O}_i^{\mathbf{F}}$ and $\mathcal{O}_i^{\mathbf{B}}$ are both derived from F_i and F_0, which are formulated as follows ($\forall i \in \{\pm 1, ..., \pm n\}$):

$$\begin{aligned} \mathcal{O}_i^{\mathbf{F}} &= m_1(\{F_i, F_0\}), \\ \mathcal{O}_i^{\mathbf{B}} &= m_2(\{F_i, F_0\}), \end{aligned} \tag{8}$$

where $m_1(\cdot)$ and $m_2(\cdot)$ denote two independent convolutional blocks that consist of several convolutional layers, $\{,\}$ denotes channel concatenation operation.

Step 2: Channel Rearrange. The $\{F_i\}_{i=-n}^{n}$ at K different moments are features extracted from the fluctuated reconstruction $\{Y_i\}_{i=-n}^{n}$, where $F_i \in \mathbb{R}^{H\times W\times C}$, $F_i = \{F_i^0, F_i^1, ..., F_i^{C-1}\}$, the superscript of F_i denotes channel index. Each feature F_i consists of C channels, with each channel encoding different information about the scene. For example, some channels capture edges, some channels focus on textures, and so on. We perform channel rearrangement (CR) to collect data at the same channel index for each feature and concatenate them, as shown in Fig. 6. We form $\{G_j\}_{j=0}^{C-1}$ from $\{F_i\}_{i=-n}^{n}$ via CR operation, where $G_j \in \mathbb{R}^{H\times W\times K}$, $G_j = \{G_j^{-n}, ..., G_j^0, ..., G_j^n\} = \{F_{-n}^j, ..., F_0^j, ..., F_n^j\}$, the subscript of G denotes channel index, the superscript of G denotes time index. G_j contains K features with channel index j at K different moments. To offer an intuitive understanding of this operation, we provide a visualization of CR in supplementary material. After CR, steps 3, 4 and 5 are performed sequentially for each G_j.

Step 3: Feature Warping. Step 2 collects temporally unaligned features that encode the same information. We perform feature warping based on deformable convolution $\mathcal{D}$ to align neighboring features $\{G_j^i\}, i = \pm 1, .., \pm n$ with key feature G_j^0. Specifically, given the forward offsets from neighboring features to key feature $\{\mathcal{O}_i^{\mathbf{F}}\}, i = \pm 1, ..., \pm n$, the aligned features could be obtained by:

$$\tilde{G}_j^i = \mathcal{D}(G_j^i, \mathcal{O}_i^{\mathbf{F}}) = \mathcal{D}(F_i^j, \mathcal{O}_i^{\mathbf{F}}), \forall i \in \{\pm 1, ..., \pm n\}. \tag{9}$$

These aligned features $\{\tilde{G}_j^i\}_{i=-n}^{n}$ are then concatenated to derive $\tilde{G}_j = \{\tilde{G}_j^{-n}, ..., \tilde{G}_j^n\}$, where $\tilde{G}_j^0 = G_j^0 = F_0^j$. $\tilde{G}_j$ could be regarded as stacked similar features.

Step 4: Transform Domain Stacked Feature Filtering. We design a Transform Domain stacked feature Filtering (TDF) step to generate refinement features $\Delta\tilde{G}_j$ from stacked similar features $\tilde{G}_j$ by leveraging representation sparsity in transform domain. TDF is designed with four submodules: transform, noise estimation, filtering, and inverse transform, as depicted in step 4 of Fig. 5. Specifically, the input $\tilde{G}_j$ is first transformed into feature $\dot{G}_j$. Simultaneously, noise feature N_j are extracted from $\tilde{G}_j$ via noise estimation submodule. $\dot{G}_j$ and N_j are passed through filter-

