Learning Temporal-Ordered Representation for Spike Streams Based on Discrete Wavelet Transforms

Jiyuan Zhang[*1], **Shanshan Jia**[*1, 2], **Zhaofei Yu**[†1, 2], **Tiejun Huang**[1, 2]

[1]School of Computer Science, Peking University
[2]Institute for Artificial Intelligence, Peking University
{jyzhang,jiashsh}@stu.pku.edu.cn, {yuzf12, tjhuang}@pku.edu.cn

Abstract

Spike camera, a new type of neuromorphic visual sensor that imitates the sampling mechanism of the primate fovea, can capture photons and output 40000 Hz binary spike streams. Benefiting from the asynchronous sampling mechanism, the spike camera can record fast-moving objects and clear images can be recovered from the spike stream at any specified timestamps without motion blurring. Despite these, due to the dense time sequence information of the discrete spike stream, it is not easy to directly apply the existing algorithms of traditional cameras to the spike camera. Therefore, it is necessary and interesting to explore a universally effective representation of dense spike streams to better fit various network architectures. In this paper, we propose to mine temporal-robust features of spikes in time-frequency space with wavelet transforms. We present a novel Wavelet-Guided Spike Enhancing (WGSE) paradigm consisting of three consecutive steps: multi-level wavelet transform, CNN-based learnable module, and inverse wavelet transform. With the assistance of WGSE, the new streaming representation of spikes can be learned. We demonstrate the effectiveness of WGSE on two downstream tasks, achieving state-of-the-art performance on the image reconstruction task and getting considerable performance on semantic segmentation. Furthermore, We build a new spike-based synthesized dataset for semantic segmentation. Code and Datasets are available at https://github.com/Leozhangjiyuan/WGSE-SpikeCamera.

Introduction

Inspired by the sampling mechanism of the fovea retina, a new type of neuromorphic sensor named spike camera (Huang et al. 2022; Dong, Huang, and Tian 2017) has been developed and achieves a sampling rate of 40000 Hz. Different from the event camera (Gallego et al. 2020; Chen et al. 2011; Brandli et al. 2014; Lichtsteiner, Posch, and Delbruck 2008; Moeys et al. 2018; Posch, Matolin, and Wohlgenannt 2011; Huang, Guo, and Chen 2017) that records the relative light intensity difference, the spike camera encodes the absolute light intensity information. The asynchronous firing mechanism of the spike camera brings the distinctive properties of high-speed sampling, low energy consumption, and high dynamic range. In contrast, the traditional camera collects dozens of images per second, inevitably having a time-domain blind area.

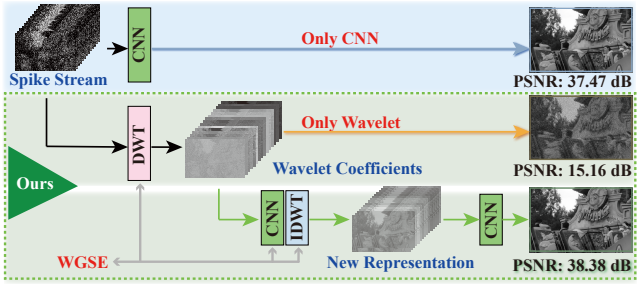

Figure 1: Three technical routes on image reconstruction. Top line: The method using CNN-based model (Zhao et al. 2021); Middle line: The method using integrated features of discrete wavelet transform (DWT) on spikes (Ours). Bottom line: The method using WGSE module. The WGSE firstly performs DWT on spikes, learns effective wavelet components combining with CNN, and outputs the new representation for spikes after inverse DWT. Finally, the downstream CNN recovers the image (Ours).

Many researchers have begun to develop computer vision algorithms suitable for the spike camera, including image reconstruction (Zhu et al. 2020, 2021; Zheng et al. 2021; Nie et al. 2020; Zhao, Xiong, and Huang 2020; Zhao et al. 2021, 2022a; She and Qing 2022), denoising (Xu et al. 2020; Chen et al. 2022), detection (Li et al. 2022a), tracking and recognition of high-speed moving objects (Huang et al. 2022; Zhao et al. 2022b), depth estimation and optical flow estimation (Hu et al. 2022a). Despite these, due to the dense time sequence information and discrete data of the spike camera, it is not easy to directly apply the existing algorithms of traditional cameras to the spike camera. Therefore, mining a general and universal representation for dense spike streams that contain temporal-ordered sequential information and more explicit features of spike data, so that it can properly apply the existing ecology, has become an exciting and necessary exploration direction.

When receiving external stimuli, the photoreceptor accumulates photon energy and converts them into the spike

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

stream. Its essence is that the spike firing rate reflects the light intensity. Thus, the time-frequency information displayed by the spike sequence in temporal order is a good representation of the stimulation information of the external scene. As the current spectral analysis tool, wavelet transform can not only investigate the frequency-domain characteristics of local time-domain processes, but also the time-domain characteristics of local frequency-domain processes (Strang and Nguyen 1996). The original signal is decomposed by a multi-scale wavelet transform to obtain wavelet coefficients. In turn, the wavelet coefficients can also be used to reconstruct the original signal without difference and redundancy. Previous studies have shown that using non-redundant wavelets can well characterize high-dimensional data (Jia et al. 2022).

For example, when completing the reconstruction task, the spike stream gets wavelet coefficients of different levels through wavelet transform, and the multi-scale time-frequency information contains all the contents conducive to reconstruction. However, relying only on the time-frequency information in the wavelet coefficients, and integrating the rough manually selected features based on experience, will make the manual features hard to adapt to new tasks and datasets (shown in Fig. 1).

Recent achievements in deep learning have led to renewed interest among researchers using convolutional neural networks (CNNs) to investigate topics in computer vision. Therefore, based on the natural time-frequency information carriers such as wavelet coefficients and the strong integration ability of CNN, we propose a lightweight module called wavelet-guided spike enhancing (WGSE). The original spike stream is integrated and enhanced by WGSE, and the output of the new data stream can be directly used for various downstream tasks. WGSE is divided into three steps: the first part is to obtain multi-level wavelet coefficients from spikes through discrete wavelet transform (DWT); secondly, further integrate the wavelet coefficients using a CNN module; Thirdly, the learned wavelet coefficients are reconstructed into a new streaming representation by inverse DWT. We connect it with various networks designed for various visual tasks to conduct end-to-end training. Experimental results demonstrate better representations being learned by WGSE which improves performance on two visual tasks. Our contributions can be summarized as followings:

- We first propose a robust representation for spikes by data-driven method, preserving rich temporal information and efficient feature for spike-based visual tasks.
- We make the first attempt at studying spike streams in the time-frequency space by DWT and propose a novel wavelet-guided spike enhancing module that learns to augment wavelet coefficients of spikes.
- We demonstrate the effectiveness of our module for two downstream tasks. Firstly, on image reconstruction, our method achieves state-of-the-art performance. Secondly, we explore semantic segmentation on spikes for the first time and get considerable performance.
- We propose a new synthetic dataset 'Spike-Cityscapes' for semantic segmentation based on the spike streams, which may facilitate future studies on the spike camera.

Related Works

Advances on Visual Tasks Using Spike Camera

Attributed to the special sampling mechanism, not only can spike camera record scenes with very-fast moving objects, it can reconstruct clear images at any specified timestamps. It is the plenty of spatial-temporal information implicitly contained in the spike streams that counts much.

