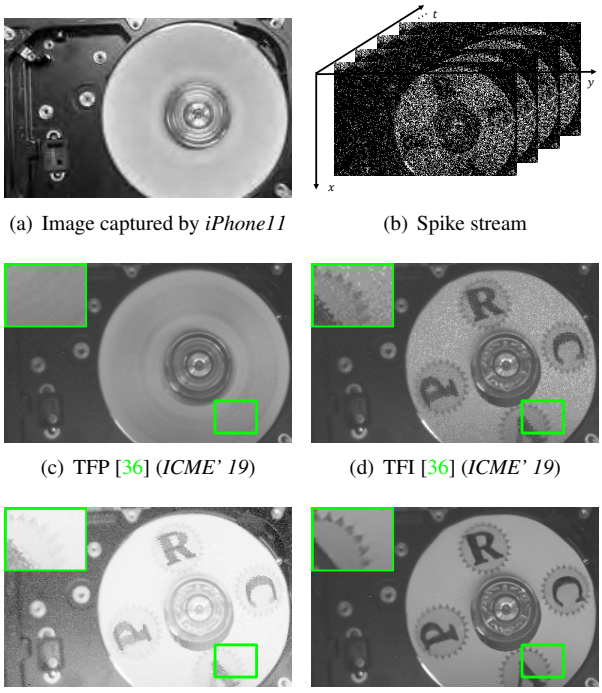
Spk2ImgNet: Learning to Reconstruct Dynamic Scene from Continuous Spike Stream

Jing Zhao[1], Ruiqin Xiong[1], Hangfan Liu[2], Jian Zhang[3], Tiejun Huang[1]
[1]School of Electronic Engineering and Computer Science, Peking University, China
[2]Center for Biomedical Image Computing and Analytics, University of Pennsylvania, US
[3]Shenzhen Graduate School, Peking University, China
{jzhaopku, rqxiong, zhangjian.sz, tjhuang}@pku.edu.cn, hfliu@upenn.edu

Abstract

The recently invented retina-inspired spike camera has shown great potential for capturing dynamic scenes. Different from the conventional digital cameras that compact the photoelectric information within the exposure interval into a single snapshot, the spike camera produces a continuous spike stream to record the dynamic light intensity variation process. For spike cameras, image reconstruction remains an important and challenging issue. To this end, this paper develops a spike-to-image neural network (Spk2ImgNet) to reconstruct the dynamic scene from the continuous spike stream. In particular, to handle the challenges brought by both noise and high-speed motion, we propose a hierarchical architecture to exploit the temporal correlation of the spike stream progressively. Firstly, a spatially adaptive light inference subnet is proposed to exploit the local temporal correlation, producing basic light intensity estimates of different moments. Then, a pyramid deformable alignment is utilized to align the intermediate features such that the feature fusion module can exploit the long-term temporal correlation, while avoiding undesired motion blur. In addition, to train the network, we simulate the working mechanism of spike camera to generate a large-scale spike dataset composed of spike streams and corresponding ground truth images. Experimental results demonstrate that the proposed network evidently outperforms the state-of-the-art spike camera reconstruction methods.



(a) Image captured by *iPhone11* (b) Spike stream

(c) TFP [36] (*ICME' 19*) (d) TFI [36] (*ICME' 19*)

(e) TVS [37] (*CVPR' 20*) (f) Proposed

Figure 1. The image for a high-speed rotation disk. For better comparison, we stuck a paper with characters "CVPR" to the disk. (a) The image captured by the camera of *iPhone11*, which suffers from severe blur. (b) Spike stream captured by the spike camera [6]. (c)-(f) Spike camera reconstruction results with different methods. Our proposed method evidently outperforms the competing methods, producing clear image for the high-speed object.

1. Introduction

With the prevalence of emerging computer vision applications, such as autonomous driving, robotics and unmanned aerial vehicle, there has been an increasing demand for capturing high-speed motion scenes, which makes the inherent limitations of conventional cameras become evident [17, 15]. Most conventional cameras use a certain exposure time window to accumulate the photoelectric information to form a snapshot image. Such an imaging mechanism requires the scene to be still during the exposure interval. Otherwise, a single point on a moving object may be projected onto different pixels on the image sensor, resulting in blurry artifacts for the moving objects.

To address this issue, a novel retina-inspired spike camera has been invented to capture dynamic scenes with improved quality [6, 7]. Instead of recording the visual information in the whole exposure interval by a snapshot, the spike camera abandons the concept of exposure window. Each pixel on spike camera sensor accumulates incoming light independently and persistently, and fires spikes whenever the dispatch threshold is reached, producing a continuous stream of spikes recorded at very high temporal resolution. Different from the bio-inspired event cameras [14, 30] that only record the *relative* light intensity changes at each pixel, the spike camera fires spikes to record the *absolute* light intensity, providing a more explicit information to reconstruct dynamic scenes.

Image reconstruction is one of the most important issues for the spike camera, and various image reconstruction methods have been proposed to recover the dynamic scenes from the recorded spike data [36, 37, 35]. TFI and TFP [36] focused on exploiting the physical properties of spike streams to infer the instantaneous light intensity, but they can not simultaneously handle the challenges that brought by both noise and high-speed motion, leading to unsatisfactory reconstruction as shown in Fig. 1(c) and 1(d). TVS [37] attempted to solve the problem by mimicking human vision. However, the mechanism of human vision is too complicated to be fully understood, and the reconstruction results are still inferior, as shown in Fig. 1(e).

