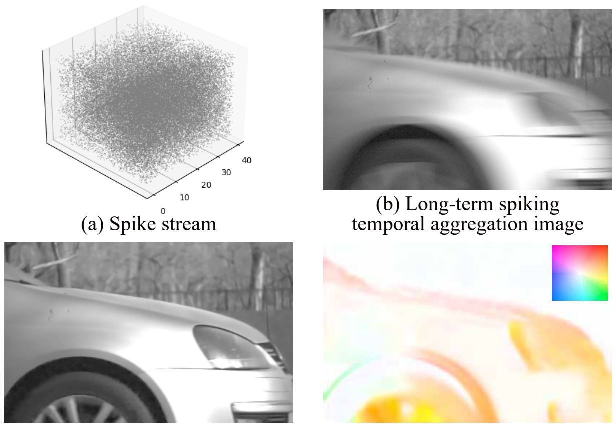
The Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)

Self-Supervised Joint Dynamic Scene Reconstruction and Optical Flow Estimation for Spiking Camera

Shiyan Chen[1], **Zhaofei Yu**[2,3*], **Tiejun Huang**[1,2,3]
[1]School of Electronic and Computer Engineering, Peking University
[2]Institute for Artificial Intelligence, Peking University
[3]School of Computer Science, Peking University
strerichia002p@stu.pku.edu.cn, {yuzf12, tjhuang}@pku.edu.cn

Abstract

Spiking camera, a novel retina-inspired vision sensor, has shown its great potential for capturing high-speed dynamic scenes with a sampling rate of 40,000 Hz. The spiking camera abandons the concept of exposure window, with each of its photosensitive units continuously capturing photons and firing spikes asynchronously. However, the special sampling mechanism prevents the frame-based algorithm from being used to spiking camera. It remains to be a challenge to reconstruct dynamic scenes and perform common computer vision tasks for spiking camera. In this paper, we propose a self-supervised joint learning framework for optical flow estimation and reconstruction of spiking camera. The framework reconstructs clean frame-based spiking representations in a self-supervised manner, and then uses them to train the optical flow networks. We also propose an optical flow based inverse rendering process to achieve self-supervision by minimizing the difference with respect to the original spiking temporal aggregation image. The experimental results demonstrate that our method bridges the gap between synthetic and real-world scenes and achieves desired results in real-world scenarios. To the best of our knowledge, this is the first attempt to jointly reconstruct dynamic scenes and estimate optical flow for spiking camera from a self-supervised learning perspective.



(a) Spike stream

(b) Long-term spiking temporal aggregation image

(c) Reconstruction image

(d) Optical Flow

Figure 1: llustrations of the spiking camera reconstruction and optical flow estimation task. The scene is about a high-speed car with a speed of 100 km/h. (a) Spike stream captured by a spiking camera. (b) Long-term spiking temporal aggregation image of the spike stream. (c-d) Reconstruction image and optical flow predicted through our framework.

Introduction

High-speed scenarios, such as autonomous driving and scientific imaging, are challenging for conventional cameras that generally accumulate photons information within a fixed exposure window. However, certain points on a fast-moving object may be projected onto different pixels on the sensor, which introduces motion blur. Besides, conventional high-speed cameras record frames synchronously at a constant shutter speed, resulting in significant data redundancy and memory consumption. In addition, the complex manufacturing process and high costs have prevented them from being widely used.

Recently, neuromorphic vision sensors (Chen et al. 2011; Dong et al. 2019; Gallego et al. 2020) have attracted extensive attention due to their outstanding ability to capture high-speed scenes. One of their newest members is a novel retina-inspired camera, named spiking camera (Huang et al. 2022; Dong, Huang, and Tian 2017). Instead of using an exposure window, the spiking camera mimics the sampling mechanism of the primate fovea (Wässle 2004; Masland 2012), with each of its photosensitive units continuously capturing photons and firing spikes asynchronously when the dispatch threshold is reached. These characteristics enable the spiking camera to record high-speed motion at a frequency of 40,000 Hz. Different from the commonly used event camera (also called dynamic vision sensor, DVS) (Lichtsteiner, Posch, and Delbruck 2008; Brandli et al. 2014), which only monitors relative brightness changes at each photosensitive unit, the spiking camera has the ability to record absolute light intensity as the spike firing frequency is proportional to the received scene radiance. This advantage allows the spiking camera to record more texture information than DVS, enabling the spiking camera more suitable for visualization of high-speed imaging.

Due to the distinct different sampling mechanisms, the

[*]Corresponding author

asynchronous spike stream generated by the spiking camera cannot be used directly for visualization. Therefore, it remains challenging to reconstruct dynamic scenes and perform common computer vision tasks for the spiking camera. Some recent works focus on reconstructing visually friendly images for the spiking camera (Zhu et al. 2019, 2020; Zheng et al. 2021; Zhao et al. 2021; Chen et al. 2022). The basic idea is to estimate pixel values from the firing frequency or firing interval (Zhu et al. 2019; Huang et al. 2022). The other works are inspired by the mechanism of the retina system (Zhu et al. 2020) and the short-term plasticity mechanism (Zheng et al. 2021). These methods can be classified as internal statistics methods and always suffer from noise and motion blur. More recent work (Zhao et al. 2021) has attempted to use convolutional neural networks to solve the reconstruction problem and achieved state-of-the-art performance. Other researchers devote to utilizing the spiking camera on downstream computer vision tasks, such as optical flow estimation and depth estimation (Hu et al. 2022). However, these deep learning-based methods usually require supervised training on large spiking synthetic datasets, which is hard and costly to generate because of unknown noise mechanisms and intricate imaging mechanisms inside the spiking camera.

In this work, we focus on excavating the potential information inside the spiking stream in a self-supervised manner. Specifically, we develop a self-supervised learning framework to jointly train a dynamic scene reconstruction network and an optical flow estimation network and share information during training. Inspired by previous self-supervised learning work (Chen et al. 2022), we employ blind spot networks (Lehtinen et al. 2018; Krull, Buchholz, and Jug 2019; Laine et al. 2019; Wu et al. 2020; Byun, Cha, and Moon 2021) to mine potential clean representations in the spiking stream and then train the optical flow network with the resulting frame-based representations. Furthermore, based on the assumption that the correct optical flow can re-render multiple reconstruction images into a blurred image, we propose an optical flow-based inverse rendering process to achieve self-supervision by minimizing the difference with respect to the original spiking temporal aggregation image. The performance of both networks can be improved by sharing information during joint training.