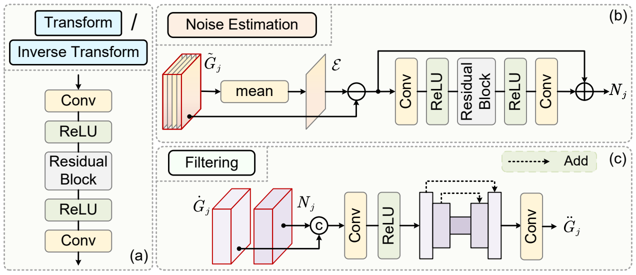

Figure 7. Illustration of submodules in step 4 of MACF. (a) Transform and inverse transform submodules; (b) noise estimation submodule; (c) filtering submodule.

ing submodule to generate $\ddot{G}_j$. $\ddot{G}_j$ are subsequently fed into inverse transform submodule. Additionally, a residual connection is applied in TDF step, producing refinement features $\Delta\tilde{G}_j$ at last.

The above four submodules are illustrated in detail in Fig. 7. Transform and inverse transform submodules are independently realized via several convolutional layers (Fig. 7 (a)), involving both spatial and inter-channel transform. For noise estimation (NE) (Fig. 7 (b)), we first average $\tilde{G}_j$ along the channel dimension to obtain a rough temporal average result $\mathcal{E}$. Then we calculate the residual between $\tilde{G}_j$ and $\mathcal{E}$ to represent fluctuations in features. Lastly, we extract deeper features from the residual to derive noise feature N_j. We concatenate the transformed feature $\dot{G}_j$ and noise feature N_j as input to the filtering submodule (Fig. 7 (c)), which contains a U-Net structure [33] and outputs $\ddot{G}_j$.

Step 5: Inverse Feature Warping and Aggregation. We propose an inverse motion alignment operation to map these multi-moment refinement features $\Delta\tilde{G}_j$ to their original positions. Specifically, given the backward offsets from key feature to neighboring features $\{\mathcal{O}_i^{\mathbf{B}}\}, i = \pm 1, ..., \pm n$, we achieve inverse feature warping by:

$$\Delta G_j^i = \mathcal{D}(\Delta\tilde{G}_j^i, \mathcal{O}_i^{\mathbf{B}}), \forall i \in \{\pm 1, ..., \pm n\}. \quad (10)$$

ΔG_j^i is repositioned refinement feature and corresponds to initial unaligned feature G_j^i. We concatenate these repositioned refinement features and obtain $\Delta G_j = \{\Delta G_j^{-n}, ..., \Delta G_j^n\}$, where $\Delta G_j^0 = \Delta\tilde{G}_j^0$. ΔG_j are then aggregated with initial features G_j to derive refined features with improved consistency:

$$\acute{G}_j = G_j + \Delta G_j. \quad (11)$$

Step 6: Inverse Channel Rearrange. Finally, the features $\{\acute{G}_j\}_{j=0}^{C-1}$ are restored to $\{\acute{F}_i\}_{i=-n}^n$ via inverse channel rearrangement (ICR), i.e., the inverse operation of CR, as shown in Fig. 6.

MACF alleviates disturbances in $\{F_i\}_{i=-n}^n$ and generate improved features $\{\acute{F}_i\}_{i=-n}^n$, which are beneficial for enhancing SR reconstruction quality.

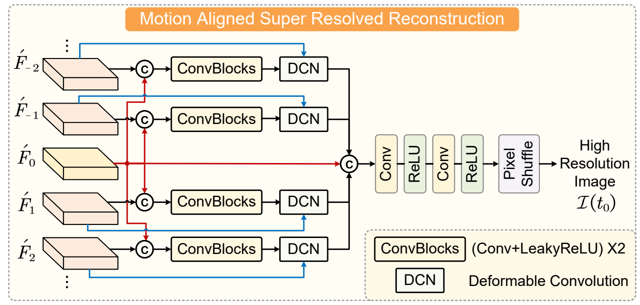

Figure 8. Illustration of motion aligned super resolved reconstruction module.