Benefiting from the fast inference and excellent representation capability, deep learning has shown great potential in low-level vision applications [31, 10, 24, 31, 10, 24]. Inspired by the success of deep learning, this paper develops a deep convolutional neural network to reconstruct the dynamic scenes clearly from the spike streams. Considering the characteristics of asynchronous spike data, we propose a hierarchical architecture to exploit the temporal correlation progressively, so as to achieve high-quality reconstruction. We first propose a spatially adaptive light inference (SALI) subnet to infer the instantaneous light intensity of different moments by exploiting the *local temporal correlation*. In particular, the SALI applies several parallel learnable filters on various temporal scales, so as to adapt the temporal scale to various motion and light conditions. Then, to further improve the reconstruction, a motion-aligned image reconstruction subnet is utilized to fuse the intermediate features of different moments, so that the *long-term temporal correlation* can be exploited to refine the reconstruction.

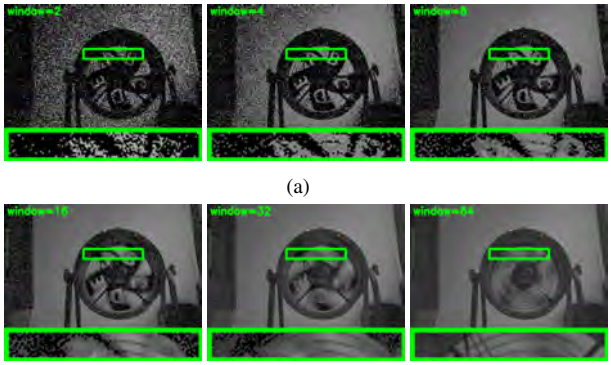
The main contributions of this paper are summarized as follows: 1) We develop an end-to-end convolutional neural network named Spk2ImgNet to reconstruct the dynamic scene from the spike camera data stream. To the best of our knowledge, this is the first attempt to solve the spike camera image reconstruction problem using an end-to-end neural network. 2) We propose a hierarchical architecture to exploit the temporal correlation progressively, so that the network can simultaneously handle the challenges brought by both noise and high-speed motion. 3) We formulate the mechanism of spike generation. Based on the analysis, we develop a spike camera simulator to generate synthesized spike stream and corresponding ground-truth images, and build a spike dataset for training spike camera image reconstruction network. 4) Experiments on both real captured spike data and synthesized spike data demonstrate that the proposed network achieves state-of-the-art reconstruction performance for dynamic scenes.

2. Related Work

Bio-inspired event camera. Event cameras [14, 30] are bio-inspired vision sensors that monitor the variation of light intensity persistently. They are good at capturing the motion information in a dynamic scene. However, as only the relative light intensity changes are recorded, event cameras can hardly reconstruct the texture details of the visual scenes, which degrades the visibility significantly. Although there are some works [1, 20] combing event camera with conventional image sensor to improve the reconstruction quality, there usually exists a motion mismatch due to the different sampling rate. To address this issue, some recent works [27, 22, 21, 2, 26] explored to use deep convolutional networks to directly reconstruct images from event streams. Different from these event cameras, the spike camera [6, 7] fires a positive signal to represent the arriving of a certain amount of photons, which provides a more explicit input format for reconstructing absolute light intensity.

Image reconstruction for spike camera. Image reconstruction is a fundamental issue for spike camera, and many reconstruction methods have been proposed in recent years. The texture from inter-spike interval (TFI) [36] inferred the instantaneous light intensity according to inter-spike intervals, which can provide a primary visual recovery of dynamic scenes, even for the regions with high-speed motion. However, due to the existence of thermal noise, such simple reconstruction usually appears to be visually unpleasant as shown in Fig. 1(d). To suppress the perturbation of noise, the texture from playback (TFP) [36] considered a longer photon accumulation period to infer the light intensity stably. However, for dynamic scenes with high-speed motions, a single point on the moving objects can be projected onto different pixels on the sensor, leading to motion blur as shown in Fig. 1(c). To address these issues, some research [37] devoted to exploiting retina-like visual image reconstruction frameworks to improve the reconstruction quality. However, the mechanism of human vision is too complicated to be fully understood and modeled, resulting in the unsatisfactory reconstruction as shown in Fig. 1(e).

Convolutional neural networks for low level vision. With the development of deep learning, recent years have



(a)

(b)

Figure 2. Reconstruction produced by accumulating the photon in various virtual exposure windows. (a) Using a small window can reconstruct the textures of dynamic objects, but suffers from noise. (b) Using a large window can stabilize the observed signal, but suffers from blur.

witnessed a rapid progress of deep convolutional neural networks for low level vision, such as super-resolution [25, 5, 8, 11], deblurring [19, 12] and denoising [31, 10, 24]. Dong *et al.* [5] proposed SRCNN to generate high resolution images from degraded low resolution images. Nah *et al.* [19] proposed a multi-scale convolutional neural network to remove blurring in dynamic scenes. Zhang *et al.* [31] proposed DnCNN to learn the mapping from noisy images to noise for image denoising. Zhang *et al.* [34] proposed RDN to fully exploit the hierarchical features for promising image restoration. Some other works embedded powerful modules, such as attention block [32, 4, 33] and feedback block [13], to further improve the performance of network. These CNN-based methods have demonstrated the great capability of deep learning for low level vision tasks, which inspires us to solve the spike camera image reconstruction problem with CNN. In particular, we aim to directly reconstruct high-quality images from the continuous spike stream instead of the degraded images.

3. Discussion on Spike Camera

In this section, we first formulate the mechanism of spike generation, and then present the spike camera reconstruction problem and its challenges.

3.1. Spike generation

Spike camera is composed of an array of pixels, each of which accumulates the incoming light $I(t)$ persistently. Once the dispatch threshold θ is reached, a spike is fired and the integrator is reset, restarting a new "integrate-and-fire" cycle. With such mechanism, the instantaneous electric charge amount on the integrator can be formulated as

$$A(t) = \int_0^t \alpha \cdot I(x)dx \mod \theta, \tag{1}$$

where α is the photoelectric conversion rate.