The main contributions are summarized as follows:

- We present the first self-supervised joint learning framework for optical flow estimation and reconstruction of spiking camera.
- We propose an optical flow based inverse rendering process to achieve self-supervision for the optical flow network, based on the assumption that the correct optical flow can re-render the reconstruction images back to the spiking temporal aggregation image.
- Experimental results on both synthetic and real-world datasets demonstrate that our method bridges the gap between synthetic and real-world scenes, and can produce noiseless desirable reconstruction images and reliably predict the optical flow.

Related Work

In this section, we briefly introduce the reconstruction and optical flow estimation algorithms of the two representative neuromorphic vision sensors, event camera and spiking camera, respectively.

Scene Reconstruction for Neuromorphic Vision Sensors. Event cameras have demonstrated their remarkable potential in capturing high-speed scenes (Gallego et al. 2020). However, the characteristic of recording only relative brightness changes makes it expert in capturing motion edge information, but poor at recording texture details in dynamic scenes. Several works (Choi, Yoon et al. 2020; Rebecq et al. 2019a,b; Pini, Borghi, and Vezzani 2018) attempt to recover texture information directly from the output event of the event camera by using convolutional neural networks. Other works (Brandli et al. 2014; Posch, Matolin, and Wohlgenannt 2008) further combine synchronized gray-scale images to supplement texture information. Recently, some researchers managed to solve this problem in a self-supervised way (Paredes-Vallés and de Croon 2021).

The advantage of the spiking camera over the event camera is that it can record more texture information. Similar to the integrate-and-fire mechanism of neurons (Gerstner and Kistler 2002), each photosensitive unit in the spiking camera sensor accumulates photons independently and generates a spike when the dispatch threshold is reached. Based on the principle of spike generation, some works (Zhu et al. 2019) manage to estimate pixel values from the firing frequency or firing interval, named “texture from play-back (TFP)” and “texture from inter-spike-intervals (TFI)”, respectively. Other works take a physiological perspective, by mimicking retina-like visual imaging (Zhu et al. 2020) or the short-term plasticity (Zheng et al. 2021). Although these methods are more explicable, the reconstruction results are unsatisfactory and suffer from noise. Spk2ImgNet (Zhao et al. 2021) introduces convolutional neural networks and achieves state-of-the-art performance, while the reconstruction results are sometimes distorted due to the use of deformable convolution (Dai et al. 2017). In addition, some researchers (Chen et al. 2022) attempt to excavate potential clean representations in the spiking stream through the blind-spot network and self-supervised mutual learning, alleviating the need for huge synthetic datasets.

Optical Flow for Neuromorphic Vision Sensors. EV-FlowNet (Zhu and Yuan 2018) is the pioneer in utilizing deep learning for event-based optical flow estimation. The training of EV-FlowNet depends on the photometric loss on gray-scale ground truth provided by the MVSEC dataset (Zhu et al. 2018). Further research manages to train the network unsupervised by using a novel loss function designed to eliminate the motion blur in event streams (Xu et al. 2021). To better excavate the temporal information in the event streams, SpikeFlowNet (Lee et al. 2020) and STE-Flow (Ding et al. 2022) integrate spiking neural networks and recurrent neural networks into the encoder to improve performance further.

Researchers in SCFlow (Hu et al. 2022) proposed the first synthetic optical flow dataset for the spiking camera. They

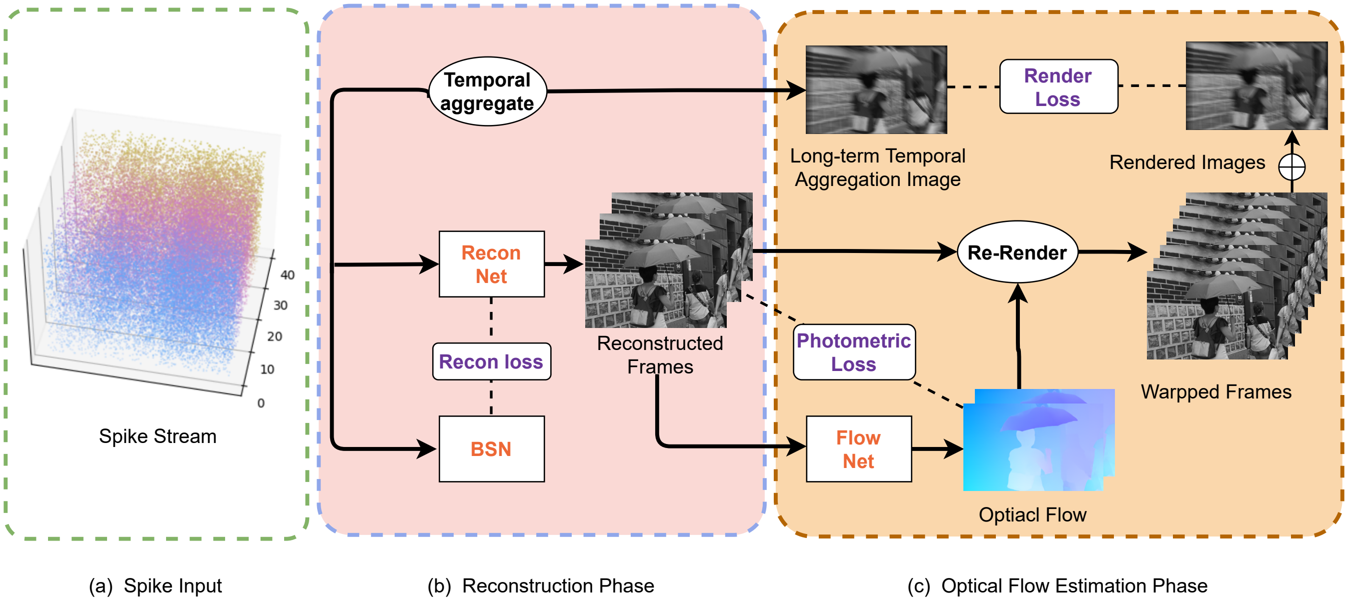


Figure 2: Illustration of the proposed framework. The framework can be divided into a reconstruction phase and an optical flow estimation phase. In this figure, a recon-net (reconstruction network) learns to produce clean frames through mutual distillation with a BSN (blind-spot network). The flow-net is trained with multi-scale photometric loss and inverse render loss. The inverse rendering process transfers information between two networks.

also propose a tailored neural network architecture with a novel input representation to estimate optical flow from the spiking stream in a supervised manner. However, similar to the event camera, it remains challenging to find a suitable way to estimate optical flow from the spiking stream from a self-supervised perspective.