5.3. Motion Aligned Super Resolved Reconstruction

The improved features $\{\acute{F}_i\}_{i=-n}^n$ are then input to motion aligned super resolved reconstruction module to generate the high resolution image $\mathcal{I}(t_0)$, as illustrated in Fig. 8. Firstly, features at neighboring moments $\{\acute{F}_i\}, i = \pm 1, ..., \pm n$ are aligned with the key feature $\acute{F}_0$ by deformable convolution [7, 56], where the features $\{\acute{F}_i\}_{i=-n}^n$ with improved consistency help enhance the accuracy of motion alignment, benefiting the utilization of temporal information. Subsequently, key feature and aligned neighboring features are concatenated and fused through several convolutional layers. The fused feature is upsampled to the HR image by pixel shuffle operation [3].

6. Experiments

6.1. Implementation Details

We set the number of input spike frames to 101. The number of partitioned short-term blocks is 9 ($K = 9, n = 4$). The temporal length L of each block B_i is 21. The channel C of F_i and $\acute{F}_i$ is 64. During training, we randomly crop the spike stream to a spatial size of 64×64. In addition, we randomly perform horizontal and vertical flips as well as rotation for data augmentation. The batch size is set to 8. We use Adam optimizer [23] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is initialized as 2e-4 and decayed by 0.7 every 100 epochs. The model is trained with $\mathcal{L}_1$ loss for 800 epochs. All models are trained and tested on NVIDIA RTX 3090 GPU.

6.2. Datasets

Synthesized Dataset. We need a large number of "LR spike stream and HR image" pairs to train the model. However, it is challenging to obtain corresponding high-quality HR images, especially in high dynamic scenes. Inspired by the experiments conducted in [48, 50], we use a spike camera simulator (detailed in supplementary material) to mimic the working mechanism of spike camera and generate synthesized SCSR datasets for training and evaluation, where

Scale	Method	REDS-based Dataset			Adobe240-based Dataset			Params(M)	Runtime(s)
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓		
×2	TFP+DAT	25.55	0.5954	0.4456	27.66	0.6782	0.3677	14.65	4.40
	Spk2ImgNet+DAT	31.77	0.9024	0.1907	34.92	0.9604	0.0949	18.56	4.63
	WGSE+DAT	31.54	0.8942	0.2055	34.75	0.9571	0.1022	18.60	4.50
	BSF+DAT	31.92	0.9044	0.1867	35.03	0.9610	0.0934	17.13	5.01
	VidarSR	31.97	0.8964	0.2040	35.13	0.9558	0.1105	10.16	2.09
	SpikeSR-Net	34.29	0.9330	0.1481	37.04	0.9704	0.0838	3.26	2.14
	Ours	34.50	0.9377	0.1403	37.22	0.9725	0.0816	3.72	0.81
×3	TFP+DAT	24.88	0.5754	0.4464	26.17	0.6577	0.3721	14.84	1.88
	Spk2ImgNet+DAT	28.90	0.8293	0.2849	30.50	0.9064	0.1708	18.75	2.10
	WGSE+DAT	28.76	0.8217	0.2961	30.33	0.9013	0.1779	18.78	2.02
	BSF+DAT	28.99	0.8315	0.2807	30.44	0.9054	0.1699	17.32	2.02
	VidarSR	29.55	0.8373	0.2791	32.01	0.9197	0.1685	11.38	0.85
	SpikeSR-Net	31.09	0.8798	0.2304	33.74	0.9439	0.1355	3.30	1.35
	Ours	31.41	0.8850	0.2231	34.11	0.9470	0.1327	3.90	0.37
×4	TFP+DAT	24.35	0.5617	0.4734	25.09	0.6407	0.3995	14.80	1.06
	TFP+BasicVSR++	23.69	0.5023	0.4850	24.21	0.5610	0.4415	7.32	0.08
	Spk2ImgNet+DAT	27.23	0.7678	0.3521	27.97	0.8459	0.2378	18.71	1.15
	Spk2ImgNet+BasicVSR++	27.28	0.7730	0.3527	27.77	0.8421	0.2526	11.23	0.17
	WGSE+DAT	27.15	0.7618	0.3597	27.91	0.8416	0.2427	18.75	1.08
	WGSE+BasicVSR++	27.16	0.7699	0.3561	27.72	0.8404	0.2558	11.08	0.12
	BSF+DAT	27.30	0.7701	0.3473	27.90	0.8451	0.2358	17.28	1.10
	BSF+BasicVSR++	27.24	0.7818	0.3374	27.43	0.8389	0.2481	9.80	0.18
	VidarSR	28.30	0.7960	0.3249	30.18	0.8834	0.2097	12.79	0.75
	SpikeSR-Net	29.22	0.8266	0.2946	31.31	0.9073	0.1885	3.34	1.07
	Ours	29.50	0.8369	0.2816	31.49	0.9119	0.1822	3.86	0.21

