Boosting Spike Camera Image Reconstruction from a Perspective of Dealing with Spike Fluctuations
— Supplementary Materials —

Rui Zhao[1,2] Ruiqin Xiong[1,2*] Jing Zhao[1,2] Jian Zhang[3] Xiaopeng Fan[4] Zhaofei Yu[1] Tiejun Huang[1,2]
[1]School of Computer Science, Peking University
[2]National Key Laboratory for Multimedia Information Processing, Peking University
[3]School of Electronic and Computer Engineering, Peking University
[4]School of Computer Science and Technology, Harbin Institute of Technology
ruizhao@stu.pku.edu.cn
{rqxiong, jzhaopku, zhangjian.sz, yuzf12, tjhuang}@pku.edu.cn fxp@hit.edu.cn

7. Proofs and Introductions

7.1. Proof of theorem 4.1

The following theorem and lemmas are based on the following assumption: There is no thermal noise and Poisson effect, and the input light intensity is constant. We refer to this assumption as "photons' arrival is constant" for simplicity. The analyses in this subsection are ***based on a single pixel***. The meanings of variables in this subsection are shown in Table 5.

Variable	Meaning
T_{r}	The time interval of spike reading.
L	The number of photons reaching a pixel area during a T_{r} time interval.
θ	Accumulation threshold for spike firing in the spike camera.
r	Firing rate of spikes.
$\mathbf{A}_{\text{res}}$	The residual accumulation at each spike-reading moment.
$\mathbf{S}$	Binary spikes.
$\mathbf{D}_{\text{SFT}}^{(1,1)}, D_{\text{SFT}}^{(1,1)}$	(1,1)-order differential of spike firing time (vector and scalar).
$\mathbf{P}$	The set of $\mathbf{D}_{\text{SFT}}^{(1,1)}$ only at moments that a spike is fired.
A_{rpf}	The residual accumulation right after the last previous firing.

Table 5. The meanings of main variables in Sec. 7.1.

Lemma 7.1. *Under the settings in Table 5 and the assumption that the photons' arrival is constant. The expectation of the spike firing rate r is:*

$$\mathbb{E}(r) = L/\theta. \tag{16}$$

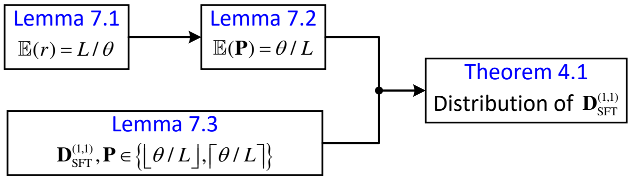

Lemma 7.1
$\mathbb{E}(r) = L / \theta$
Lemma 7.2
$\mathbb{E}(\mathbf{P}) = \theta / L$
Lemma 7.3
$\mathbf{D}_{\text{SFT}}^{(1,1)}, \mathbf{P} \in \{\lfloor \theta / L \rfloor, \lceil \theta / L \rceil\}$
Theorem 4.1
Distribution of $\mathbf{D}_{\text{SFT}}^{(1,1)}$

Figure 11. Proof pipeline of Theorem 4.1.

Proof. Suppose the residual accumulation in the accumulator at each spike-reading moment, i.e., at the end of each integrate-and-fire cycle is $\mathbf{A}_{\text{res}}(nT_{\text{r}})$. Since it has been noted in Sec. 3 that $\theta > L$, $\mathbf{A}_{\text{res}}$ of the pixel follows the following properties:

$$\mathbf{A}_{\text{res}}(nT_{\text{r}}) = \mathbf{A}_{\text{res}}((n-1)T_{\text{r}}) + L - \theta\mathbf{S}(nT_{\text{r}}), \tag{17}$$

where $\mathbf{S}(nT_{\text{r}})$ is 1 if there is a spike at time nT_{r} else it's 0. The average spike firing rate over n spike-reading cycles can be obtained by summing the residual accumulation from T_{r} to nT_{r}:

$$\sum_{k=1}^{n} \mathbf{A}_{\text{res}}(kT_{\text{r}}) = \sum_{k=1}^{n} \mathbf{A}_{\text{res}}((k-1)T_{\text{r}}) + nL - \theta \sum_{k=1}^{n} \mathbf{S}(kT_{\text{r}}), \tag{18}$$

Move all the terms with $\mathbf{A}_{\text{res}}$ to the left of the equal sign:

$$\sum_{k=1}^{n} \mathbf{A}_{\text{res}}(kT_{\text{r}}) - \sum_{k=1}^{n} \mathbf{A}_{\text{res}}((k-1)T_{\text{r}}) = nL - \theta \sum_{k=1}^{n} \mathbf{S}(kT_{\text{r}}), \tag{19}$$

$$\Longrightarrow \mathbf{A}_{\text{res}}(nT_{\text{r}}) - \mathbf{A}_{\text{res}}(0) = nL - \theta \sum_{k=1}^{n} \mathbf{S}(kT_{\text{r}}). \tag{20}$$

[*] Corresponding author.