In recent years, research on the spike camera has made great progress. Most of them studied on solve common visual tasks. We approximately divided them into two directions. Firstly, image reconstruction from spikes, as the most basic and initial task, has been explored in depth. Dong et al. (2017) recover the intensity simply by counting the number of spikes with a time window or directly deducing from the interval between adjacent spikes. Zheng et al. (2021) develop a high-speed reconstruction approach through the short-term plasticity (STP) mechanism of the brain (Tsodyks and Markram 1997; Tsodyks, Pawelzik, and Markram 1998). Zhu et al. (2020) propose a reconstruction framework using spiking neural networks, while Zhao et al. (2021) propose the first deep learning model which achieves state-of-the-art performance. Hereafter, She et al. (2022) introduce Transformer architecture to implement image reconstruction. Secondly, solving various downstream visual tasks from spikes. Hu et al. (2022a) take the first step on spike-based optical flow estimation, and Li et al. (2022a) explore object detection based on spikes.

In summary, many methods are learning-based and always try to figure out the common question that how to mine more effective features carrying strong spatial-temporal correlations from the spike streams.

Wavelet Transforms

Wavelet transform can be expressed as a group of multi-scale high-pass filters and low-pass filters. First, different filters are used to find the frequency information contained in the signal, and then the filter slides on the original signal to realize the positioning of its frequency information. Therefore, it has become a multi-resolution analysis tool widely used in signal processing, digital image processing, and image or video compression (Sakar et al. 2019; Chen et al. 2018; Pan et al. 2022; Liu et al. 2020; Suzuki 2020). Some neuroscience researchers have shown that wavelet decomposition and information theory can be combined to encode neural signals such as the spike stream, calcium signal, and EEG signal (Lopes-dos Santos et al. 2015, 2018; Jia et al. 2022). It is found that the non-redundant wavelet can well characterize the neural response of a single neuron or even the population in a certain state.

Combining CNN with wavelet transform has attracted more attention in recent years, which benefits many low-level visual tasks. Liu et al. (2018) utilize DWT to balance receptive field size and computational efficiency and achieve impressive performance on image restoration tasks. In addition, many works make progress on super-resolution (SR) (Huang et al. 2017; Li et al. 2022b). Yang et al. (2020)

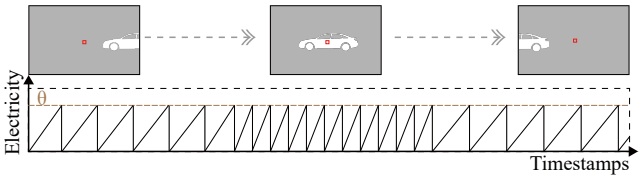

Figure 2: Illustration of how the spike camera firing spikes when light intensity changing caused by a moving object.

enhance the performance of image deraining by operating channel attention after wavelet transform. Li et al. (2022) segments images by applying DWT and learning better low-frequency components by CNN. Chen et al. (2018) developed the wavelet-like auto-encoder, providing a general method for neural network acceleration. Most of these methods aim at static images and solve low-level tasks. We make the first attempt to deal with the spike streams by combining 1-D DWT/IDWT and temporal convolutional neural networks (CNN) with learning. We regard CNN as a selector and enhancer for wavelet coefficients of such dense and irregular 0-1 spike streams, whose results may benefit various downstream tasks for spike cameras.

Preliminaries: Mechanisms of Spike Camera

The spike camera senses the arrival of the photon through its array of special units. Every units that corresponds to each pixel on the imaging plane asynchronously integrates the intensity of photons. Specifically, each unit has three components: the photoreceptor always receives photons, the accumulator converts the intensity of photons into electricity and integrates the voltage, and the comparator determines if the voltage achieves the preset threshold Θ. When the recorded voltage of a unit reaches the pre-defined threshold Θ, a spike is triggered and the voltage will be reset to 0. The mechanism can be formulated as:

$$\int_0^{t_i} \alpha I(t)dt = i \cdot \Theta, i = 1, 2, 3, \ldots, \tag{1}$$

where $I(t)$ describes the intensity of a photon, t_i denotes the firing times of the i-th spikes, and α is the preset photoelectric conversion rate. The backend circuit synchronously reads out discrete signals S with a constant interval δt = 25us. For the pixel at (x, y) at the readout time $T_n = n \cdot \delta t$ $(n = 1, 2, ...)$, $S(x, y, n) = 1$ if a spike fired at (x, y) at time t that $T_n - \delta t < t \leq T_n$, else $S(x, y, n) = 0$. Thus for an interval with T times readout, the output binary spike streams S is in size of $H \times W \times T$, where H, W is the spatial resolution. Fig. 2 illustrates an example of when a car is driving through the scene. The background scene is darker than the surface of the car. Intuitively, the more frequent spikes triggered, the higher the brightness, which can be seen from the red dot in the figure.

Methods

Our method aims to learn a better representation of spikes that effectively assists various visual tasks. Specifically, we first decompose the spike stream with the DWT in the temporal domain into multi-level high-pass components and one low-pass component. Secondly, we build a compact, tiny but efficient module, temporal residual CNN (TRCNN), which deals with the original components and output new components through a learning process. In the end, we recover the new stream $S^{'}$ with the same size of $H \times W \times T$ by inverse wavelet transform. We name the above model as wavelet-guided spike enhancing (WGSE) module M_{WGSE}, which outputs the new representation for spikes. The overview of the proposed model is shown in Fig. 3. To train and validate the scheme, we choose image reconstruction as our first visual task. Besides, we also validate our proposed scheme on semantic segmentation on the spike streams which is also an essential task for parsing scenes using spike camera and has not been explored.

Discrete Wavelet Transform on Spike Streams

Wavelet transform is a local transform in time-frequency space, which performs multi-scale analysis of signals through scaling and translation operations. In the process of wavelet decomposition, the wavelet bases used are finite in length and energy concentrated, which can obtain various frequencies contained in the signal and also locate the specific location in the time domain. In general, 1-D DWT will use a group of decomposition filters D_L, D_H and reconstruction filters R_H, R_L to complete the decomposition and reconstruction of signals.

For the spike data S, the spike stream fired by the photoreceptor in row i and column j is denoted as s_{ij}. The specific process of decomposition is to use a low-pass filter D_L and a high-pass filter D_H to convolute and downsample the signal s_{ij}, so as to obtain low-frequency wavelet coefficients s_{l1} and high-frequency wavelet coefficients s_{h1}. The data $s_{ij}^{'}$ can be reconstructed by using the reconstruction filter to operate the up-sampled wavelet coefficients. The process can be formulated as followings:

$$s_{l1} = (2\downarrow)(D_L * s_{ij}),\ s_{h1} = (2\downarrow)(D_H * s_{ij}), \tag{2}$$

$$s_{ij}^{'} = R_L * (2\uparrow)s_{l1} + R_H * (2\uparrow)s_{h1}, \tag{3}$$

where, $(2\downarrow)$ and $(2\uparrow)$ denote downsampling and upsampling, respectively. $*$ represents a convolution operation. Arrange the low-frequency wavelet coefficients decomposed by the spike stream of all photoreceptors together to obtain the low-frequency coefficient matrix F_{L1}. Similarly, the high-frequency coefficient matrix F_{H1} can be obtained. The above are the details of the wavelet transform of the first level. In the process of multi-level wavelet decomposition, we will continue to use the decomposition filters to further decompose the low-frequency coefficients of the previous level. After the decomposition of 5 layers, finally the wavelet coefficients $F = \{F_{H1}, F_{H2}, F_{H3}, F_{H4}, F_{H5}, F_{L5}\}$ are obtained.