Ideally, a pixel may fire spikes at arbitrary time t_k, which satisfies:

$$\int_0^{t_k} \alpha \cdot I(x)dx = k\theta \tag{2}$$

namely, $A(t_k) = 0$. Here k denotes the spike index. However, due to the limitations of circuit technology, the spike reading times is quantified and a pixel can only read out spikes as discrete-time signals $S(n)$. To be specific, the pixel periodically checks the spike flag at the time $t = nT, n = 1, 2, \cdots$, where T is a short interval of microseconds. If a spike flag has been set up at the time $t = nT$, it reads out $S(n) = 1$, and resets the flag for the arriving of the next spike. Otherwise, it reads out $S(n) = 0$.

As the light comes in continuously, all the pixels on the sensor work simultaneously and independently, firing spikes to represent the arrival of every certain amount of photons. The sensor uses a high-speed polling to check the spike status ("0" or "1") of every pixel, generating an $H \times W$ spike frame. As the time goes on, the camera would produce a sequence of spike frames, i.e., an $H \times W \times N$ binary spike stream $S(x, y, n)$ as shown in Fig. 1(b).

3.2. Image reconstruction

Problem statement. Image reconstruction is a fundamental issue for spike cameras. For simplicity, we use $S_n \in \{0, 1\}^{W \times H}$ to denote the n-th spike frame, and use I_n to denote the instantaneous light intensity at the n-th polling. The goal of spike camera image reconstruction is to restore the original light intensity $\{I_n\}, n = 1, 2, \cdots N$, from the recorded binary spike stream $\{S_n\}, n = 1, 2, \cdots N$.

Challenges. An intuitive way to reconstruct I_n is to accumulate the photon in a virtual exposure window as the conventional imaging model. Considering that each spike corresponds to a certain amount of photons, we can estimate $I_n(x, y)$ by counting the number of spikes, which can be formulated as

$$\hat{I}_n(x, y) = \frac{\theta}{|\phi_n|} \cdot \sum_{i \in \phi_n} S_i(x, y), \tag{3}$$

where ϕ_n is the virtual exposure window. However, it is hard to choose an appropriate virtual exposure window. Fig. 2 shows the reconstruction using different virtual exposure windows. Note that the *short-term photon accumulation* using a small virtual exposure window can reconstruct the outlines of fast moving objects, but the reconstruction is quite noisy. This is because the number of incoming photons in a very short interval is a random variable, which is typically assumed to be Possion distributed. As a result, the signal is temporally unstable. In addition, the quantization effects of spike reading time (as discussed in Sec. 3.1) can also incur noise. On the other hand, the *long-term photon accumulation* using a large virtual exposure window can

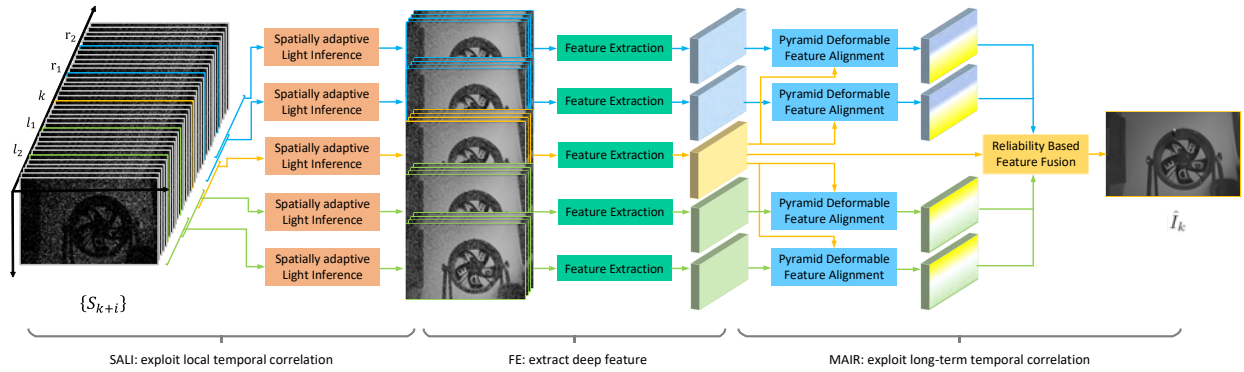


Figure 3. Illustration of the proposed end-to-end spike camera reconstruction network (Spk2ImgNet), which consists of three components: spatially adaptive light inference (SALI) subnet, feature extractor (FE) subnet and motion aligned image reconstruction (MAIR) subnet.

stabilize the observed light signal, but the high-speed motion objects appears to be blurry. This is because a single point on the moving objects can be projected onto different pixels on the image sensor.

To address these issues, we develop a new image reconstruction approach for spike cameras, integrating the advantages of both short-term photon accumulation and long-term photon accumulation, to effectively handle the challenges brought by both noise and high-speed motion.

4. Spike Camera Reconstruction Network

4.1. Overview

To reconstruct the dynamic scene from the binary spike stream, we develop an end-to-end trainable spike camera reconstruction network, dubbed Spk2ImgNet. The overall framework is illustrated in Fig. 3. To reconstruct the frame I_k, Spk2ImgNet takes a set of consecutive spike frames $\mathcal{S}_k = \{S_{k+i}\}, i = \pm1, \pm2, \cdots$ around S_k as input, so that it can exploit the temporal correlation to generate a high-quality reconstruction $\hat{I}_k$.