Table 1. Quantitative results on REDS-based and Adobe240-based SCSR evaluation dataset. Red and blue indicate the best and the second-best performance, respectively. It is noted that BasicVSR++ [4] only support ×4 SR scale. The runtime is tested on input spike stream of sizes 360 × 640 × 101 (×2), 240 × 426 × 101 (×3) and 180 × 320 × 101 (×4), respectively.

Case	SIR	MACF			PSNR↑	SSIM↑
		step 3	step 4	step 5		
(A)					29.07	0.8207
(B)	✓				29.15	0.8237
(C)		✓			29.17	0.8244
(D)			✓		29.32	0.8306
(E)		✓		✓	29.34	0.8316
(F)		✓	✓		29.35	0.8322
(G)			✓	✓	29.33	0.8304
(H)		✓	✓	✓	29.46	0.8344
(I)	✓	✓	✓	✓	**29.50**	**0.8369**

Table 2. Ablation study of the proposed Spk2SRImgNet on REDS-based SCSR evaluation dataset (×4).

each sample consists of a spike stream-image pair.

Specifically, we use the 120 fps REDS dataset [30] with a spatial resolution of 720 × 1280 to generate the synthesized dataset. REDS includes 240 scenes for training and 30 scenes for evaluation, and we follow its original partition. For each scene, we choose 5 segments, resulting in 240 × 5 = 1200 and 30 × 5 = 150 pairs for training and evaluation, respectively. Each pair consists of spike stream of size 360 × 640 × 101 (×2) / 240 × 426 × 101 (×3) / 180 × 320 × 101 (×4) (101 is the temporal length) and a ground truth image of size 720 × 1280 (×2 and ×4) / 720 × 1278 (×3). To demonstrate the generalization of the proposed model, we generate another SCSR evaluation dataset from Adobe240 dataset [34]. Adobe240 dataset includes 133 videos of 240 fps with resolution of 720 × 1280 taken by hand-held cameras. We follow the test set partition of Adobe240 in [44], which includes 17 scenes. For each scene, we choose 10 segments, generating 17×10 = 170 pairs for evaluation.

Real-World Captured Data. In addition to the synthesized data, we also evaluate the model performance on real-world spike data captured by the spike camera with a spatial resolution of 250 × 400. Due to the absence of ground truth, we perform qualitative analysis for these data.