We use the Daubechies(db) wavelet because it is an orthogonal wavelet and can be easily realized by the fast wavelet transform. In addition, the Daubechies wavelet is often used in signal compression and denoising, so the wavelet coefficients can not only represent the spike data but also

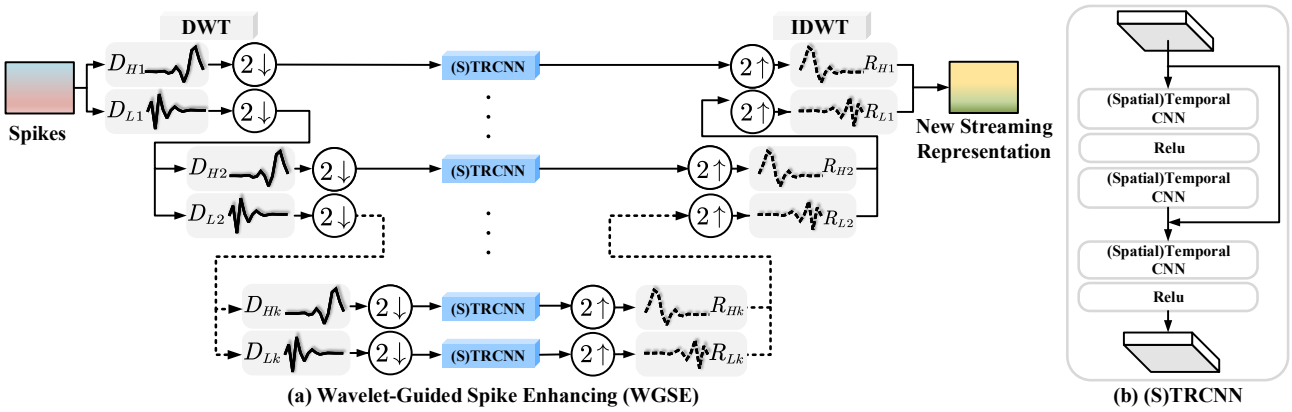

Figure 3: Architecture overview of Wavelet-Guided Spike Enhancing module. $\{D_{H1}, D_{H2}, ..., D_{Hk}\}$ represents high-pass filters while $\{D_{L1}, D_{L2}, ..., D_{Lk}\}$ are low-pass filters. $2\downarrow$ denotes $2\times$ downsampling and $2\uparrow$ denotes $2\times$ upsampling.

may suppress the noise in the spike train. We adopt 'db8'('8' denotes the dimension of the wavelet) as our wavelet in implementation.

Wavelet-Guided Spike Enhancing Module

After DWT on spike steams along the temporal axis, we aim to learn more robust features from the wavelet coefficients in time-frequency space or, in other words, adjust coefficients separately on different high and low-frequency passes. With the wavelet coefficients F after decomposing spikes S by filter bank $\{D_{H1}, D_{H2}, D_{H3}, D_{H4}, D_{H5}, D_{L5}\}$, we use a CNN-based feature extraction module $f($ as shown in Fig. 3(b)) dealing with the coefficients.

We design the module from the perspective of simplicity, lightweight, and effectiveness. It simply consists of several convolutional layers and activation layers attached to a skip connection. For each F_i in F, with the size of $H \times W \times C_i$ where C_i is the number of coefficients channels which is time-ordered, we get $F_i^{'} = f(F_i : \theta)$ with θ denoting the parameter to be optimized in the module f. Specifically, each F_i firstly inputs to two consecutive temporal CNN layers with a ReLU layer in the middle, getting the residual feature F_i^{res} and output F_i^{mid} after adding to F_i. The module finally outputs $F_i^{'}$ after inputting F_i^{mid} through a temporal CNN layer with a ReLU. The computational process can be formulated as:

$$F_i^{mid} = ReLU(F_i * W_{conv1}) * W_{conv2} + F_i, \quad (4)$$

$$F_i^{'} = ReLU(F_i^{mid} * W_{conv3}), \quad (5)$$

where $*$ denotes the convolution operation and W_{conv1}, W_{conv2}, W_{conv3} are weight matrix in three convolutional layers. It is worth mentioning that the $\otimes$ can be done with *Conv1d* or with *Conv2d*, *Conv3d*. We prefer *Conv1d* as our standard implementation because it is very lightweight and with considerably good results. It also keeps the time order of signals, which is an important factor for good performance. Experiments in ablation studies give a detailed analysis of three implementations.

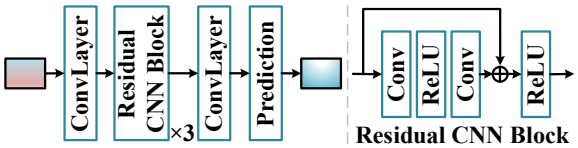

Figure 4: Downstream network for image reconstruction.

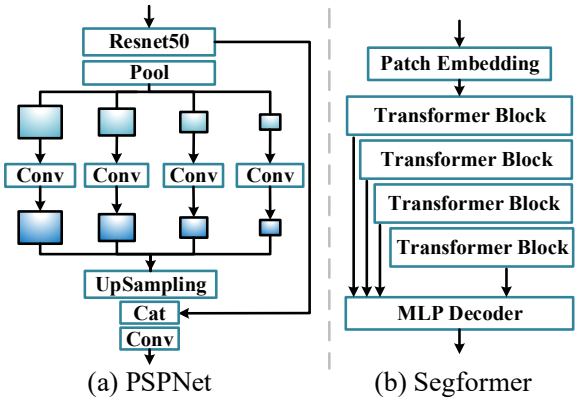

Figure 5: Sketches of PSPNet(Zhao et al. 2017) and Segformer(Xie et al. 2021) for semantic segementation.

After the above steps, new learned wavelet coefficients $F^{'} = \{F_{H1}^{'}, F_{H2}^{'}, F_{H3}^{'}, F_{H4}^{'}, F_{H5}^{'}, F_{L5}^{'}\}$ are obtained and the new stream S' with the same size as S is generated through IDWT. Empirically, we regard the CNN-based module as a rectifier and selector for wavelet coefficients. For different frequency channels, CNNs are capable of mining or enhancing valid coefficients that are beneficial for tasks and suppressing the ones helpless.

Architectures for Downstream Tasks

In the WGSE, one piece of spike stream S goes through wavelet transform, wavelet coefficients are learned by the TRCNN module, and the new stream S' is obtained through

inverse wavelet transform. We denote the above process as $S' = f_{M_{\text{WGSE}}}(S : \theta_{M_{\text{WGSE}}})$. Due to the same variable size between S and S', for networks adopted for downstream tasks, we only need to add such lightweight M_{WGSE} at the head position. We collectively denote downstream networks as M_{task}, the prediction result as $\hat{Y}_{task}$ and the corresponding label as Y^{gt}_{task}. Thus, the whole computing process can be formulated as:

$$\hat{Y}_{task} = f_{M_{task}}(f_{M_{\text{WGSE}}}(S : \theta_{M_{\text{WGSE}}}) : \theta_{M_{task}}), \quad (6)$$

Firstly, we aim to fully explore and demonstrate the effectiveness of the M_{WGSE} on image reconstruction. Thus there is no carefully designed architecture for it. Instead, we simply use sequential CNN layers with a small number of residual blocks, as illustrated in Fig. 4. The prediction layer is one convolutional layer that outputs a reconstructed image with the size of $H \times W \times 1$.