In order to take full advantage of the temporal correlation, we propose a hierarchical architecture to exploit the temporal correlation progressively. To be specific, Spk2ImgNet mainly consists of three components: spatially adaptive light inference (SALI) subnet, feature extractor (FE) subnet and motion-aligned image reconstruction (MAIR) subnet. Firstly, the input spike stream is partitioned into five overlapping short-term spike blocks $\{\mathcal{B}_{l_2}, \mathcal{B}_{l_1}, \mathcal{B}_k, \mathcal{B}_{r_1}, \mathcal{B}_{r_2}\}$ and the light inference subnet is applied to these spike blocks to take advantage of the local temporal correlation adaptively, producing several coarse estimation stacks $\{\tilde{I}_{l_2}, \tilde{I}_{l_1}, \tilde{I}_k, \tilde{I}_{r_1}, \tilde{I}_{r_2}\}$. Here, $\{l_1, l_2\}$ and $\{r_1, r_2\}$ are reference time points distributed symmetrically on the side of point k. In particular, the light inference modules are spatially adaptive, which can automatically adjust the temporal scale to adapt to various motion and light conditions. Subsequently, five share-weight feature extraction modules are applied to the coarse reconstructions to extract deep features $\{F_{l_2}, F_{l_1}, F_k, F_{r_1}, F_{r_2}\}$. Finally, to further improve the reconstruction quality, a motion-aligned image reconstruction subnet is cascaded to exploit the long-term temporal correlation, while avoiding motion blur. In this subnet, a pyramid deformable alignment is first employed to align the reference features to the key features, generating the aligned reference features $\{\bar{F}_{l_2}, \bar{F}_{l_1}, \bar{F}_{r_1}, \bar{F}_{r_2}\}$. Then, the aligned reference features are integrated with the key features to reconstruct the high-quality image $\hat{I}_k$.

4.2. Spatially adaptive light inference

Considering the fact that the light intensity within an extremely short interval is relatively consistent, we first exploit the local temporal correlation to infer the instantaneous light intensity of different moments. However, due to the diversity of scene contents, a fixed temporal scale can hardly properly handle all the pixels in the scene. For example, for the dynamic regions with very high-speed motion, an extremely short temporal scale should be used to avoid motion blur. Conversely, for the low light regions, a relatively long temporal scale should be used to accumulate enough photon information to reconstruct image textures. To address these issues, we propose a spatially adaptive light inference subnet (as shown in Fig. 4) to exploit the local temporal correlation adaptively, so that it can adapt to various motion and light conditions. Firstly, a multi-scale light inference module is applied to inter the instantaneous light intensity with various temporal scale. Then, a spatially adaptive selection module is employed to automatically select the appropriate temporal scale.

Multi-scale light inference. Several parallel learnable filters with different temporal scales are applied to the input short-term spike block $\mathcal{B}_t \in \mathbb{R}^{H \times W \times N}$, each of which generates a coarse instantaneous light intensity estimate with

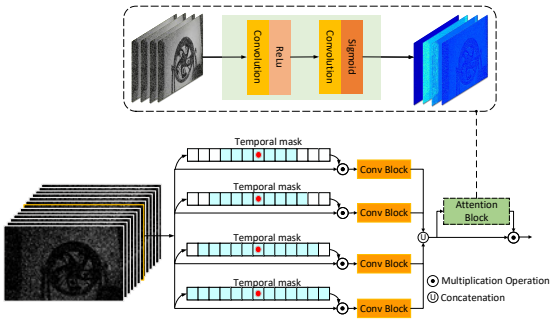


Figure 4. The architecture of spatially adaptive light inference subnet. Each branch works on a different temporal scale so that the multi-scale temporal dependency can be adaptively exploited.

certain temporal scale:

$$H_t(i) = h_i\left(M_i \circ \mathcal{B}_t\right). \tag{4}$$

Here M_i denotes the temporal mask of the i-th branch, $h_i(\cdot)$ denotes the learnable temporal filtering operation based on concatenated convolution and activation layers and $\circ$ represents element-wise multiplication.

Spatially adaptive temporal scale selection. An attention block $a(\cdot)$ is introduced to learn the modulation scaler maps

$$[W_t(1), \cdots, W_t(m)] = a([H_t(1), \cdots, H_t(m)]), \tag{5}$$

so that the network can be more focused on the estimate using the appropriate temporal scale. Here m is the number of branches and $[,]$ denotes feature concatenation. The output of the SALI subnet is a stack of scaled coarse estimates based on different temporal scale dependencies:

$$\tilde{I}_t = [W_t(1) \circ H_t(1), \cdots, W_t(m) \circ H_t(m)]. \tag{6}$$

In this paper, SALI is applied to five short-term spike blocks $\{\mathcal{B}_{l_2}, \mathcal{B}_{l_1}, \mathcal{B}_k, \mathcal{B}_{r_1}, \mathcal{B}_{r_2}\}$ respectively, producing the intermediate coarse estimation stacks $\{\tilde{I}_{l_2}, \tilde{I}_{l_1}, \tilde{I}_k, \tilde{I}_{r_1}, \tilde{I}_{r_2}\}$ to roughly describe the instantaneous light intensity at different moments.

4.3. Feature extraction

Then, we extract deep features F_t from each coarse estimation stack, which can be formulated as

$$F_t = f_t(\tilde{I}_t). \tag{7}$$

Here $f_t(\cdot)$ represents the feature extraction operation. Inspired by the excellent performance of residual blocks [9], which learn residual mappings by incorporating the self-identity, we use several stacked residual blocks to build our feature extraction module as illustrated in Fig. 5. In particular, a long-skip connection is used to forward information and ease the training difficulty. To reduce the complexity of the network, the parameters of $f_t(\cdot)$ are shared across all the branches.