Divide both sides of the Eq. (20) by n:

$$\frac{\mathbf{A}_{\text{res}}(nT_{\text{r}}) - \mathbf{A}_{\text{res}}(0)}{n} = L - \theta\frac{1}{n}\sum_{k=1}^{n}\mathbf{S}(kT_{\text{r}}). \tag{21}$$

Sort out the Eq. (21) and let $n \to +\infty$, we can obtain:

$$\begin{aligned}\mathbb{E}(r) &= \lim_{n\to+\infty}\frac{1}{n}\sum_{k=1}^{n}\mathbf{S}(kT_{\text{r}}) \\ &= \lim_{n\to+\infty}\frac{\mathbf{A}_{\text{res}}(nT_{\text{r}}) - \mathbf{A}_{\text{res}}(0)}{n\theta} + \frac{L}{\theta} = \frac{L}{\theta}.\end{aligned} \tag{22}$$

Finish proving Lemma 7.1. □

Lemma 7.2. *Under the settings in Table 5 and the assumption that the photons' arrival is constant. Define* $\mathbf{P}$ *is the set of (1,1)-order DSFT with spike firing at the corresponding moment:*

$$\mathbf{P} = \left\{\mathbf{D}_{SFT}^{(1,1)}(kT_r)\right\}_{k\in\{k \mid \mathbf{S}(kT_r)=1\}_{k=1}^{n}}.$$

There is:

$$\mathbb{E}(\mathbf{P}) = \theta/L. \tag{23}$$

Proof. Suppose the total time t_{all} is sufficiently large, it can be represented by either $\mathbb{E}(r)$ or $\mathbf{D}_{\text{SFT}}^{(1,1)}$:

$$t_{\text{all}} = \sum_{k=1}^{n}\Big(\mathbf{S}(kT_{\text{r}})\cdot\mathbf{D}_{\text{SFT}}^{(1,1)}(kT_{\text{r}})\Big), \tag{24}$$

$$t_{\text{all}} = \Big(\sum_{k=1}^{n}\mathbf{S}(kT_{\text{r}})\Big)/\mathbb{E}(r), \tag{25}$$

By considering the above two equations and Lemma 7.1, we can obtain:

$$\frac{\sum_{k=1}^{n}\Big(\mathbf{S}(kT_{\text{r}})\cdot\mathbf{D}_{\text{SFT}}^{(1,1)}(kT_{\text{r}})\Big)}{\sum_{k=1}^{n}\mathbf{S}(kT_{\text{r}})} = \frac{1}{\mathbb{E}(r)} = \frac{\theta}{L}. \tag{26}$$

It is noted that the leftmost term of Eq. (26) seems like an expectation. But it's not the expectation of $\{\mathbf{D}_{\text{SFT}}^{(1,1)}(kT_{\text{r}})\}_{k=1}^{n}$ since the $\{\mathbf{S}(kT_{\text{r}})\}_{k=1}^{n}$ has masked a lot of values of the $\mathbf{D}_{\text{SFT}}^{(1,1)}$.

Define $\mathbf{P}$ is the set of (1,1)-order DSFT with spike firing at the moment:

$$\mathbf{P} = \left\{\mathbf{D}_{\text{SFT}}^{(1,1)}(kT_{\text{r}})\right\}_{k\in\{k \mid \mathbf{S}(kT_{\text{r}})=1\}_{k=1}^{n}}. \tag{27}$$

We can obtain:

$$\mathbb{E}(\mathbf{P}) = \frac{\sum_{k=1}^{n}\Big(\mathbf{S}(kT_{\text{r}})\cdot\mathbf{D}_{\text{SFT}}^{(1,1)}(kT_{\text{r}})\Big)}{\sum_{k=1}^{n}\mathbf{S}(kT_{\text{r}})} = \frac{\theta}{L}. \tag{28}$$

Finish proving Lemma 7.2. □

Lemma 7.3. *Under the settings in Table 5 and the assumption that the photons' arrival is constant. Suppose* $\theta \bmod L \neq 0$. *For one pixel at different moments, the* $\mathbf{D}_{SFT}^{(1,1)}$ *and* $\mathbf{P}$ *have only two values:* $\{\lfloor\theta/L\rfloor, \lceil\theta/L\rceil\}$.

Proof. Suppose the residual accumulation right after the last previous firing is A_{rpf}, which contributes to the current spike firing. It's obvious that when $A_{\text{rpf}} = 0$, the current $D_{\text{SFT}}^{(1,1)}$ is maximum since the previous integration does not help the accumulation of this time. Since $\theta > L$ and $\theta \bmod L \neq 0$, the current (1,1)-order DSFT is $\max(\mathbf{D}_{\text{SFT}}^{(1,1)}) = \lfloor\theta/L\rfloor + 1 = \lceil\theta/L\rceil$.