To further validate the efficiency of M_{WGSE}, we choose semantic segmentation (SS), a more complicated visual task. At this step, we pick PSPNet (Zhao et al. 2017) and Segformer (Xie et al. 2021) as structures of the task network M_{seg}, whose sketches are shown in Fig. 5. PSPNet is a typical model for SS which contains an efficient pyramid pooling module and fully convolutional networks (FCN). Segformer is one of the state-of-the-art methods on SS for images that adopts popular Transformer architecture. It consists of hierarchical Transformer blocks for encoder and aggregates information with multilayer perceptron (MLP) decoders. They are typical networks implemented with CNN and MLP architecture. To fit the spike stream S into the network, we change the input channels of the network with T and add the M_{WGSE} as the head of the network.

Loss Functions

Image Reconstruction The recovered gray-scale image $\hat{Y}_{rec}$ and the ground-truth image Y^{gt}_{rec} are in size of $H \times W \times 1$. We simply use L1 loss to optimize the network, which can be formulated as followings where n is the number of valid ground truth pixels:

$$\mathcal{L}_{rec} = \frac{1}{n}\sum_{i=1}^{n} |\hat{Y}_{rec}(i) - Y^{gt}_{rec}(i)|, \quad (7)$$

Semantic Segmentation Networks output matrix $\hat{Y}_{seg}$ represents the probability of class for each pixel. It is in size of $H \times W \times C$, where C denotes the number of semantic classes of the dataset. We calculate the cross entropy loss $\mathcal{L}_{CE}$ with online hard example mining (Ohem) (Shrivastava, Gupta, and Girshick 2016) strategy, which can be formulated as followings,

$$\mathcal{L}_{seg} = \mathcal{L}_{OhemCE} = \frac{1}{n}\sum_{i=1}^{n} \ell_{CE}(i) \cdot \mathbf{1}\{\ell_{CE}(i) > \beta\}, \quad (8)$$

$$\ell_{CE}(i) = \frac{1}{n}\sum_{i=1}^{n}\left(-\sum_{c=1}^{C} w_c \log \frac{\exp(x_{n,c})}{\exp(\sum_{k=1}^{C} x_{n,k})} y_{n,c}\right), \quad (9)$$

where β is a threshold that determines hard examples, $y_{n,c}$ is a binary value that denotes whether the n-th sample belongs to class c and w_c controls weights for each class.

Experiments

Datasets

For image reconstruction on spikes, we adopt a synthesized dataset (Zhao et al. 2021) generated from videos in the REDS dataset (Nah et al. 2019). To test the performance in the real world, we use a real spike dataset (Zhu et al. 2020).

For semantic segmentation on spikes, we build a new synthesized spike-based dataset to train networks, due to no available datasets so far. We adopt Cityscapes (Cordts et al. 2016) as the base dataset. For every video snippet in the Cityscapes, we use a state-of-the-art interpolation method M2M (Hu et al. 2022b) to interpolate intermediate frames and get high frame-rate videos. By simulating the mechanism of generating spikes, with video frames as inputs, we get spike stream S with the size of $H \times W \times T(512 \times 1024 \times 129)$. Thus, the spike version of Cityscapes is built, which is named as 'Spike-Cityscapes'.

Training Details

The whole training process is implemented with Pytorch. We use the Adam optimizer during training and set the initial learning rate to 0.0001. For image reconstruction, we train all networks for 600 epochs, and the learning rate decays to 0.00002 after 400 epochs. Models are trained on 1 NVIDIA-A100(40GB) GPU with the a batch size of 16. Input batches for training are augmented with the random crop of the spatial size 128×128, random vertical flip, and horizontal flip. For semantic segmentation, we train all networks for 500 epochs optimized by the AdamW optimizer. For Segformer and Segformer+WGSE, the initial learning rate is 0.0002 after 10-epoch warmup and the decay rate used in optimizer set to 0.002. For PSPNet and PSPNet+WGSE, the initial learning rate is 0.001 after a 10-epoch warmup, and the decay rate used in the optimizer is set to 0.01. Input batches for training are augmented with the random crop of the spatial size 256×512, random vertical flip, and horizontal flip. Models are trained on 1 NVIDIA-A100(40GB) GPU with a batch size of 4.

Experiment Results

We evaluate the proposed WGSE on image reconstruction (**IR**) and semantic segmentation (**SS**) tasks. For **IR**, we compare our method with the state-of-the-arts on synthesized REDS dataset and the real dataset. For **SS**, we explore the validity of WGSE with PSPNet(Zhao et al. 2017) and Segformer (Xie et al. 2021) as base architectures.

A. Qualitative and Quantitative Comparison of IR We compare our method with the commonly used reconstruction methods, including TFI, TFP (Dong, Huang, and Tian 2017), TVS (Zhu et al. 2020), STP (Zheng et al. 2021), SSML-SCR (Chen et al. 2022) and Spk2ImgNet (Zhao et al. 2021). The Spk2ImgNet is the state-of-the-art one now,

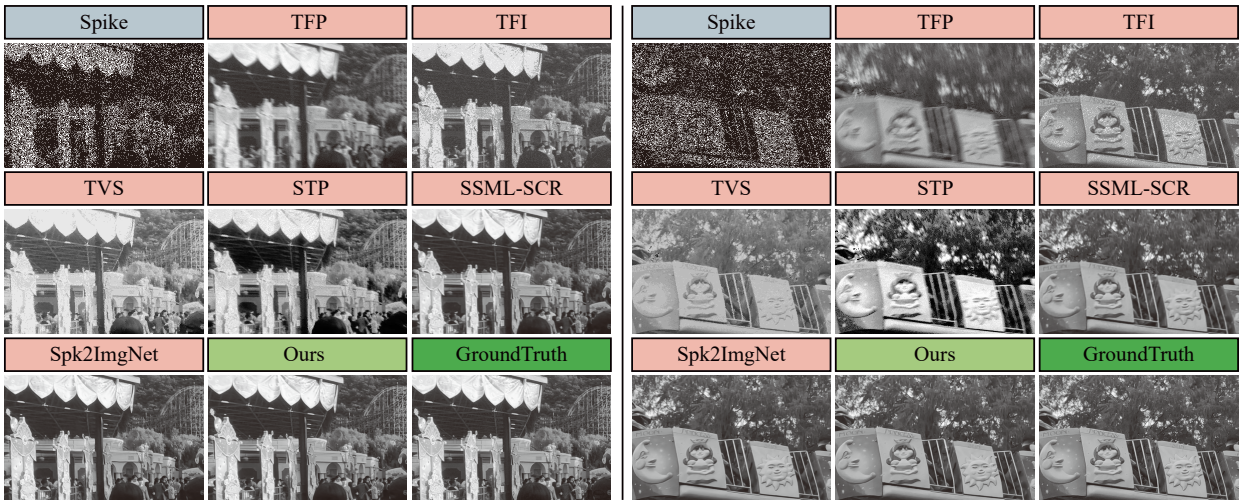

Figure 6: Reconstruction results on validation set compared with other methods.