Preliminary

In this section, we introduce the working mechanism of spiking camera in detail, and discuss a basic internal statistics method for spiking camera reconstruction.

Working Mechanism of Spiking Camera

The spiking camera is mainly composed of three parts: the photoreceptor, the accumulator, and the comparator. The photoreceptor consists of an $H \times W$ array of photosensitive units. The property of spiking camera to record high-speed scenes comes from the fact that each photosensitive unit continuously captures photons and fires spikes asynchronously when the dispatch threshold is reached. Specifically, we assume the incoming light intensity to be $L(t)$. The comparator detects whether the instantaneous electric charge amount $A(t)$ on the accumulator reaches a dispatch threshold θ. When θ is reached, a spike is fired, then the accumulator is reset to 0. We can formulate the process as:

$$A(t) = \int_0^t \alpha \cdot L(x)\mathrm{d}x \mod \theta, \tag{1}$$

where α is the photoelectric conversion rate.

Given $A(t_k) = 0$, a spike can fire at arbitrary time t_k in theory. However, due to the limitations of circuit technology, the spike fired at arbitrary time t_k can only be read out at discrete times. In fact, a spike flag is periodically checked with a fixed interval $T = 25$ μs, leading to the camera's sampling frequency of 40,000 Hz. A spike will be read out $S(x, y, n) = 1$ $(n = 1, 2, \ldots)$ if the spike flag has been set up at the time t in position (x, y), with $(n-1)T < t \leqslant nT$. In other cases, it reads out $S(x, y, n) = 0$. Thus a spike frame $S_n \in \{0, 1\}^{H \times W}$ is generated at each discrete timestamp n, and a spike stream $S \in \{0, 1\}^{H \times W \times N}$ is generated in a fixed time window N.

Basic Internal Statistics Reconstruction Method

One of the basic reconstruction methods for spiking camera is "texture from play-back (TFP)" (Zhu et al. 2019). As the spike firing frequency of the spiking camera is proportional to the received scene radiance, we can approximate the intensity at the moment n by calculating the number of spikes in a time window, and generate a spiking temporal aggregation image, which can be formulated as:

$$I_n^{\text{TFP}} = \frac{N_w}{w} \cdot C, \tag{2}$$

where w is the size of time window, N_w is the total number of spikes aggregated in the time window, and the C refers to the maximum dynamic range. The visual effect of I_n^{TFP} varies with different lengths of the window. While a larger window can eliminate the effect of noise, it will introduce more motion blur.

In general, as a basic refactoring method, TFP is simple enough and effective. In this work, we will utilize this kind of spiking temporal aggregation images in different windows to achieve self-supervision.

Methods

In this section, we first present the overall framework of our methods, and then introduce the reconstruction and optical flow estimation part, respectively. Base on the two parts, we will show the self-supervised joint training methodology of our framework.

Scene Reconstruction and Optical Flow Estimation

As illustrated in Fig. 2, our framework can be viewed as a pipeline consisting of a reconstruction network, an optical flow network, and an inverse rendering module that follows. We will start with the problem statement to introduce our methods.

Problem Statement. We denote $S_n \in \{0,1\}^{H \times W}$ as a spike frame at discrete timestamp n, and $S \in \{0,1\}^{H \times W \times N}$ as a spike stream in a fixed time window N. The goal of reconstruction is to obtain a visually friendly image I_n at time stamp n from the spiking stream S, corresponding to S_n. The goal of optical flow estimation is to predict optical flow f_{n_i,n_j} from time stamp n_i to n_j from the spiking stream S, corresponding to optical flow from S_{n_i} to S_{n_j}.

Spiking Temporal Aggregation Images. Both long and short-term spiking temporal aggregation images are generated by adding the input spiking stream through the time channel directly. As the generation of them is similar to the physics of the exposure window, we can find the images with short window tend to be noisy, which is suitable for training the blind-spot network (BSN) in a noise2void manner (Krull, Buchholz, and Jug 2019; Laine et al. 2019). The long window leads to more motion blur, thus can be used in the inverse render loss function. We'll cover the details in the following sections.

Blind-Spot Network for Reconstruction. As discussed in the Section of "Basic Internal Statistics Reconstruction Method", spiking temporal aggregation image in a small window appears to be noisy. We take a page from previous work (Chen et al. 2022) to use BSN to excavate potential clean representations in the spiking stream. The BSN is trained using noisy short-term spiking temporal aggregation image as its self-supervised signal. Due to the blind-spot constraints, BSN trained with noisy labels can produce clean output, which has been widely discussed in previous works (Lehtinen et al. 2018; Krull, Buchholz, and Jug 2019; Laine et al. 2019; Wu et al. 2020; Byun, Cha, and Moon 2021). We also use a similar mutual distillation method (Chen et al. 2022) to enhance the performance. To be specific, we use a shallow ResNet to produce K reconstructed frames, and mutual distillation is performed between the ResNet and the BSN. As the distillation is carried out among multiple frames, the mutual distillation can be viewed as a self-ensembling method, which is different from the previous self-supervised work (Chen et al. 2022).