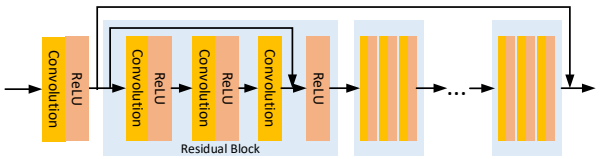


Figure 5. The architecture of feature extraction subnet.

4.4. Motion-aligned image reconstruction

Finally, a motion-aligned subnet is cascaded to exploit the long-term temporal correlation and refine the reconstruction, without introducing motion blur.

4.4.1 Pyramid deformable feature alignment

To improve the reconstruction quality, we further explore to exploit the long-term temporal correlation, while avoiding motion blur. To this end, we introduce a pyramid deformable alignment (PDA) module based on deformable convolution [3, 38] to align the reference features $\{F_{l_2}, F_{l_1}, F_{r_1}, F_{r_2}\}$ to the key features F_k.

Due to the promising transformation modeling capability, deformable convolution has been widely used to align features without explicit motion estimation [28, 25, 29, 23]. In deformable convolution, additional offsets are learned to augment the spatial sampling locations. Specially, to handle the large and complex motions effectively, we perform the deformable alignment in coarse-to-fine process. As illustrated in Fig. 6, given reference features $F_t, t \in \{l_2, l_1, r_1, r_2\}$ and key features F_k, we downsample the features into L-level features, i.e., $\{F_t^{(l)}\}$ and $\{F_k^{(l)}\}$, $l = 1, 2, \cdots, L$. With the pyramid features, we first align the features in lower scales with coarse estimations, and then propagate the aligned features and learned offsets to higher scales to refine the estimations. The aligned reference features at the l-th level can be calculated by

$$\bar{F}_t^{(l)}(p) = \sum_{i=1}^{n} w_i \cdot F_t^{(l)}(p + p_i + \Delta p_i^{(l)}) \cdot \Delta c_i^{(l)}, \tag{8}$$

where $p = (x, y)$ is the center coordinate, n is the number of sampling locations, w_i is the i-th weight and p_i is the i-th fixed offset. $\Delta c_i^{(l)}$ and $\Delta p_i^{(l)}$ are the i-th modulation scalar and the i-th learnable offset, which are predicted according to the $(l-1)$-th level estimated offsets $\Delta P^{(l-1)}$, and the l-th level reference features $F_t^{(l)}$ and key features $F_k^{(l)}$.

4.4.2 Reliability based feature fusion

With the aligned features $\{\bar{F}_{l_2}, \bar{F}_{l_1}, F_k, \bar{F}_{r_1}, \bar{F}_{r_2}\}$, we use a feature fusion module (as shown in Fig. 7) to fuse the features across long temporal range. Due to the existence of object occlusion and illumination changes, the alignment

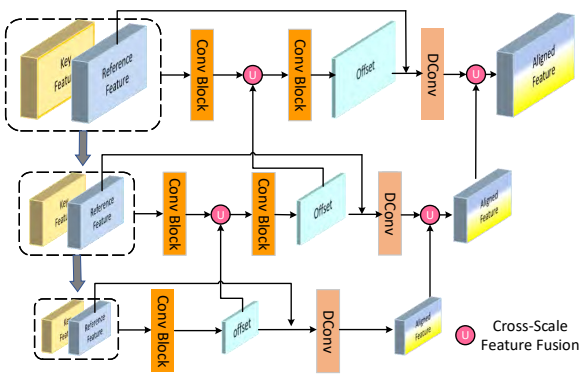


Figure 6. The architecture of the pyramid deformable alignment module. We employ a pyramid model to handle various scale motion in coarse-to-fine progress.

can be non-reliable in some regions. In order to integrate the aligned features effectively, we introduce a reliability estimator (RE) to measure the reliability of each pixel in aligned features. To obtain the reliability maps for aligned features $\bar{F}_t, t \in \{l_2, l_1, r_1, r_2\}$, we feed $\bar{F}_t$ along with the key features F_k to the reliability estimator $e_t(\cdot)$, producing the reliability map:

$$A_t = e_t([\bar{F}_t, F_k]). \tag{9}$$

With the aid of reliability maps, a robust fusion is adopted to reconstruct the final image:

$$\hat{I}_k = g([A_{l_2} \circ \bar{F}_{l_2}, A_{l_1} \circ \bar{F}_{l_1}, F_k, A_{r_1} \circ \bar{F}_{r_1}, A_{r_2} \circ \bar{F}_{r_2}]). \tag{10}$$

Here $g(\cdot)$ denotes the reconstruction function based on concatenated convolution and activation layers.

4.5. Loss function

Since high-quality intermediate estimates would promote the final reconstruction, we adopt a hierarchical loss function to optimize our network, which is formulated as

$$\mathcal{L} = \lambda \sum_{t \in \phi_k} \sum_{i=1}^{m} \|H_t(i) - I_t\|_1 + \|\hat{I}_k - I_k\|_1. \tag{11}$$

The first term is used to produce intermediate estimates and the second term is used to generate the high-quality final reconstruction. λ is the parameter balancing these two losses. $\phi_k = \{l_2, l_1, k, r_1, r_2\}$ is the set of key time points.