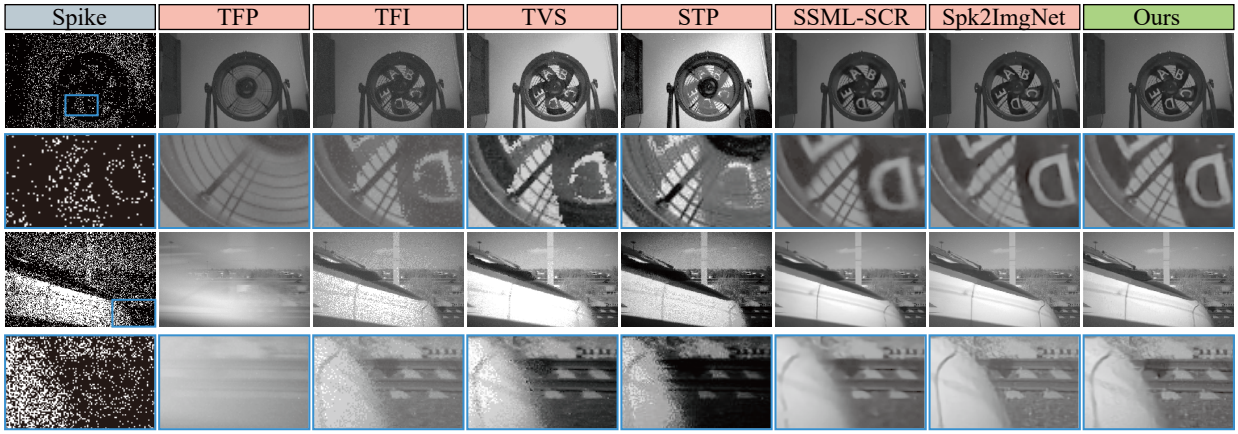

Figure 7: Reconstruction results on real dataset compared with other methods.

Scheme	Model	PSNR ↑	SSIM ↑
Unsup	TFP(2017)	22.37	0.5801
	TFI(2017)	24.94	0.7150
	TVS(2020)	19.03	0.7452
	STP(2021)	22.37	0.7300
	SSML-SCR(2022)	34.26	0.9718
Sup	Spk2ImgNet(2021)	38.44	0.9767
	Ours(WGSE1d)	**38.88**	**0.9774**

Table 1: Quantitative comparison on synthesized REDs dataset. ↑ indicates the higher is better.

which trains a deep CNN architecture in a supervised manner. We adopt the WGSE-1D module and concatenate it with the simple and feedforward structure illustrated in Fig. 4.

Tab. 1 reports the quantitative comparison results. We use peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) as the image quality assessment. As illustrated in Tab. 1, our method outperforms all the comparison methods on both metrics. The PSNR reaches the highest 38.88dB on the synthesized dataset. From the perspective of structure, our method only concatenates WGSE with a simple CNN that consists of three residual blocks and a few convolutional layers. In addition, the WGSE module is composed of only three 1D convolutional layers, and connections are also simple. Therefore we consider our results impressive, and it is the effectiveness of the WGSE that improves the performance. In contrast, Spk2ImgNet has many complicated modules that are very time-consuming. The results demonstrate that the WGSE learns a more efficient representation for spike streams in time-frequency space. Two groups of visualized results are shown in Fig. 6, from which we can see that our method can reconstruct high-quality images from spike streams. The estimated images own fine texture and expected pixel intensity.

B. Evaluation on Real-World Dataset of IR To further ver-

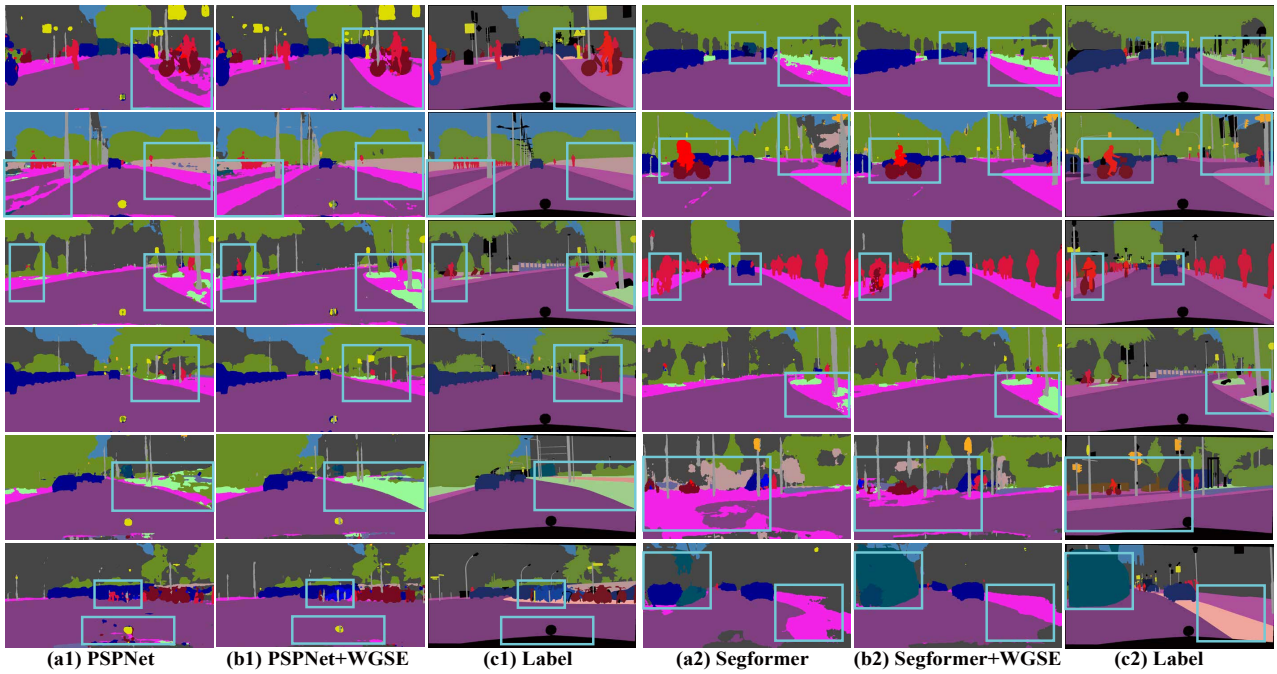

Figure 8: Visualized results on semantic segmentation w/ or w/o WGSE. (a1)(b1)(c1) give comparison with PSPNet as the task network, (a2)(b2)(c2) give comparison with Segformer as the task network. Boxes in cyan emphasize details for comparison.

ify the generalization of our method, we train the network for reconstruction with a synthesized dataset and test on a real spike dataset composed of four challenging sequences named *Fan*, *Train*, *Car*, and *Doll*. We use three metrics to assess the image quality: NIQE(Mittal, Soundararajan, and Bovik 2013), BRISQUE(Mittal, Moorthy, and Bovik 2012), and 2D entropy(Xi, Guosui, and Ni 1999). While the smaller value indicates higher image quality for NIQE and BRISQUE, a higher value of 2D Entropy indicates higher quality because it refers to the information contained in an image. Tab. 1 reports the quantitative results on these metrics, values in bold represent top-2 results while values underlined represent the best ones.

Compared with other methods, our methods achieve all top-2 ranking on three metrics and especially performs best on BRISQUE. It indicates that our method owns good generalization on unseen real datasets. We visualize the sequence **Fan** and **Train** in Fig. 7, from which the first and the three line show full-resolution results. Figures in the second and fourth lines are zoomed patches corresponding to the blue boxes. One can find that the reconstructed images predicted by our method are clearer and more realistic, from which we can find more details of textures and contours. For example, the "ring structure" on the back of the fan is more clearly recovered by our method and the "rail" in the **Train** sequence has more realistic contours with stronger contrast.