Instead of building input representations by assuming novel inductive bias in previous works (Hu et al. 2022), we directly feed the spiking stream S into the reconstruction network to learn naive spiking representations to reduce computational complexity. To be specific, we regard the temporal dimension of S as the channel dimension. Thus the input of the network turns to $S_{in} \in \{0,1\}^{C \times H \times W}$, and the outputs of the network are K reconstructed frames $I_{0,\ldots,K}$ with time interval M.

We use the same blind-spot strategy as in (Laine et al. 2019). To be specific, the rotated versions of the input spike stream are concatenated together in the batch dimension and then passed through a shifted-conv based U-Net. The BSN output are then split into four parts in the batch dimension and concatenated together in the channel dimension, and finally passes through 1×1 convolutions to produce the estimated reconstruction frames.

The loss function of reconstruction can be formulated as:

$$\mathcal{L}_{Recon} = \frac{1}{K}\sum_{k=1}^{K} \|I_k - I_k^{bsn}\|_2^2 + \frac{\lambda_1}{K}\sum_{k=1}^{K} \|I_k^{bsn} - I_k^{tfp}\|_2^2, \tag{3}$$

where λ_1 is a weighting parameter, I_k denotes the k-th frame of the reconstruction outputs, I_k^{bsn} denotes the k-th frame of the BSN outputs, and the I_k^{tfp} denotes the k-th short-term spiking temporal aggregation images.

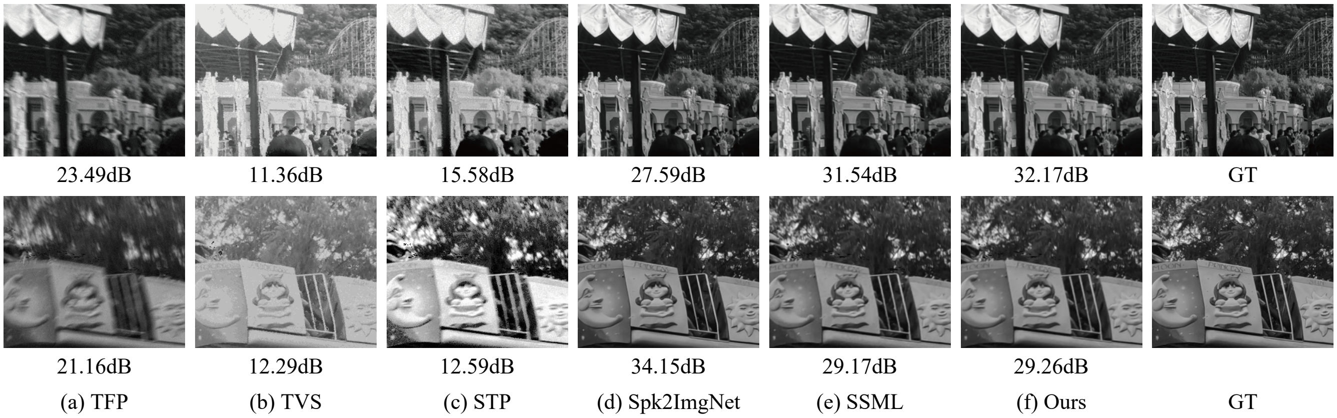
Optical Flow Estimation for Spiking Camera. As discussed above, the optical flow network aims to predict optical flow f_{n_i,n_j} from time stamp n_i to n_j from the spiking stream S, corresponding to optical flow from S_{n_i} to S_{n_j}. In previous work, SCFLow (Hu et al. 2022) takes two overlapped spiking streams as the input and predicts optical flow between the respective central time stamps of the two spiking streams. In this work, we prefer to predict a optical flow from single spiking stream between two timestamps. Our optical flow network takes clean spiking representations from the reconstructed network output as input to predict the optical flow between their corresponding timestamps.

We adopt the well-known PWC-Net (Sun et al. 2018) as our optical flow network. Despite of the fact that the original PWC-Net is trained in a supervised manner, we adopt multi-scale photometric loss to achieve self-supervision, referring to the previous research (Liu et al. 2020a). In detail, following the coarse-to-fine principle, we apply bidirectional photometric loss between every two adjacent reconstructed frames I_k and I_{k+1} at each level:

$$\begin{aligned}\mathcal{L}^l_{photo} = \frac{1}{2(K-1)} \sum_{k=1}^{K-1} (\rho(I^l_k(x) - I^l_{k+1}(x + f_{k,k+1})) \\ + \rho(I^l_{k+1}(x+1) - I^l_k(x + f_{k+1,k}))),\end{aligned} \tag{4}$$

where $\rho(*)$ denotes L1-loss function or SSIM loss function (Liu et al. 2020a), and I^l_k denotes the downsampling image at level l and time stamp k. And the multi-scale photometric loss is calculated as:

$$\mathcal{L}_{msphoto} = \sum_{l=1}^{L} w_l \mathcal{L}^l_{photo}. \tag{5}$$



(a) TFP (b) TVS (c) STP (d) Spk2ImgNet (e) SSML (f) Ours

Figure 3: Qualitative evaluation of different methods on synthetic reconstruction dataset.

where w_l denote weights of each level, and L denotes the number of pyramid levels. We follow the same occlusion handling method in ARFlow (Liu et al. 2020a).

Optical Flow Based Inverse Rendering

Given a set of clean and time-consistent frames L_t in a time window T, we can generate a blurry image $\hat{B}$ by simulating the physics of the exposure window:

$$\hat{B} = \frac{1}{T}\int_{t\in T} L_t dt. \tag{6}$$

Back to our work, we have discussed in the Section of "Basic Internal Statistics Reconstruction Method" that the spiking temporal aggregation image in a large window tends to be blurry. Further, the generation of long-term spiking temporal aggregation image is similar to the physics of the exposure window. Therefore, in this view, we can derive a blur-consistency constraint (Xu et al. 2021; Rozumnyi et al. 2021; Liu et al. 2020b) by comparing the re-rendered blurred image with the long-term spiking temporal aggregation image.