6.3. Comparison with Existing Methods

We compare our proposed method with existing SCSR method: VidarSR [42] and SpikeSR-Net [50]. Besides, we combine four spike camera image reconstruction methods: TFP [53], Spk2ImgNet [48], WGSE [45] and BSF [52] with

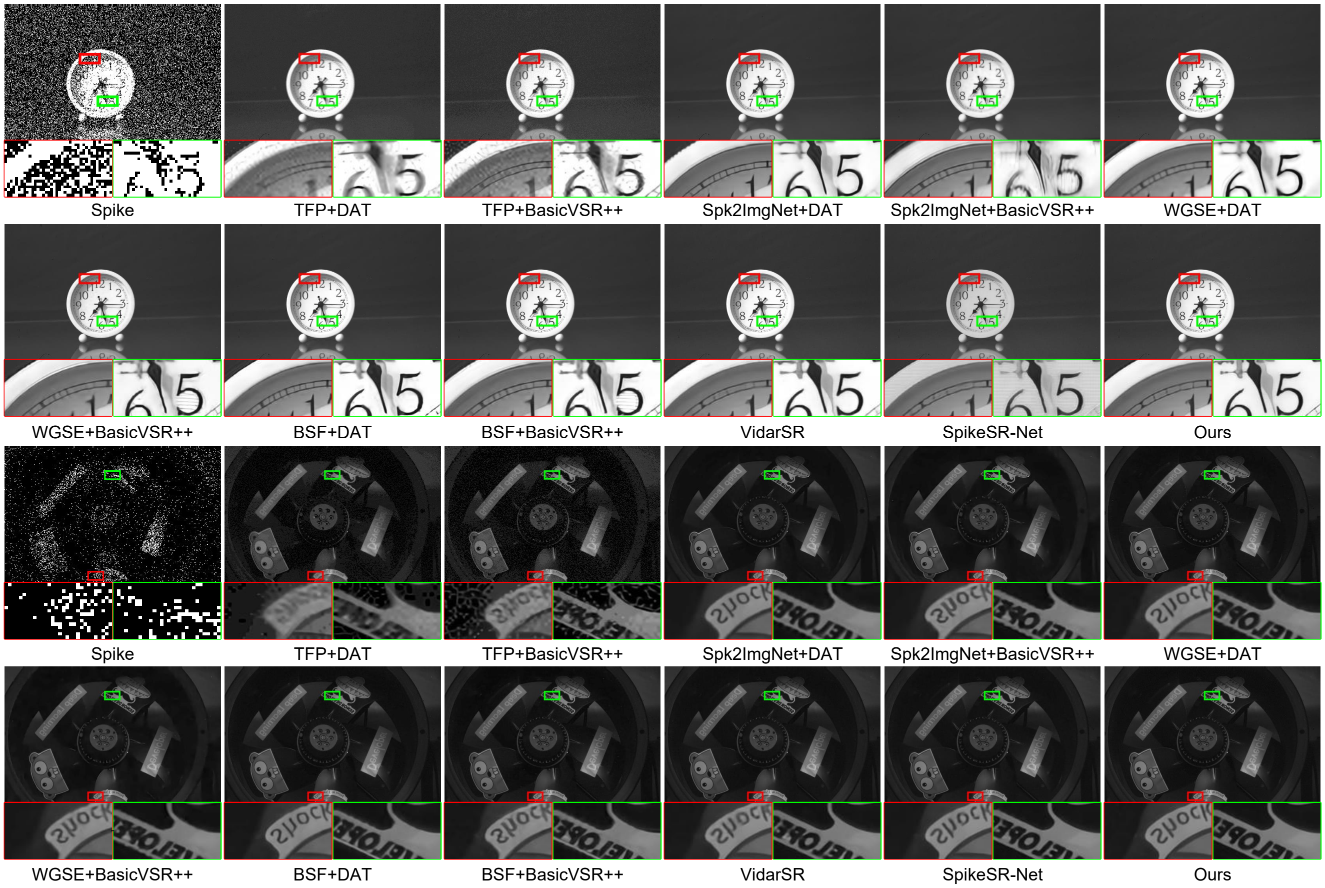

Figure 9. Visual comparison ($\times 4$) on real-world spike data. The first scene is a stationary clock recorded by a fast-moving spike camera. The second scene is a rotating fan. Please enlarge the figure for better comparison.