C. Comparison Results of SS With spikes as input, instead of redesigning a new network, we use existing networks for RGB images (PSPNet (Zhao et al. 2017) and Segformer(B0) (Xie et al. 2021)). To fit the spike stream into these networks, we simply change their input channel to T. Our method concatenates our WGSE-1D module with the above networks. We also train these two task networks without WGSE-1D for direct comparison. We use mean IOU (mIOU), mean accuracy (mACC), and all accuracy (allAcc) as metrics. Quantitative results are given in Tab. 3. For PSPNet, with WGSE applied, mIOU raises to 0.556 from 0.531 and mAcc raises to 0.633 from 0.602, while for Segformer, mIOU raises to 0.488 from 0.464 and mAcc raises to 0.578 from 0.557. Trainable parameter numbers are given in the table. Downstream networks occupy most of the parameters. The WGSE only holds 0.021M parameters compared to 46.747M (PSPNet) and 3.779M (Segformer). The lightweight WGSE brings remarkable improvements, which demonstrates its importance. With the help of WGSE, the wavelet coefficients of spikes in the time-frequency space are learned to extract valid information for semantics.

Model	NIQE	BRISQUE	2D Entropy
TFP	7.392	19.416	9.394
TFI	10.895	43.350	7.779
TVS	7.449	33.571	10.036
STP	5.673	27.556	**12.928**
SSML-SCR	4.938	28.662	8.816
Spk2ImgNet	**3.843**	**22.278**	9.988
Ours	**4.012**	**19.407**	**10.444**

Table 2: Quantitative comparison on real datasets with no-reference metrics.

Model	mIOU	mAcc	allAcc	Params
(a)PSPNet	0.531	0.602	0.915	46.737M
(a)+WGSE	**0.556**	**0.633**	**0.916**	46.758M
(b)SegFormer	0.464	0.557	0.891	3.779M
(b)+WGSE	**0.488**	**0.578**	**0.898**	3.800M

Table 3: Quantitative comparison on semantic segmentation for models w/ or w/o WGSE.

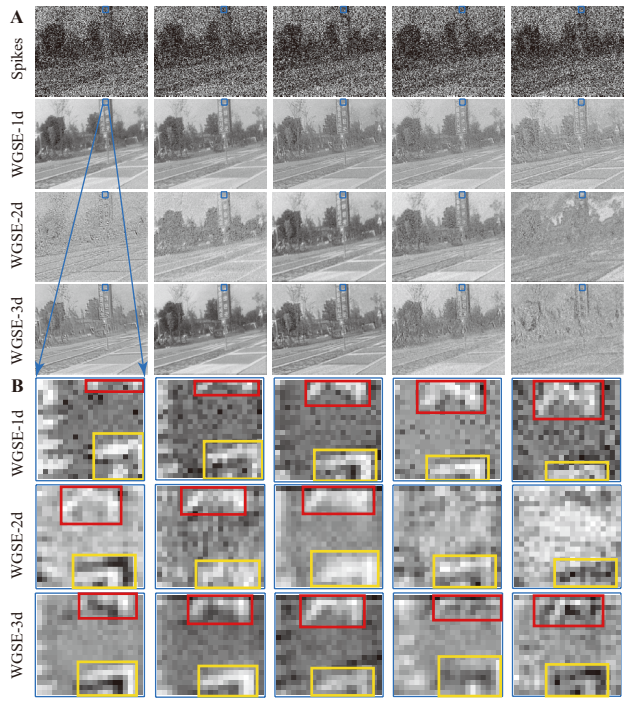

Figure 9: Output frames of WGSE-1d, WGSE-2d and WGSE-3d. Details are highlighted with red and yellow boxes.

Some visualized results are illustrated in Fig. 8 in which boxes in cyan highlight details for comparison. It can be seen that networks applied with the WGSE present higher accuracy on classifying pixels and the results are more smooth from the visualization result. To be specific, from Fig. 8(a1)(b1)(c1), compared to the network applied with WGSE, results from raw PSPNet show coarser classification results and some confusion among 'road', 'sidewalk', 'motorcycle', 'car' and 'rider'. From Fig. 8(a2)(b2)(c2), our method shows more accurate results for discriminating 'car', 'rider', 'sidewalk', 'road', 'bus' and 'building'.

Ablation Studies and Analysis

A. Analysis of Wavelet Coefficients To find detailed differences among WGSE-1d, WGSE-2d, and WGSE-3d, we analyze the output of WGSE. Fig. 9, 10, 11 show the output frames $\{3, 11, 21, 31, 39\}$ on three test samples. Fig. 9A, 10A, 11A show in full resolution, while

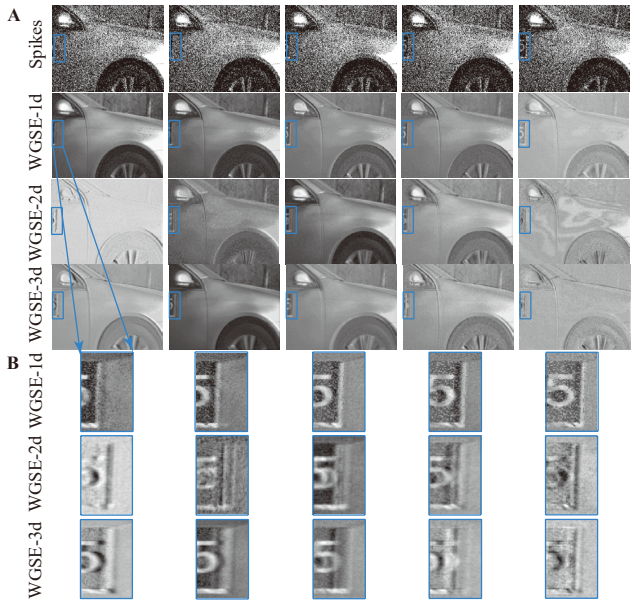

Figure 10: Output frames of WGSE-1d, WGSE-2d and WGSE-3d. The scene shows a high-speed car driving through the view.

Metrics	WGSE-1d	WGSE-2d	WGSE-3d
PSNR	**38.88**	37.904	*38.94*
SSIM	**0.977**	0.973	*0.978*

Table 4: Comparison results on different Conv in the WGSE.

Fig. 9B, 10B, 11B show the enlargement of the area in the blue boxes in A. In WGSE-1d from Fig. 9, the pattern in the red and yellow box smoothly moves from the top right to the bottom left. The red box is gradually larger and the yellow box is gradually smaller. From Fig. 10, the pattern 'Five' on the car maintains better temporal order and clearer contours than the other two. From the pattern 'A' in Fig. 11, it can be seen that the output of WGSE replays the rotation process of the fan with a clear texture. It can be seen that the new stream representation learned by WGSE-1D has preserved the temporality of the original spike stream, while the WGSE-2d has not seen the retention of the temporality. As shown in Tab. 4, the reconstruction performance of 1D and 3D networks is better than that of 2D networks, which proves that the preservation of temporal information has an important advantage for image reconstruction from spike streams.