5. Experiments

5.1. Dataset

Training data. To train the network, a training dataset, including a large amount of spike streams with corresponding ground truth images is required. However, producing high-quality ground truth images for dynamic scenes is challenging. To address this issue, we develop a spike camera simulator to simulate the spike generation mechanism as introduced in Sec. 3.1, so that we can generate a large-scale training dataset with spike streams and ground truth images from video-based virtual scenes. To be specific, we regard each selected video as the scene to record, and assume that the motion between two adjacent frames is consistent. With the motion information, we can approximate the dynamic light intensity variation process, so that each pixel of the "sensor" accumulates the light intensity (i.e., the pixel intensity of image) continuously and checks the accumulated value periodically, producing a sequence of $H \times W$ spike frames. Besides, we also extract the "sensor" monitoring regions of the frames, producing corresponding ground-truth images. Here we use the videos from REDS [18] as the virtual scenes and employ the optical flow method [16] to estimate the motion information. As the PKU FSM spike camera [6], we set the sensor spatial resolution (i.e., $H \times W$) to 400×250. Finally, we establish a spike dataset composed of 800 spike stream-ground truth pairs, for spike camera reconstruction.

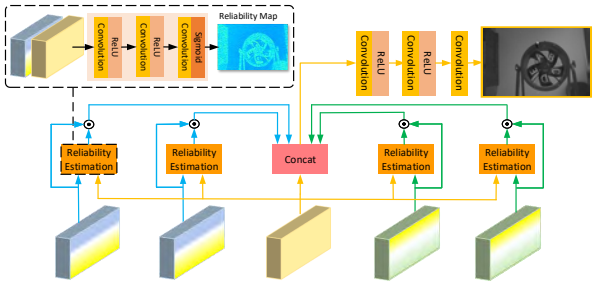


Figure 7. The architecture of the reliability based feature fusion.

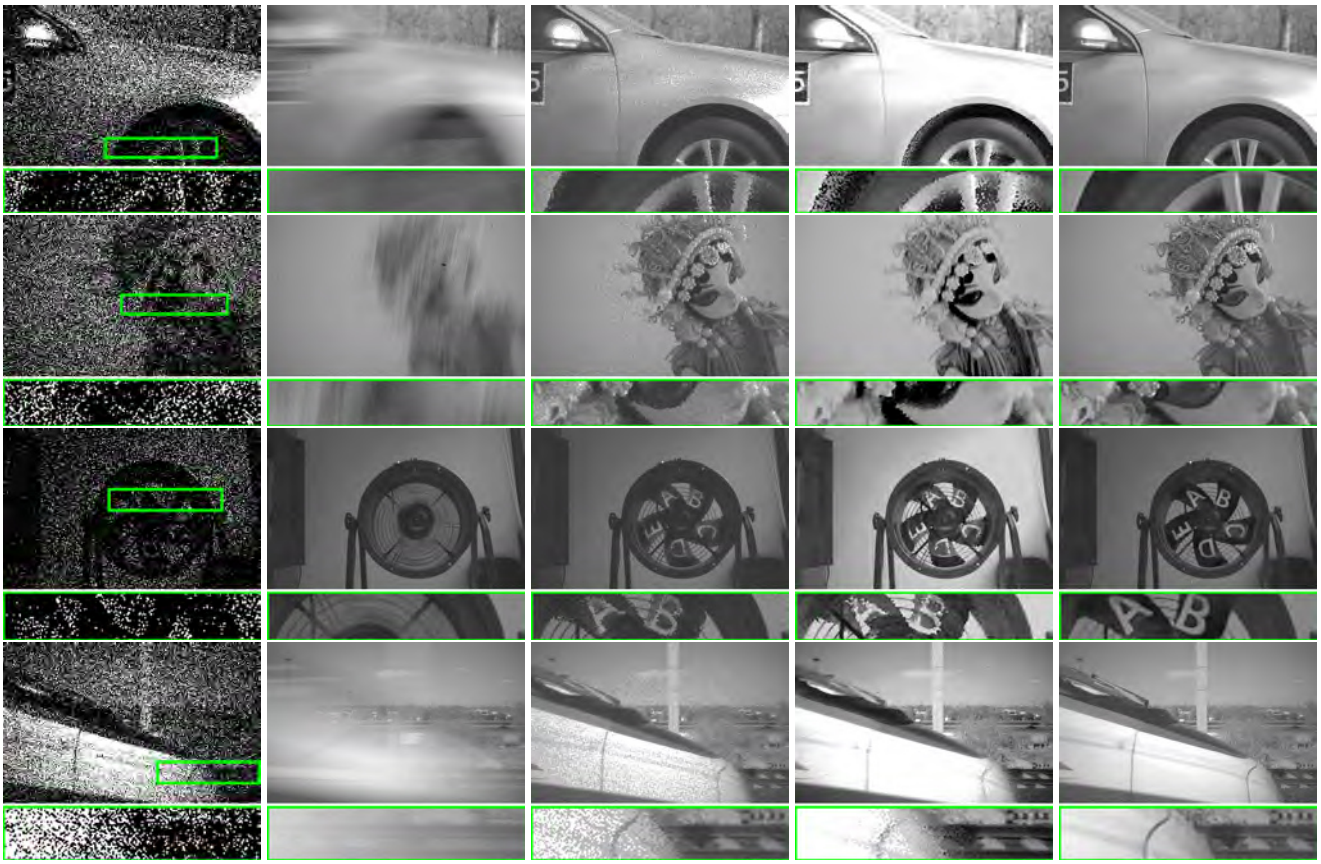
Table 1. Detail information of real-life spike streams

Name	Sample rate	Resolution	Description
Car	20000 HZ	400×250	run in 100km/h
Doll	20000 HZ	400×250	fall from a height
Fan	40000 HZ	400×250	rotate in 2600 rpm
Train	20000 HZ	400×250	run in 350 km/h

Testing data. To evaluate the performance of our proposed network, we conduct experiments on both synthesized data and real-life data. For synthesized data, we use the developed simulator to generate a test dataset composed of 40 spike stream-ground truth pairs, refer to Spike40[1]. For the real-life data, we not only use the PKU-Spike-High-Speed Dataset[2] but also capture several additional spike streams using the PKU FSM spike camera. The detail of the real-life spike streams is illustrated in Table 1.