image / video SR method: DAT [5] / BasicVSR++ [4], resulting in another eight comparison methods, as presented in Tab. 1. We conduct experiments on $\times 2$, $\times 3$, and $\times 4$ SR scales. We employ PSNR, SSIM [39] and LPIPS [46] as quantitative metrics. All compared methods except TFP [53] are retrained on the REDS-based SCSR training dataset with the same settings as ours. As shown in Tab. 1, our method achieves the best performance on the three metrics. Besides, we conduct experiments on real-world captured spike stream, as shown in Fig. 9, our method can restore better textures and details. More evaluation and visual results are provided in supplementary material.

6.4. Ablation Study

To verify the effectiveness of our methodology, we perform ablation study with REDS-based evaluation dataset on $\times 4$ SR scale, as presented in Tab. 2. Since ablation study only involves steps 3, 4, and 5 in MACF, the remaining steps are omitted in Tab. 2 (Steps 2 and 6 do not involve network parameters, and step 1 is present only when step 3 and/or step 5 are present). Case (I) is the final model. Case (A) is formed by removing SIR and MACF modules from Spk2SRImgNet. Comparisons between (A) and (B), (H) and (I) illustrate the advantages of converting information from spike domain to image domain (SIR module). Comparisons between (A) and (H), (B) and (I) demonstrate the effectiveness of MACF module, showing gains of 0.39dB and 0.35dB, respectively. Comparisons between (C-G) and (H) validate the effectiveness of each step in MACF. More ablation study is provided in supplementary material.

7. Conclusion

This paper develops an end-to-end neural network named Spk2SRImgNet to reconstruct high resolution images from low resolution spike stream. To handle spike fluctuations, we propose a motion aligned collaborative filtering (MACF) module motivated by key ideas in classic image restoration schemes. Specifically, MACF leverages the temporal similarity of spike stream to enhance the consistency of multi-moment features along the motion trajectory, mitigating fluctuations and improving the quality of super resolution reconstruction. Experimental results demonstrate that the proposed Spk2SRImgNet achieves state-of-the-art performance on synthesized and real-captured spike data.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 62072009 and U22B2035.

References

[1] Patrick Bardow, Andrew J. Davison, and Stefan Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 884–892, 2016. 2

[2] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014. 1

[3] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2848–2857, 2017. 2, 6

[4] Kelvin C.K. Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5972–5981, 2022. 2, 7, 8

[5] Zheng Chen, Yulun Zhang, Jinjin Gu, Linghe Kong, Xiaokang Yang, and Fisher Yu. Dual aggregation transformer for image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12312–12321, 2023. 8

[6] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007. 2, 3

[7] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 764–773, 2017. 2, 6

[8] Siwei Dong, Tiejun Huang, and Yonghong Tian. Spike camera and its coding methods. In *2017 Data Compression Conference (DCC)*, pages 437–437, 2017. 1

[9] Siwei Dong, Lin Zhu, Daoyuan Xu, Yonghong Tian, and Tiejun Huang. An efficient coding method for spike camera using inter-spike intervals. In *2019 Data Compression Conference (DCC)*, pages 568–568, 2019. 1

[10] Yanchen Dong, Jing Zhao, Ruiqin Xiong, and Tiejun Huang. 3D residual interpolation for spike camera demosaicing. In *IEEE International Conference on Image Processing (ICIP)*, pages 1461–1465. IEEE, 2022. 2

[11] Yanchen Dong, Ruiqin Xiong, Jian Zhang, Zhaofei Yu, Xiaopeng Fan, Shuyuan Zhu, and Tiejun Huang. Super-resolution reconstruction from bayer-pattern spike streams. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24871–24880, 2024. 2

[12] Yanchen Dong, Ruiqin Xiong, Jing Zhao, Jian Zhang, Xiaopeng Fan, Shuyuan Zhu, and Tiejun Huang. Joint demosaicing and denoising for spike camera. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1582–1590, 2024. 2

[13] Yanchen Dong, Ruiqin Xiong, Jing Zhao, Jian Zhang, Xiaopeng Fan, Shuyuan Zhu, and Tiejun Huang. Learning a deep demosaicing network for spike camera with color filter array. *IEEE Transactions on Image Processing*, 33:3634–3647, 2024.