B. Ablation Study on Different Configurations of the WGSE. We report ablation results for different configurations of the WGSE to further prove the validity of our method. Firstly, we validate whether our scheme of wavelet transform works or the TRCNN structure in the WGSE works. Thus, we take out the TRCNN and simply concatenate it with downstream Resblocks, which make up a new network with original spikes as input. In addition, we adjust the input channels of Resblocks with T and make them

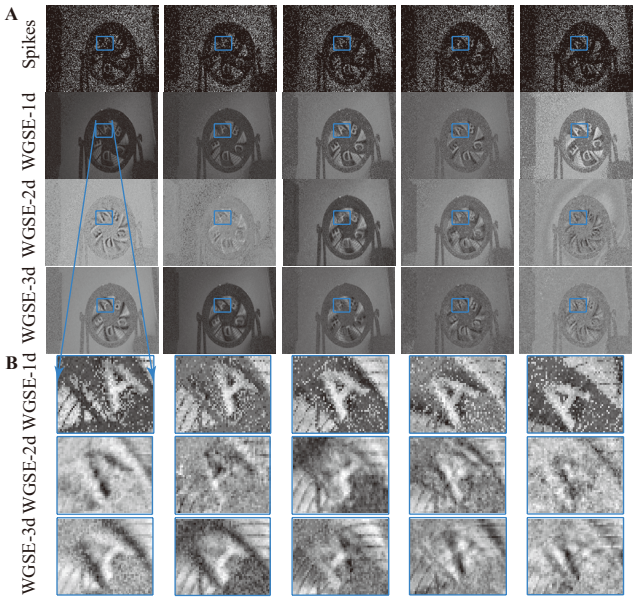

Figure 11: Output frames of WGSE-1d, WGSE-2d and WGSE-3d. The scene shows a rotating fan on which some letters are printed.

Model	PSNR ↑	SSIM ↑
Resblocks	38.20	0.974
TRCNN + Resblocks	38.28	0.975
WGSE-1d(Long) + Resblocks	38.80	0.977
WGSE-1d + Resblocks	**38.88**	**0.977**

Table 5: Results on different configurations of the WGSE.

a simple network. From the first two lines and the last line in Tab. 5, our method with WGSE-1D shows an advantage over the other two, which demonstrates that the WGSE outputs a good representation for the spike stream after learning with wavelet coefficients in the time-frequency space. In the second, we double the number of residual blocks and convolutional layers in TRCNN to get a complicated version of the WGSE. From the last two lines in Tab. 5, the long version of the WGSE performs worse than the proposed one, which proves that it is the proposed learning mechanism with help of the wavelet that works, rather than the parameter number.

C. Robustness for the Hyper-Parameters in the Wavelet Transform The hyper-parameters consist of the dimension of wavelet and decomposition levels. We compare the results using Daubechies(db) wavelets with different dimensions and decomposition levels. Shorter filters or fewer levels lead to poor results because they decompose signals to less time-frequency information. Longer filters or more levels reduce computational efficiency and make the results saturated because for a finite discrete signal with a length of N, the appropriate level of decomposition is log base 2 of N. In addition, the hyper-parameters we used show good results on image reconstruction and segmentation, proving the robustness.

D. Further Analysis Compared With Denoising Algorithms. Our method aims to learn a better representation of spikes for downstream tasks in a data-driven manner, not typically used for denoising. Though, it has a certain denoising effect. To give a thorough comparison and analysis of the proposed WGSE, we adopted denoising filters for spike/DVS data and added them in the front of the network. We tested three filters with experiments on image reconstruction. The first is the mean filter plus the subsequent CNN. The PSNR is 35.801 and the SSIM is 0.962 for a spatial filter, while PSNR is 35.336 and SSIM is 0.958 for a temporal filter. The second is an STP filter for the spike data (Zheng et al. 2021), achieving 26.868 on PSNR and 0.881 on SSIM. The last one is a typical spatial-temporal filter for DVS data (Delbruck 2008), achieving 36.812 on PSNR and 0.964 on SSIM. Our method achieves 38.88 on PSNR and 0.997 on SSIM, which outperforms all these simple approaches. The reason for the better performance of our method is that after the wavelet transform, wavelet coefficients of spikes carry multi-level time-frequency information. Besides, our learning-based manner rectifies or enhances the signals by CNN, which benefits downstream tasks.

Conclusion

We propose a novel WGSE module that operates 1D DWT on the spike stream, learns to augment its wavelet coefficients and outputs new representation by IDWT. The new representation for spikes preserves temporal-ordered and effective information for visual tasks. It is the first attempt to study spikes in the time-frequency space. We demonstrate the effectiveness of the WGSE on two tasks, achieving state-of-the-art performance on the image reconstruction task and getting performance improvement on semantic segmentation. A new synthetic dataset 'Spike-Cityscapes' based on spikes is building, which may facilitate future studies.

Acknowledgements

This work was supported by STI 2030-Major Projects2021ZD0200300 and the National Natural Science Foundation of China under Grant No. 62176003 and No. 62088102.

References

Brandli, C.; Berner, R.; Yang, M.; Liu, S.-C.; and Delbruck, T. 2014. A 240×180 130 dB 3μs Latency Global Shutter Spatiotemporal Vision Sensor. *IEEE Journal of Solid-State Circuits*, 49(10): 2333–2341.

Chen, D. G.; Matolin, D.; Bermak, A.; and Posch, C. 2011. Pulse-Modulation Imaging—Review and Performance Analysis. *IEEE Transactions on Biomedical Circuits and Systems*, 5(1): 64–82.

Chen, S.; Duan, C.; Yu, Z.; Xiong, R.; and Huang, T. 2022. Self-Supervised Mutual Learning for Dynamic Scene Reconstruction of Spiking Camera. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI)*, 2859–2866.

Chen, T.; Lin, L.; Zuo, W.; Luo, X.; and Zhang, L. 2018. Learning a Wavelet-Like Auto-Encoder to Accelerate Deep Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 32.

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3213–3223.

Delbruck, T. 2008. Frame-free dynamic digital vision. In *Proceedings of Intl. Symp. on Secure-Life Electronics, Advanced Electronics for Quality Life and Society*, volume 1, 21–26.

Dong, S.; Huang, T.; and Tian, Y. 2017. Spike Camera and Its Coding Methods. In *Data Compression Conference (DCC)*, 437–437. 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.

Hu, L.; Zhao, R.; Ding, Z.; Ma, L.; Shi, B.; Xiong, R.; and Huang, T. 2022a. Optical Flow Estimation for Spiking Camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 17844–17853.

Hu, P.; Niklaus, S.; Sclaroff, S.; and Saenko, K. 2022b. Many-to-many Splatting for Efficient Video Frame Interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3553–3562.

Huang, H.; He, R.; Sun, Z.; and Tan, T. 2017. Wavelet-SRNet: A Wavelet-Based CNN for Multi-Scale Face Super Resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 1689–1697.

Huang, J.; Guo, M.; and Chen, S. 2017. A Dynamic Vision Sensor with Direct Logarithmic Output and Full-frame Picture-On-Demand. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, 1–4.

Huang, T.; Zheng, Y.; Yu, Z.; Chen, R.; Li, Y.; Xiong, R.; Ma, L.; Zhao, J.; Dong, S.; Zhu, L.; Li, J.; Jia, S.; Fu, Y.; Shi, B.; Wu, S.; and Tian, Y. 2022. 1000× Faster Camera and Machine Vision with Ordinary Devices. *Engineering*.

Jia, S.; Li, X.; Huang, T.; Liu, J. K.; and Yu, Z. 2022. Representing the Dynamics of High-Dimensional Data With Non-Redundant Wavelets. *Patterns*, 3(3): 100424.

Li, J.; Wang, X.; Zhu, L.; Li, J.; Huang, T.; and Tian, Y. 2022a. Retinomorphic Object Detection in Asynchronous Visual Streams. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 1332–1340.