However, we cannot accurately simulate this process with a few discrete frames. To address the issue, we utilize the estimated optical flow to interpolate the discrete frames to get more frames, and then use them to re-render the blurred image. In practical, we can approximate this process using the average of N consistent frames $I_i (i = 1, 2, ..., N)$ with a time interval M between two adjacent frames to produce a blurry image I_B:

$$I_B = \frac{1}{N}\sum_{i=1}^{N} I_i, \tag{7}$$

where $N = (K-1)M + K$, which consists of K output frames directly from the reconstructed network and $(K-1)M$ frames produced by interpolation. Assuming that the forward optical flow between two output frames I_{n_1} and I_{n_2} is f_{n_1,n_2}, we can obtain intermediate frames by optical flow interpolation:

$$I_i^f(x) = I_{n_2}(x + \frac{m}{M+1} f_{n_1,n_2}). \tag{8}$$

Methods	Sup	Unsp			
	Zhao's	TVS	STP	SSML	**Ours**
PSNR	38.44	23.15	22.37	34.26	34.57
SSIM	0.9767	0.7452	0.7300	0.9718	0.9536

Table 1: Comparison among different reconstruction methods on synthetic dataset.

Or we can use the backward optical flow:

$$I_i^b(x) = I_{n_1}(x + \frac{m}{M+1} f_{n_2,n_1}), \tag{9}$$

where $m \in [1, M]$.

Then we can re-render the blurred image with Equ. 7 to generate I_B^f and I_B^b with forward and backward optical flow, respectively. We refer to the long-term spiking temporal aggregation image as I^{TFP}, then the blur-consistency can be formulated as:

$$\mathcal{L}_{render} = \|I_B^f - I^{TFP}\|_1 + \|I_B^b - I^{TFP}\|_1 \tag{10}$$

The blur-consistency constraint enables the transmission of information between the optical flow network and the reconstruction network, which provides further self-supervision for our framework.

Self-Supervised Joint Training

The proposed framework is trained jointly with two main networks sharing useful information through the novel losses, and the two networks are optimized alternately in an epoch. For the optical flow network, the total loss is:

$$\mathcal{L}_{flow} = \mathcal{L}_{msphoto} + \lambda_2\mathcal{L}_{render} + \lambda_3\mathcal{L}_{smooth} \tag{11}$$

where $\mathcal{L}_{smooth}$ is used to enhance the spatial consistency of neighboring flows through minimizing the difference between the neighboring flow.

Note that our framework is fully self-supervised. Our framework is the first work to relax the strong dependency of deep-learning-based approaches on ground-truth and huge synthetic data for dynamic scene reconstruction and optical flow estimation.

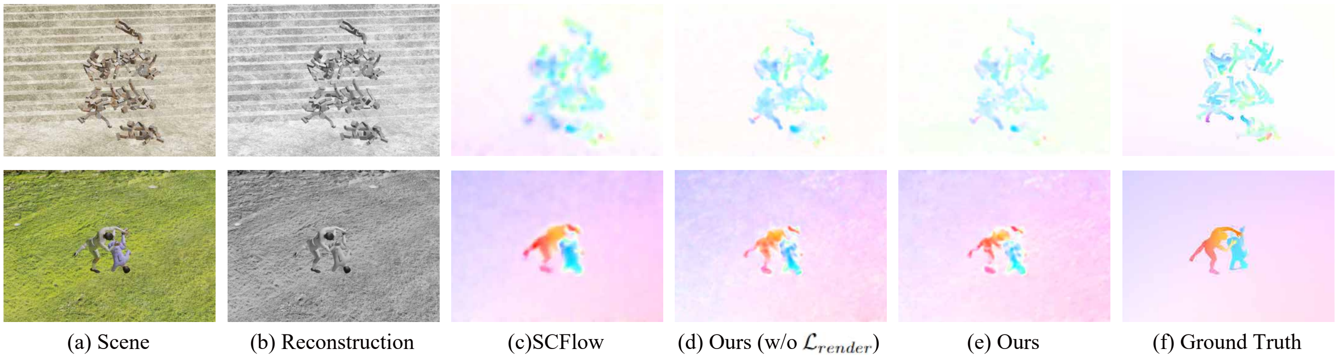

(a) Scene (b) Reconstruction (c)SCFlow (d) Ours (w/o $\mathcal{L}_{render}$) (e) Ours (f) Ground Truth

Figure 4: Qualitative Evaluation of Optical Flow estimation on Synthetic Dataset.

	Method	Ball	Cook	Dice	Doll	Fan	Fly	Hand	jump	Poker	Top	AVG.
$\Delta t = 10$	EV-FlowNet	0.571	3.000	0.761	1.273	0.928	11.599	5.011	0.742	1.167	2.762	3.590
	Spike-FlowNet	0.482	3.509	0.568	0.786	0.901	12.003	4.898	0.780	0.693	2.617	3.577
	SCFlow (supervised)	0.632	1.620	1.224	0.259	0.293	9.418	1.811	0.130	0.943	2.171	**2.568**
	Ours (w/o reblur)	0.488	2.705	1.303	0.539	0.447	9.679	3.632	0.241	1.244	2.473	3.085
	Ours	0.481	2.734	1.692	0.664	0.446	9.344	4.137	0.235	1.368	2.473	*2.613*
$\Delta t = 20$	EV-FlowNet	1.151	5.637	1.927	1.821	1.854	22.828	9.608	0.827	2.522	5.316	6.980
	Spike-FlowNet	0.987	7.048	1.122	3.039	1.839	25.130	9.816	1.902	1.397	5.423	7.565
	SCFlow (supervised)	1.115	3.320	2.582	0.515	0.566	20.835	4.442	0.240	1.884	4.301	**5.583**
	Ours (w/o reblur)	1.126	5.235	1.917	0.834	0.922	20.616	7.369	0.545	1.548	4.833	6.129
	Ours	1.410	4.696	2.520	0.764	0.788	19.628	6.570	0.464	1.723	4.793	*5.908*

Table 2: Evaluation on Optical Flow Synthetic Dataset. Bold: best. Italic: second.

Experiments

In this section, we evaluate the performance of our method on both synthetic and real-world datasets. We first introduce the dataset, then we compare method with the state-of-the-art image reconstruction and optical flow estimation methods. The pre-processing, hyper-parameters, and training details are provided in the Appendix.