[1]The dataset is available at https://cove.thecvf.com/datasets/517.

[2]The dataset is available at https://www.pkuml.org/resources/pku-spike-high-speed.html.



(a) Spike (b) TFP [36] (c) TFI [36] (d) TVS [37] (e) Spk2ImgNet

Figure 8. Comparison of different reconstruction methods on real captured spike streams. Please enlarge the figure for more details.

Table 2. Comparison of quantitative results on synthesized dataset.

Metric	TFP [36]	TFI [36]	TVS [37]	Proposed
PSNR	22.37	24.94	19.03	**38.44**
SSIM	0.5801	0.7150	0.7452	**0.9767**

Table 3. Comparison of NIQE (↓) on real-life spike streams.

Name	TFP [36]	TFI [36]	TVS [37]	Proposed
Car	7.6423	13.0197	9.3054	**4.0028**
Doll	8.2026	7.9594	7.4768	**3.9737**
Fan	7.2340	11.9794	6.2319	**3.7233**
Train	6.4892	10.6230	6.7824	**3.7140**
Average	7.3920	10.8954	7.4491	**3.8532**

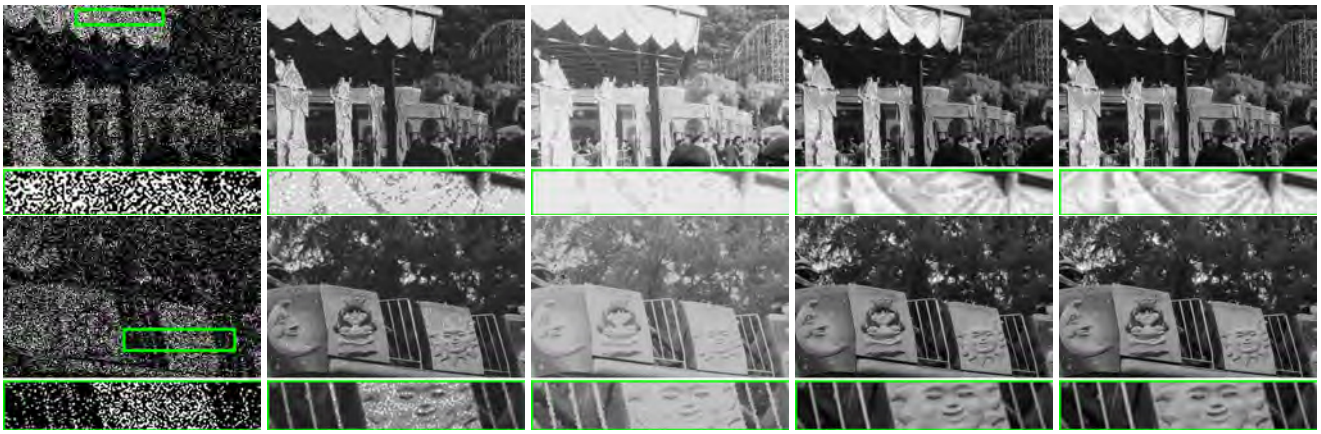
5.2. Implementation details

In our implementation, fifteen residual blocks are used in feature extraction subnet. The parameter λ of loss function is set to 0.02. We crop the spike frames into 40×40 patches and set batch size to 16. In addition, we set the long-term temporal window size and short-term temporal window size to 41 and 11, respectively. During training, data augmentation is performed by randomly rotating $90°$, $180°$, $270°$ and horizontally flipping. We use Adam optimizer with the default setting to optimize our network and implement our experiments with GTX 1080Ti GPU.

5.3. Comparison with state-of-the-art methods

To evaluate our proposed Spk2ImgNet, we compare it with recent works, i.e., texture from playback (TFP) [36], texture from inter-spike interval (TFI) [36] and texture via spiking neural model (TVS) [37].

Qualitative evaluation. Fig. 8 and Fig. 9 show the reconstruction results of different methods for real data and synthesized data, respectively. The visual quality of the reconstructions produced by our proposed method is evidently better than the competing methods. There are severe undesired motion blurry artifacts in the reconstruction of TFP, especially for the regions with high-speed motion. Although the TFI and TVS can well reconstruct the outlines of fast moving objects, the reconstruction typically appears to be noisy. In contrast, our proposed method achieves sta-



(a) Spike (b) TFI [36] (c) TVS [37] (d) Spk2ImgNet (e) GT

Figure 9. Comparison of different reconstruction methods on synthesized spike streams. Please enlarge the figure for more details.

Table 4. Evaluation for the effect of the proposed modules.

Metric	Net_a	Net_b	Net_c	Net_d	Net_e
SALI		✓		✓	✓
PDA			✓	✓	✓
RE					✓
PSNR	37.03	37.27	37.99	38.27	38.44
SSIM	0.9710	0.9718	0.9743	0.9768	0.9767

ble reconstruction, while restoring clear textures and details, even for high-speed moving objects.

Quantitative evaluation. To compare different reconstruction methods quantitatively, we use two reference image quality assessment (IQA) metrics, i.e., PSNR and SSIM, to evaluate the performance on synthesized data, and a no-reference IQA metric, i.e., NIQE, to evaluate the performance on real captured data. As illustrated in Table 2, we note that our Spk2ImgNet achieves the best reconstruction performance on the synthesized dataset. In particular, our Spk2ImgNet achieves a PSNR gain over 10dB, which demonstrates its effectiveness. In addition, from Table 3, we can observe that the Spk2ImgNet achieves the best NIQE quality for all the real-life spike streams.