When $A_{\text{rpf}} \neq 0$, i.e., $A_{\text{rpf}} > 0$, suppose for ***contradiction*** that there are arbitrary two (1,1)-order DSFT satisfy:

$$(D_{\text{SFT}}^{(1,1)})_a - (D_{\text{SFT}}^{(1,1)})_b > 1. \tag{29}$$

Suppose A_{all} is the accumulation before reset when firing spike in this firing cycle, there is:

$$\begin{aligned}&(D_{\text{SFT}}^{(1,1)})_a\cdot L - (D_{\text{SFT}}^{(1,1)})_b\cdot L \\ &= ((A_{\text{all}})_a - (A_{\text{rpf}})_a) - ((A_{\text{all}})_b - (A_{\text{rpf}})_b) \\ &= (A_{\text{rpf}})_b - (A_{\text{rpf}})_a > L.\end{aligned} \tag{30}$$

Thus, there is $(A_{\text{rpf}})_b > (A_{\text{rpf}})_a + L > L$.

Suppose $(D_{\text{SFT}}^{(1,1)})_{\text{last}}$ is the $D_{\text{SFT}}^{(1,1)}$ of the last previous fired spike, it satisfies:

$$(D_{\text{SFT}}^{(1,1)})_{\text{last}}\cdot L \geq \theta, \tag{31}$$

$$\Big((D_{\text{SFT}}^{(1,1)})_{\text{last}} - 1\Big)\cdot L < \theta. \tag{32}$$

From Eq. (32) we can obtain that $(D_{\text{SFT}}^{(1,1)})_{\text{last}}\cdot L < \theta + L$. Furthermore, since $A_{\text{rpf}} = (D_{\text{SFT}}^{(1,1)})_{\text{last}}\cdot L - \theta$, so $A_{\text{rpf}} < L$.

In light of the erroneous assumption Eq. (29), the conclusion is ***inconsistent*** with the assumed objective fact $A_{\text{rpf}} > L$. Thus, there must exist a reality contrary to Eq. (29), i.e., $(D_{\text{SFT}}^{(1,1)})_a - (D_{\text{SFT}}^{(1,1)})_b \leq 1$.

Since $\mathbb{E}(\mathbf{P}) \neq \max(\mathbf{P}) = \max(\mathbf{D}_{\text{SFT}}^{(1,1)})$ and it can only take integers. Thus, there are:

$$\min(\mathbf{P}) = \min(\mathbf{D}_{\text{SFT}}^{(1,1)}) = \max(\mathbf{D}_{\text{SFT}}^{(1,1)}) - 1 = \lfloor L/\theta\rfloor \tag{33}$$

As a result, $\mathbf{D}_{\text{SFT}}^{(1,1)}$ and $\mathbf{P}$ have two values: $\{\lfloor\theta/L\rfloor, \lceil\theta/L\rceil\}$.

Finish proving Lemma 7.3. □

Theorem 4.1. Under the settings in Table 5 and the assumption that the photons' arrival is constant. Suppose $\theta \bmod L \neq 0$. the (1,1)-order DSFT has only two values $\{\lfloor\theta/L\rfloor, \lceil\theta/L\rceil\}$ and its distribution is as follows:

$$\begin{cases}\Pr\left\{\mathbf{D}_{\text{SFT}}^{(1,1)} = \lfloor\theta/L\rfloor\right\} = p_1 = (\lceil\theta/L\rceil - \theta/L)\cdot\dfrac{\lfloor\theta/L\rfloor}{\theta/L} \\ \Pr\left\{\mathbf{D}_{\text{SFT}}^{(1,1)} = \lceil\theta/L\rceil\right\} = p_2 = (\theta/L - \lfloor\theta/L\rfloor)\cdot\dfrac{\lceil\theta/L\rceil}{\theta/L}\end{cases}$$

where $\Pr\{\cdot\}$ means probability.