Li, Q.; and Shen, L. 2022. WaveSNet: Wavelet Integrated Deep Networks for Image Segmentation. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, 325–337.

Li, Z.; Kuang, Z.-S.; Zhu, Z.-L.; Wang, H.-P.; and Shao, X.-L. 2022b. Wavelet-Based Texture Reformation Network for Image Super-Resolution. *IEEE Transactions on Image Processing*, 31: 2647–2660.

Lichtsteiner, P.; Posch, C.; and Delbruck, T. 2008. A 128× 128 120 dB 15 μs Latency Asynchronous Temporal Contrast Vision Sensor. *IEEE Journal of Solid-State Circuits*, 43(2): 566–576.

Liu, P.; Zhang, H.; Zhang, K.; Lin, L.; and Zuo, W. 2018. Multi-Level Wavelet-CNN for Image Restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 773–782.

Liu, Q.; Yang, S.; Liu, J.; Xiong, P.; and Zhou, M. 2020. A Discrete Wavelet Transform and Singular Value Decomposition-Based Digital Video Watermark Method. *Applied Mathematical Modelling*, 85: 273–293.

Lopes-dos Santos, V.; Panzeri, S.; Kayser, C.; Diamond, M. E.; and Quian Quiroga, R. 2015. Extracting Information in Spike Time Patterns With Wavelets and Information Theory. *Journal of Neurophysiology*, 113(3): 1015–1033.

Lopes-dos Santos, V.; Rey, H. G.; Navajas, J.; and Quiroga, R. Q. 2018. Extracting Information From the Shape and Spatial Distribution of Evoked Potentials. *Journal of Neuroscience Methods*, 296: 12–22.

Mittal, A.; Moorthy, A. K.; and Bovik, A. C. 2012. No-Reference Image Quality Assessment in the Spatial Domain. *IEEE Transactions on Image Processing*, 21(12): 4695–4708.

Mittal, A.; Soundararajan, R.; and Bovik, A. C. 2013. Making a "Completely Blind" Image Quality Analyzer. *IEEE Signal Processing Letters*, 20(3): 209–212.

Moeys, D. P.; Corradi, F.; Li, C.; Bamford, S. A.; Longinotti, L.; Voigt, F. F.; Berry, S.; Taverni, G.; Helmchen, F.; and Delbruck, T. 2018. A Sensitive Dynamic and Active Pixel Vision Sensor for Color or Neural Imaging Applications. *IEEE Transactions on Biomedical Circuits and Systems*, 12(1): 123–136.

Nah, S.; Baik, S.; Hong, S.; Moon, G.; Son, S.; Timofte, R.; and Lee, K. M. 2019. NTIRE 2019 Challenge on Video Deblurring and Super-Resolution: Dataset and Study. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1996–2005.

Nie, K.; Shi, X.; Cheng, S.; Gao, Z.; and Xu, J. 2020. High Frame Rate Video Reconstruction and Deblurring Based on Dynamic and Active Pixel Vision Image Sensor. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(8): 2938–2952.

Pan, W.; Shi, H.; Zhao, Z.; Zhu, J.; He, X.; Pan, Z.; Gao, L.; Yu, J.; Wu, F.; and Tian, Q. 2022. Wnet: Audio-Guided Video Object Segmentation via Wavelet-Based Cross-Modal Denoising Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1320–1331.

Posch, C.; Matolin, D.; and Wohlgenannt, R. 2011. 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.

Sakar, C. O.; Serbes, G.; Gunduz, A.; Tunc, H. C.; Nizam, H.; Sakar, B. E.; Tutuncu, M.; Aydin, T.; Isenkul, M. E.; and Apaydin, H. 2019. A Comparative Analysis of Speech Signal Processing Algorithms for Parkinson's Disease Classification and the Use of the Tunable Q-Factor Wavelet Transform. *Applied Soft Computing*, 74: 255–263.

She, C.; and Qing, L. 2022. SpikeFormer: Image Reconstruction from the Sequence of Spike Camera Based on Transformer. In *International Conference on Image and Graphics Processing (ICIGP)*, 72–78.

Shrivastava, A.; Gupta, A.; and Girshick, R. 2016. Training Region-Based Object Fetectors With Online Hard Example Mining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 761–769.

Strang, G.; and Nguyen, T. 1996. *Wavelets and Filter Banks*. SIAM.

Suzuki, T. 2020. Wavelet-Based Spectral–Spatial Transforms for CFA-Sampled Raw Camera Image Compression. *IEEE Transactions on Image Processing*, 29: 433–444.

Tsodyks, M.; Pawelzik, K.; and Markram, H. 1998. Neural Networks with Dynamic Synapses. *Neural Computation*, 10(4): 821–835.

Tsodyks, M. V.; and Markram, H. 1997. The Neural Code Between Neocortical Pyramidal Neurons Depends on Neurotransmitter Release Probability. *Proceedings of the National Academy of Sciences*, 94(2): 719–723.

Xi, L.; Guosui, L.; and Ni, J. 1999. Autofocusing of ISAR Images Based on Entropy Minimization. *IEEE Transactions on Aerospace and Electronic Systems*, 35(4): 1240–1252.

Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 12077–12090.

Xu, J.; Xu, L.; Gao, Z.; Lin, P.; and Nie, K. 2020. A Denoising Method Based on Pulse Interval Compensation for High-Speed Spike-Based Image Sensor. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(8): 2966–2980.

Yang, H.-H.; Yang, C.-H. H.; and Wang, Y.-C. F. 2020. Wavelet Channel Attention Module With A Fusion Network For Single Image Deraining. In *IEEE International Conference on Image Processing (ICIP)*, 883–887. IEEE.

Zhao, H.; Shi, J.; Qi, X.; Wang, X.; and Jia, J. 2017. Pyramid Scene Parsing Network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2881–2890.

Zhao, J.; Xiong, R.; and Huang, T. 2020. High-Speed Motion Scene Reconstruction for Spike Camera via Motion Aligned Filtering. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, 1–5. IEEE.

Zhao, J.; Xiong, R.; Liu, H.; Zhang, J.; and Huang, T. 2021. Spk2ImgNet: Learning to Reconstruct Dynamic Scene From Continuous Spike Stream. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11996–12005.

Zhao, J.; Xiong, R.; Xie, J.; Shi, B.; Yu, Z.; Gao, W.; and Huang, T. 2022a. Reconstructing Clear Image for High-Speed Motion Scene With a Retina-Inspired Spike Camera. *IEEE Transactions on Computational Imaging*, 8: 12–27.

Zhao, J.; Yu, Z.; Ma, L.; Ding, Z.; Zhang, S.; Tian, Y.; and Huang, T. 2022b. Modeling The Detection Capability Of High-Speed Spiking Cameras. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4653–4657.

Zheng, Y.; Zheng, L.; Yu, Z.; Shi, B.; Tian, Y.; and Huang, T. 2021. High-Speed Image Reconstruction Through Short-Term Plasticity for Spiking Cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6358–6367.

Zhu, L.; Dong, S.; Li, J.; Huang, T.; and Tian, Y. 2020. Retina-Like Visual Image Reconstruction via Spiking Neural Model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1438–1446.

Zhu, L.; Li, J.; Wang, X.; Huang, T.; and Tian, Y. 2021. NeuSpike-Net: High Speed Video Reconstruction via Bio-Inspired Neuromorphic Cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2380–2389.