[14] Yanchen Dong, Ruiqin Xiong, Xiaopeng Fan, Shuyuan Zhu, Jin Wang, and Tiejun Huang. Dynamic scene reconstruction for color spike camera via zero-shot learning. *IEEE Transactions on Computational Imaging*, 11:129–141, 2025. 2

[15] Daniel Gehrig and Davide Scaramuzza. Low-latency automotive vision with event cameras. *Nature*, 629(8014):1034–1040, 2024. 1

[16] Arren Glover, Valentina Vasco, and Chiara Bartolozzi. A controlled-delay event camera framework for on-line robotics. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2178–2183. IEEE, 2018. 1

[17] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of field robotics*, 37(3):362–386, 2020. 1

[18] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2862–2869, 2014. 2, 3, 4

[19] Tiejun Huang, Yajing Zheng, Zhaofei Yu, Rui Chen, Yuan Li, Ruiqin Xiong, Lei Ma, Junwei Zhao, Siwei Dong, Lin Zhu, Jianing Li, Shanshan Jia, Yihua Fu, Boxin Shi, Si Wu, and Yonghong Tian. 1000× faster camera and machine vision with ordinary devices. *Engineering*, 25:110–119, 2023. 1

[20] Arpit Jadon, Mohd Omama, Akshay Varshney, Mohammad Samar Ansari, and Rishabh Sharma. Firenet: A specialized lightweight fire & smoke detection model for real-time iot applications. *arXiv preprint arXiv:1905.11922*, 2019. 2

[21] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K. Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging*, 2(2):109–122, 2016. 2

[22] Hanme Kim, Ankur Handa, Ryad Benosman, Sio-Hoi Ieng, and Andrew Davison. Simultaneous mosaicing and tracking with an event camera. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014. 2

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[24] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. Vrt: A video restoration transformer. *IEEE Transactions on Image Processing*, 33:2171–2182, 2024. 2

[25] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128×128 120 db 15 μs latency asynchronous temporal con-

trast vision sensor. *IEEE journal of solid-state circuits*, 43 (2):566–576, 2008. 1

[26] Martin Litzenberger, Christoph Posch, D Bauer, Ahmed Nabil Belbachir, P Schon, B Kohn, and H Garn. Embedded vision system for real-time object tracking using an asynchronous transient vision sensor. In *2006 IEEE 12th Digital Signal Processing Workshop & 4th IEEE Signal Processing Education Workshop*, pages 173–178. IEEE, 2006. 1

[27] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2272–2279, 2009. 2, 3, 4

[28] Mohammad Mostafavi, Yeongwoo Nam, Jonghyun Choi, and Kuk-Jin Yoon. E2sri: Learning to super-resolve intensity images from events. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6890–6909, 2022. 2

[29] Elias Mueggler. *Event-based vision for high-speed robotics*. PhD thesis, University of Zurich, 2017. 1

[30] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 7

[31] Christoph Posch, Daniel Matolin, and Rainer Wohlgenannt. A qvga 143 db dynamic range frame-free pwm image sensor with lossless pixel-level video compression and time-domain cds. *IEEE Journal of Solid-State Circuits*, 46(1):259–275, 2010. 1

[32] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3857–3866, 2019. 2

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 6

[34] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1279–1288, 2017. 7

[35] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. Tdan: Temporally-deformable alignment network for video super-resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[36] Lin Wang, Tae-Kyun Kim, and Kuk-Jin Yoon. Eventsr: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8312–8322, 2020. 2