5.4. Ablation study

To investigate the effect of the proposed modules, we compare five different models. Net_a is a basic baseline without SALI, PDA or RE, which infers the intermediate light intensity over a fixed temporal range and directly fuse the deep intermediate features of different moments using a concatenation operation without PDA and RE. Net_b adds SALI structure into Net_a, which adaptively exploits the multi-scale temporal correlation to infer the intermediate light intensity. Net_c adds PDA into Net_a, which aligns the intermediate features before fusion. Net_d simultaneously adds SALI and PDA into Net_a. Net_e is a full model with SALI, PDA and RE, i.e., the proposed Spk2ImgNet.

From Table 4, we note that Net_b outperforms Net_a by about 0.25dB, which validates that adaptively exploiting the multi-scale temporal correlation is beneficial for improving the reconstruction. We also observe that Net_c achieves much better performance than Net_a, demonstrating that the motion alignment is important for our reconstruction task. In addition, we observe that Net_e further improves PSNR over Net_d by 0.16dB. This is because the alignment may not be reliable in some cases, and our proposed RE can produce valid reliability maps to guide a more effective fusion.

6. Conclusion

This paper presents an end-to-end neural network named Spk2ImgNet to reconstruct dynamic scenes from continuous spike stream. To simultaneously handle the challenges brought by both noise and high-speed motion, we propose a hierarchical architecture to exploit the temporal correlation progressively, so that the network can exploit the photo-electronic information across a long temporal range, while avoiding undesired motion blur. In addition, to train Spk2ImgNet, we develop a simulator to simulate the spike camera working mechanism and generate a large-scale synthesized spike dataset, including spike streams and corresponding ground truth images. Experiments on both real-life spike data and synthesized spike data show that the proposed Spk2ImgNet can reconstruct high-quality images for dynamic scenes, which significantly outperforms state-of-the-art spike camera reconstruction methods.

Acknowledge

This work was supported by the National Natural Science Foundation of China under Grant 62072009, 62088102, 61772041, and also supported by the Cloud Brain II of Peng Cheng Lab, Shenzhen, China.

References

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

[2] Jonghyun Choi, Kuk-Jin Yoon, et al. Learning to super resolve intensity images from events. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2768–2776, 2020. 2

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 764–773, 2017. 5

[4] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *IEEE conference on computer vision and pattern recognition*, pages 11065–11074, 2019. 3

[5] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision (ECCV)*, pages 184–199. Springer, 2014. 3

[6] Siwei Dong, Tiejun Huang, and Yonghong Tian. Spike camera and its coding methods. *Data Compression Conference, DCC*, page 437, 2017. 1, 2, 6

[7] Siwei Dong, Lin Zhu, Daoyuan Xu, Yonghong Tian, and Tiejun Huang. An efficient coding method for spike camera using inter-spike intervals. *Data Compression Conference, (DCC)*, page 568, 2019. 2

[8] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3897–3906, 2019. 3

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 5

[10] Xixi Jia, Sanyang Liu, Xiangchu Feng, and Lei Zhang. Focnet: A fractional optimal control network for image denoising. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6054–6063, 2019. 2, 3

[11] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3224–3232, 2018. 3

[12] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8183–8192, 2018. 3

[13] Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu. Feedback network for image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[14] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128×128 120 db 15μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. 2

[15] 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 *Digital Signal Processing Workshop-signal Processing Education Workshop*, pages 173–178, 2006. 1

[16] Ce Liu et al. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Massachusetts Institute of Technology, 2009. 6

[17] Thomas B Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006. 1

[18] 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 *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020. 6

[19] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3883–3891, 2017. 3

[20] Christoph Posch, Daniel Matolin, and Rainer Wohlgenannt. An asynchronous time-based image sensor. *IEEE International Symposium on Circuits and Systems*, pages 2130–2133, 2008. 2

[21] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3857–3866, 2019. 2

[22] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 2

[23] Gyumin Shim, Jinsun Park, and In So Kweon. Robust reference-based super-resolution with similarity-aware deformable convolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8425–8434, 2020. 5

[24] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *IEEE International Conference on Computer Vision (ICCV)*, pages 4539–4547, 2017. 2, 3

[25] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. Tdan: Temporally-deformable alignment network for video super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3360–3369, 2020. 3, 5

[26] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8315–8325, 2020. 2

[27] Lin Wang, I. S. Mohammad Mostafavi, Yo Sung Ho, and Kuk Jin Yoon. Event-based high dynamic range image and

very high frame rate video generation using conditional generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[28] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 0–0, 2019. 5

[29] Xiaoyu Xiang, Yapeng Tian, Yulun Zhang, Yun Fu, Jan P Allebach, and Chenliang Xu. Zooming slow-mo: Fast and accurate one-stage space-time video super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3370–3379, 2020. 5

[30] Daniel F Yu and Jeffrey A Fessler. Mean and variance of single photon counting with deadtime. *Physics in medicine and biology*, 45(7):2043–56, 2000. 2

[31] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2016. 2, 3

[32] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *European Conference on Computer Vision (ECCV)*, pages 286–301, 2018. 3

[33] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. *arXiv preprint arXiv:1903.10082*, 2019. 3

[34] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 3

[35] Jing Zhao, Ruiqin Xiong, and Tiejun Huang. High-speed motion scene reconstruction for spike camera via motion aligned filtering. In *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE, 2020. 2

[36] Lin Zhu, Siwei Dong, Tiejun Huang, and Yonghong Tian. A retina-inspired sampling method for visual texture reconstruction. *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1432–1437, 2019. 1, 2, 7, 8

[37] Lin Zhu, Siwei Dong, Jianing Li, Tiejun Huang, and Yonghong Tian. Retina-like visual image reconstruction via spiking neural model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1438–1446, 2020. 1, 2, 7, 8

[38] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9308–9316, 2019. 5