Implementation Details

The parameter λ_1 of the reconstruction loss function is set to 100, and λ_2, λ_3 of the optical flow estimation loss function set to 0.1, 50. The input spike stream is cropped into 256×256 patches with a batch size of 4. The length of the input spike stream sequence is set to 41. The two networks are optimized alternately with two Adam optimizers, both of which have a learning rate of 1e-4 and $[\beta_1, \beta_2]$ of $[0.9, 0.99]$. We train the framework for 120 epochs for REDS dataset. For SPIFT dataset, we train the framework for 60 epochs and 120 epochs for $dt = 10$ and $dt = 20$ setting, respectively. Moreover, we train the reconstruction network for 15 epochs in advance in order to make the optical flow network converges better.

The number of reconstructed frames K is set to 3. The time interval M used to interpolate between two frames is set to 9. We set the window size of the short-term spiking temporal aggregation image I^{tfp} to 7 for REDS and real-world dataset, 5 for SPIFT dataset. The window size of long-term spiking temporal aggregation image I^{TFP} is set to 27 and 25 for REDS, real-world dataset and SPIFT dataset in $dt = 10$ setting and 41 for all dataset in $dt = 20$ setting.

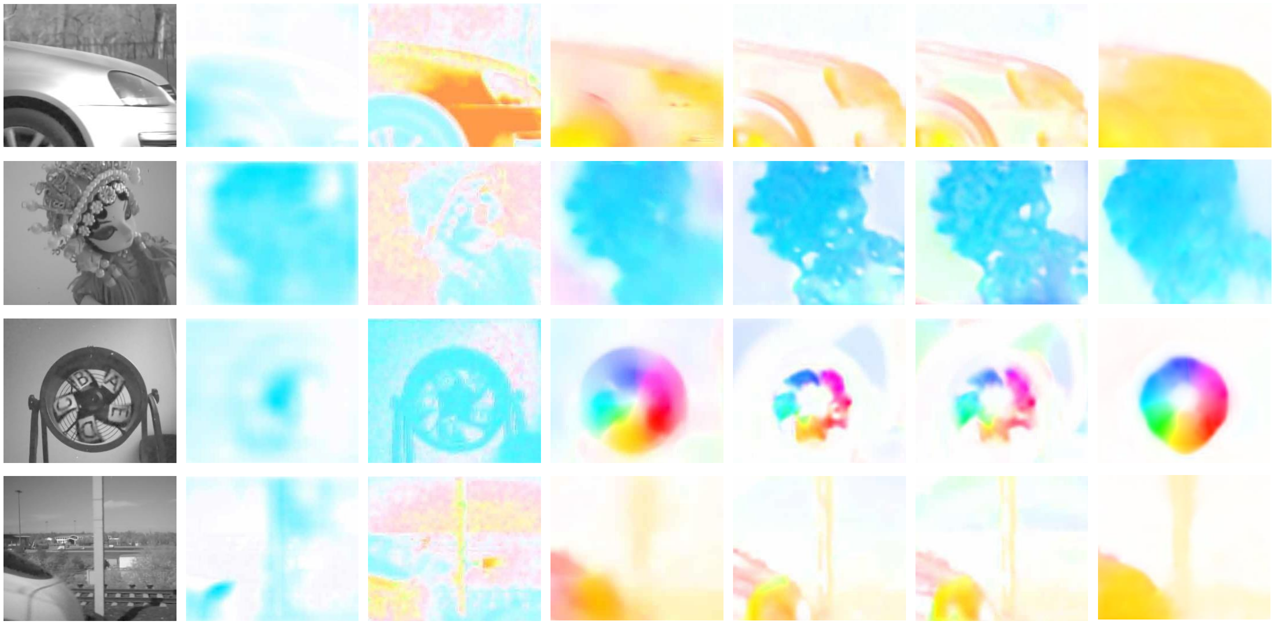
Datasets

As there are currently no datasets for both reconstruction and optical flow estimation, we conduct our experiments on two separate datasets.

The REDS reconstruction dataset is proposed by Spk2ImgNet (Zhao et al. 2021), which is generated by converting videos from REDS to spike stream. The training set consists of 800 spike stream-ground truth pairs with a spatial resolution of 400 × 250, and the testing set consists of 40 spike stream-ground truth pairs of the same size.

For the optical flow dataset, we use SPIFT (Hu et al. 2022) as the training set, which is generated from a spiking camera simulator. The SPIFT consists of 100 categories with a sequence length of 500 for each category, accompanying ground-truth optical flow. The corresponding testing set named PHM contains 10 categories.

The real-world dataset includes PKU-Spike-High-Speed Dataset (Zhu et al. 2020), which is directly captured by a spiking camera with a sampling rate of 40,000 Hz. This dataset consists of four different sequences, including a high-speed train in 350 km/h, a car in 100 km/h, a rotating



(a) Scene / Reconstruction (b) EV-FlowNet (c) Spike-FlowNet (d)SCFlow (e) Ours (w/o $\mathcal{L}_{render}$) (f) Ours (g) Ours (fine-tuned)

Figure 5: Qualitative Evaluation of optical flow estimation on Real-World Dataset

fan of 2600 rpm (revolutions per minute) and a doll in free fall.

Evaluation on Reconstruction Dataset

We compare our method with advanced internal statistics methods, TVS (Zhu et al. 2020), STP (Zheng et al. 2021) and deep learning methods, Zhao's Spk2ImgNet (Zhao et al. 2021) and SSML (Chen et al. 2022).

As illustrated in Tab. 1, our method outperforms previous internal statistics methods greatly and achieves achieves state-of-the-art performance in self-supervised methods. The quantitative results are slightly lower than the supervised method Spk2ImgNet, which demands a large synthetic dataset with ground truth. Besides, we found from the qualitative results that the reconstruction results produced by Spk2ImgNet sometimes appear distorted. As shown in Fig. 3, the distortion will lead to a lower PSNR value. We attribute this to the use of deformable convolution in Spk2ImgNet. In contrast, our method learns directly from inside the spiking stream without any labels and can obtain desirable results.