[37] Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1954–1963, 2019. 2

[38] Yuanlin Wang, Ruiqin Xiong, Jing Zhao, and Tiejun Huang. Reconstruct dynamic scene for spike camera based on 3d space time similarity. In *2024 IEEE International Conference on Image Processing (ICIP)*, pages 1595–1601, 2024. 2

[39] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 8

[40] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1385–1392, 2013. 2

[41] Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Event-based video reconstruction using transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[42] Xijie Xiang, Lin Zhu, Jianing Li, Yixuan Wang, Tiejun Huang, and Yonghong Tian. Learning super-resolution reconstruction for high temporal resolution spike stream. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(1):16–29, 2023. 1, 2, 7

[43] Ruiqin Xiong, Hangfan Liu, Xinfeng Zhang, Jian Zhang, Siwei Ma, Feng Wu, and Wen Gao. Image denoising via bandwise adaptive modeling and regularization exploiting nonlocal similarity. *IEEE Transactions on Image Processing*, 25 (12):5793–5805, 2016. 2, 3, 4

[44] Gang Xu, Jun Xu, Zhen Li, Liang Wang, Xing Sun, and Ming-Ming Cheng. Temporal modulation network for controllable space-time video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6388–6397, 2021. 7

[45] Jiyuan Zhang, Shanshan Jia, Zhaofei Yu, and Tiejun Huang. Learning temporal-ordered representation for spike streams based on discrete wavelet transforms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 137–147, 2023. 7

[46] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–595, 2018. 8

[47] Jing Zhao, Jiyu Xie, Ruiqin Xiong, Jian Zhang, Zhaofei Yu, and Tiejun Huang. Super resolve dynamic scene from continuous spike streams. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2533–2542, 2021. 2

[48] Jing Zhao, Ruiqin Xiong, Hangfan Liu, Jian Zhang, and Tiejun Huang. Spk2imgnet: Learning to reconstruct dynamic scene from continuous spike stream. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11996–12005, 2021. 2, 6, 7

[49] Jing Zhao, Ruiqin Xiong, Jiyu Xie, Boxin Shi, Zhaofei Yu, Wen Gao, and Tiejun Huang. Reconstructing clear image for high-speed motion scene with a retina-inspired spike camera. *IEEE Transactions on Computational Imaging*, 8:12–27, 2021. 3

[50] Jing Zhao, Ruiqin Xiong, Jian Zhang, Rui Zhao, Hangfan Liu, and Tiejun Huang. Learning to super-resolve dynamic scenes for neuromorphic spike camera. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3579–3587, 2023. 1, 2, 6, 7

[51] Rui Zhao, Ruiqin Xiong, Jian Zhang, Zhaofei Yu, Shuyuan Zhu, Lei Ma, and Tiejun Huang. Spike camera image reconstruction using deep spiking neural networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(6): 5207–5212, 2024. 2

[52] Rui Zhao, Ruiqin Xiong, Jing Zhao, Jian Zhang, Xiaopeng Fan, Zhaofei Yu, and Tiejun Huang. Boosting spike camera image reconstruction from a perspective of dealing with spike fluctuations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24955–24965, 2024. 2, 7

[53] Lin Zhu, Siwei Dong, Tiejun Huang, and Yonghong Tian. A retina-inspired sampling method for visual texture reconstruction. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1432–1437. IEEE, 2019. 2, 3, 4, 7, 8

[54] Lin Zhu, Siwei Dong, Jianing Li, Tiejun Huang, and Yonghong Tian. Ultra-high temporal resolution visual reconstruction from a fovea-like spike camera via spiking neuron model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):1233–1249, 2022. 2

[55] Lin Zhu, Xiao Wang, Yi Chang, Jianing Li, Tiejun Huang, and Yonghong Tian. Event-based video reconstruction via potential-assisted spiking neural network. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3584–3594, 2022. 2

[56] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9308–9316, 2019. 2, 6