Evaluation on Optical Flow Dataset

We compare our optical flow results with SCFlow (Hu et al. 2022), which is trained in SPIFT dataset in a supervised manner. The quantitative comparison results are shown in Tab. 2 in both $\Delta t = 10$ and $\Delta t = 20$ settings. One can find that our self-supervised method can get comparable performance to the state-of-the-art supervised method SCFlow (2.613 vs. 2.586 for $\Delta t = 10$, and 5.908 vs. 5.583 for $\Delta t = 20$). We also conduct an ablation study for the novel inverse render loss. As shown in Fig. 4, our methods with the novel inverse render loss achieve finer motion boundaries and cleaner textures than the one without inverse render loss. And we can find from the quantitative results in Tab. 2 that our framework trained with the proposed novel loss function achieve better results, which demonstrates the efficiency of the proposed inverse rendering process in conveying useful information between two networks.

Qualitative Evaluation on Real-World Dataset

We also conduct experiments on the real-world dataset to demonstrate the effectiveness of our method. As the motion in the small real-world dataset is too simple to be suitable for optical flow training, we train the framework jointly on the real-world dataset and only take the reconstruction network. We use the flow network trained on the SPIFT dataset to estimate optical flow. We compare our optical flow results with SCFlow (Hu et al. 2022), EV-FlowNet (Zhu and Yuan 2018) and Spike-FlowNet (Lee et al. 2020). Note that SCFlow is designed for spike-based optical flow and trained in SPIFT dataset supervisedly, EV-FlowNet and Spike-FlowNet are designed for event-based optical flow and trained in a self-supervised manner. As shown in Fig. 5, the event-based optical flow methods can not predict correctly optical flow for spike stream. Compared to the other method, our method obtains more clear boundaries and sharper motion regions. Fig. 6 compares the reconstruction results on real-world dataset. We can find that our framework can produce reconstruction results that almost the same qualitative performance as the supervised method. Our method outperforms previous unsupervised reconstruction methods and achieves comparable performance to the supervised method, Spk2ImgNet.

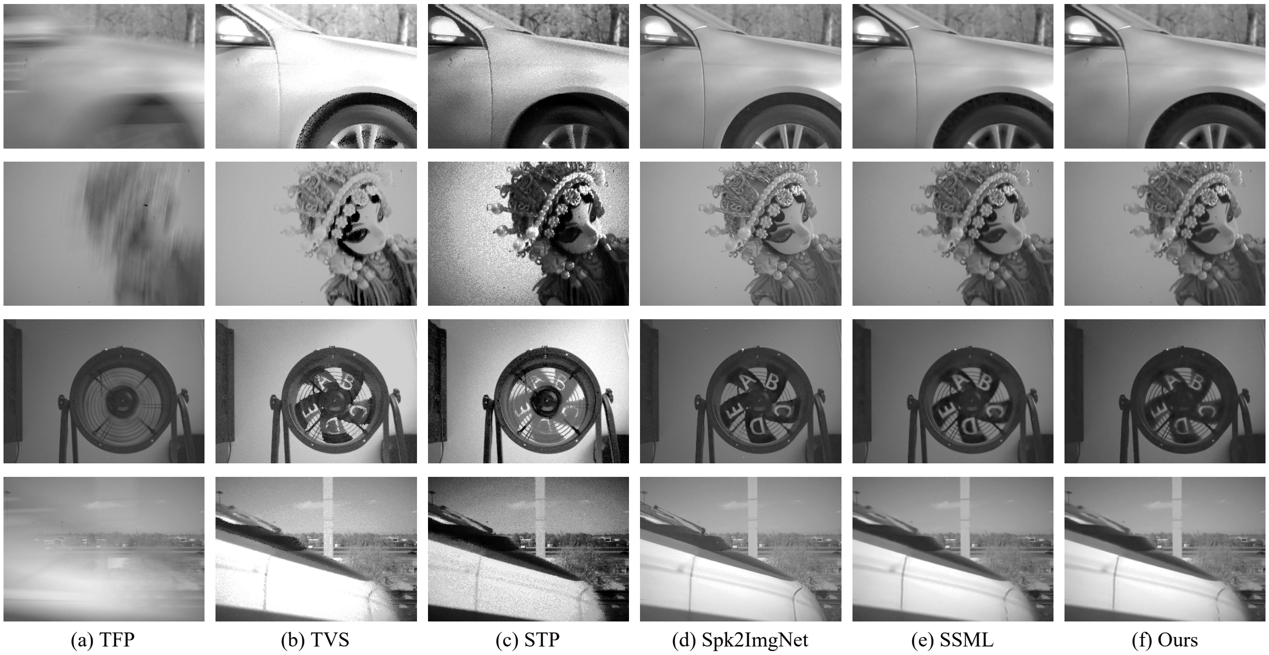


Figure 6: Qualitative Evaluation of reconstruction on Real-World Dataset

Note that our framework is fully trained in a self-supervised manner, thus we can fine-tune our network directly on real-world dataset. As shown in Fig. 5, the fine-tuned model can produce smoother optical flow with sharper motion regions. Our approach can eliminate the dependency on synthetic datasets and can bridge the gap between synthetic and real-world scenes.

Conclusion

In this paper, we present a self-supervised learning framework for dynamic scene reconstruction and optical flow estimation. We employ the self-ensembling within a blind-spot network to improve the self-supervised reconstruction performance. The clean spiking representations from the reconstruction network output are then sent to the optical flow network to predict the optical flow between their corresponding timestamps. In order to achieve self-supervision, we adopt multi-scale photometric loss and propose a novel inverse render loss based on the physical mechanism of blur generation and the long-term spiking temporal aggregation process. Experiments on both synthetic and real-world datasets have demonstrated the effectiveness of the proposed framework. To our best knowledge, this is the first work to obtain desirable reconstruction image and optical flow from the spiking stream in a self-supervised manner. Our work can alleviate the dataset requirement of the spiking camera and promote the practical application of the spiking camera.

Acknowledgments

This work was supported by 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 3 μs Latency Global Shutter Spatiotemporal Vision Sensor. *IEEE Journal of Solid-State Circuits*, 49(10): 2333–2341.

Byun, J.; Cha, S.; and Moon, T. 2021. FBI-Denoiser: Fast Blind Image Denoiser for Poisson-Gaussian Noise. In *CVPR*, 5768–5777.

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 *IJCAI*, 2859–2866.

Choi, J.; Yoon, K.-J.; et al. 2020. Learning to Super Resolve Intensity Images from Events. In *CVPR*, 2768–2776.

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable Convolutional Networks. In *ICCV*, 764–773.

Ding, Z.; Zhao, R.; Zhang, J.; Gao, T.; Xiong, R.; Yu, Z.; and Huang, T. 2022. Spatio-Temporal Recurrent Networks for Event-Based Optical Flow Estimation. In *AAAI*, volume 36, 525–533.

Dong, S.; Huang, T.; and Tian, Y. 2017. Spike Camera and Its Coding Methods. *DCC*.

Dong, S.; Zhu, L.; Xu, D.; Tian, Y.; and Huang, T. 2019. An Efficient Coding Method for Spike Camera using Inter-Spike Intervals. *DCC*.

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.

Gerstner, W.; and Kistler, W. M. 2002. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press.

Hu, L.; Zhao, R.; Ding, Z.; Ma, L.; Shi, B.; Xiong, R.; and Huang, T. 2022. Optical Flow Estimation for Spiking Camera. In *CVPR*, 17844–17853.

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*.

Krull, A.; Buchholz, T.-O.; and Jug, F. 2019. Noise2void-Learning Denoising from Single Noisy Images. In *CVPR*, 2129–2137.

Laine, S.; Karras, T.; Lehtinen, J.; and Aila, T. 2019. High-Quality Self-Supervised Deep Image Dnenoising. *NeurIPS*, 32: 6970–6980.

Lee, C.; Kosta, A. K.; Zhu, A. Z.; Chaney, K.; Daniilidis, K.; and Roy, K. 2020. Spike-FlowNet: Event-Based Optical Flow Estimation with Energy-Efficient Hybrid Neural Networks. In *ECCV*, 366–382. Springer.

Lehtinen, J.; Munkberg, J.; Hasselgren, J.; Laine, S.; Karras, T.; Aittala, M.; and Aila, T. 2018. Noise2Noise: Learning Image Restoration without Clean Data. In *ICML*, 2971–2980.

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

Liu, L.; Zhang, J.; He, R.; Liu, Y.; Wang, Y.; Tai, Y.; Luo, D.; Wang, C.; Li, J.; and Huang, F. 2020a. Learning by analogy: Reliable supervision from transformations for unsupervised optical flow estimation. In *CVPR*, 6489–6498.

Liu, P.; Janai, J.; Pollefeys, M.; Sattler, T.; and Geiger, A. 2020b. Self-Supervised Linear Motion Deblurring. *IEEE Robotics and Automation Letters*, 5(2): 2475–2482.

Masland, R. H. 2012. The Neuronal Organization of the Retina. *Neuron*, 76(2): 266–280.

Paredes-Vallés, F.; and de Croon, G. C. 2021. Back to Event Basics: Self-Supervised Learning of Image Reconstruction for Event Cameras via Photometric Constancy. In *CVPR*, 3446–3455.

Pini, S.; Borghi, G.; and Vezzani, R. 2018. Learn to See by Events: Color Frame Synthesis from Event and RGB Cameras. *arXiv preprint arXiv:1812.02041*.

Posch, C.; Matolin, D.; and Wohlgenannt, R. 2008. An Asynchronous Time-based Image Sensor. In *ISCAS*, 2130–2133.

Rebecq, H.; Ranftl, R.; Koltun, V.; and Scaramuzza, D. 2019a. Events-to-Video: Bringing Modern Computer Vision to Event Cameras. In *CVPR*, 3857–3866.

Rebecq, H.; Ranftl, R.; Koltun, V.; and Scaramuzza, D. 2019b. High Speed and High Dynamic Range Video with an Event Camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Rozumnyi, D.; Oswald, M. R.; Ferrari, V.; Matas, J.; and Pollefeys, M. 2021. Defmo: Deblurring and Shape Recovery of Fast Moving Objects. In *CVPR*, 3456–3465.

Sun, D.; Yang, X.; Liu, M.-Y.; and Kautz, J. 2018. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. In *CVPR*, 8934–8943.

Wässle, H. 2004. Parallel Processing in the Mammalian Retina. *Nature Reviews Neuroscience*, 5(10): 747–757.

Wu, X.; Liu, M.; Cao, Y.; Ren, D.; and Zuo, W. 2020. Unpaired Learning of Deep Image Denoising. In *ECCV*, 352–368. Springer.

Xu, F.; Yu, L.; Wang, B.; Yang, W.; Xia, G.-S.; Jia, X.; Qiao, Z.; and Liu, J. 2021. Motion Deblurring with Real Events. In *ICCV*, 2583–2592.

Zhao, J.; Xiong, R.; Liu, H.; Zhang, J.; and Huang, T. 2021. Spk2ImgNet: Learning to Reconstruct Dynamic Scene from Continuous Spike Stream. In *CVPR*, 11996–12005.

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 *CVPR*, 6358–6367.

Zhu, A. Z.; Thakur, D.; Özaslan, T.; Pfrommer, B.; Kumar, V.; and Daniilidis, K. 2018. The Multivehicle Stereo Event Camera Dataset: An Event Camera Dataset for 3D Perception. *IEEE Robotics and Automation Letters*, 3(3): 2032–2039.

Zhu, A. Z.; and Yuan, L. 2018. EV-FlowNet: Self-Supervised Optical Flow Estimation for Event-based Cameras. In *Robotics: Science and Systems*.

Zhu, L.; Dong, S.; Huang, T.; and Tian, Y. 2019. A Retina-inspired Sampling Method for Visual Texture Reconstruction. In *ICME*, 1432–1437. IEEE.

Zhu, L.; Dong, S.; Li, J.; Huang, T.; and Tian, Y. 2020. Retina-like Visual Image Reconstruction via Spiking Neural Model. In *CVPR*, 1438–1446.