Proof. Based on Lemma 4.1, since $\mathbf{P}$ has only two values $\{\lfloor\theta/L\rfloor, \lceil\theta/L\rceil\}$, its distribution is as follows:

$$\begin{cases} \Pr\{\mathbf{P} = \lfloor\theta/L\rfloor\} = p_a \\ \Pr\{\mathbf{P} = \lceil\theta/L\rceil\} = p_b = 1 - p_a \end{cases} \tag{34}$$

According to the calculation formula of the expectation and Lemma 7.2, we can obtain:

$$\mathbb{E}(\mathbf{P}) = \lfloor\theta/L\rfloor \cdot p_a + \lceil\theta/L\rceil \cdot p_b, \tag{35}$$

and then we can solve:

$$\begin{cases} \Pr\{\mathbf{P} = \lfloor\theta/L\rfloor\} = p_a = \lceil\theta/L\rceil - \theta/L \\ \Pr\{\mathbf{P} = \lceil\theta/L\rceil\} = p_b = \theta/L - \lfloor\theta/L\rfloor \end{cases} \tag{36}$$

Consider the relationship between $\mathbf{D}_{\text{SFT}}^{(1,1)}$ and $\mathbf{P}$. $\mathbf{P}$ only includes the $\mathbf{D}_{\text{SFT}}^{(1,1)}$ at moments that a spike is fired. But at moments without a spike, the $\mathbf{D}_{\text{SFT}}^{(1,1)}$ still has the same value as the last spike firing moment. Thus, in $\mathbf{D}_{\text{SFT}}^{(1,1)}$, $\lfloor\theta/L\rfloor$ occurs $\lfloor\theta/L\rfloor$ times as often as that in $\mathbf{P}$, and $\lceil\theta/L\rceil$ occurs $\lceil\theta/L\rceil$ times as often as that in $\mathbf{P}$. Thus, If we suppose the distribution of $\mathbf{D}_{\text{SFT}}^{(1,1)}$ is:

$$\begin{cases} \Pr\{\mathbf{D}_{\text{SFT}}^{(1,1)} = \lfloor\theta/L\rfloor\} = p_1 \\ \Pr\{\mathbf{D}_{\text{SFT}}^{(1,1)} = \lceil\theta/L\rceil\} = p_2 \end{cases} \tag{37}$$

We can solve the p_1 and p_2 through:

$$p_1 = \frac{p_a \cdot \lfloor\theta/L\rfloor}{p_a \cdot \lfloor\theta/L\rfloor + p_b \cdot \lceil\theta/L\rceil}, \tag{38}$$

$$p_2 = \frac{p_b \cdot \lceil\theta/L\rceil}{p_a \cdot \lfloor\theta/L\rfloor + p_b \cdot \lceil\theta/L\rceil}. \tag{39}$$

Finally, we can obtain the distribution of $\mathbf{D}_{\text{SFT}}^{(1,1)}$:

$$\begin{cases} \Pr\left\{\mathbf{D}_{\text{SFT}}^{(1,1)} = \lfloor\theta/L\rfloor\right\} = p_1 = (\lceil\theta/L\rceil - \theta/L) \cdot \dfrac{\lfloor\theta/L\rfloor}{\theta/L} \\ \Pr\left\{\mathbf{D}_{\text{SFT}}^{(1,1)} = \lceil\theta/L\rceil\right\} = p_2 = (\theta/L - \lfloor\theta/L\rfloor) \cdot \dfrac{\lceil\theta/L\rceil}{\theta/L} \end{cases} \tag{40}$$

Finish proving Theorem 4.1. □

7.2. Introduction to harmonic mean inequality

Harmonic mean inequality. Suppose $\{a_1, a_2, \ldots, a_n\}$ is a set of positive numbers. There is:

$$\frac{1}{n}\sum_{k=1}^{n} a_k \geq n\left(\sum_{k=1}^{n}\frac{1}{a_k}\right)^{-1}, \tag{41}$$

where the equal sign is only achievable when $a_1, a_2, \ldots, a_n$ are completely equal.

It's a fundamental inequality that can be proved easily based on the convexity of the logarithmic function.

8. Additional Experimental Results and Details

8.1. Additional Details of Comparison Experiments

A. Parameters, running time, and costs.
As shown in Table 6, we compare the proposed BSF and other methods on the number of parameters, computational complexity (MACs), inference time, and GPU memory cost. The inference times are tested on an NVIDIA RTX 3090 GPU with inputs of 256 × 400 resolution.

B. Details of quantitative evaluation.
As shown in Fig. 9 in the main paper, we use 12 scenes for comparison on data captured by spike cameras in the real world. Each scene has 400 frames of spikes. For each scene, we generate the 50th, 60th, ..., and 140th frames for quantitative comparison with non-reference image quality assessment metrics. For each scene of the REDS-SCIR, we use 110 spike frames and 4 image frames for training and evaluation. REDS-SCIR has 8640 and 360 scenes for training and evaluation, respectively. Thus, for training, it has 950.4 k spike frames and 34.56 k images. For evaluation, it has 39.6k spike frames and 1440 images.

C. Number of input frames of comparable methods.
We have compared SSML [1], Spk2ImgNet [5], and WGSE [4] in the experimental parts of the main paper with their original weights and weights retrained with the same settings as ours, respectively. These three methods are designed to use 41 frames of spikes as input, while our method is designed to use 61 frames of spikes as input. For a more rigorous comparison, we modify these three methods to use 61 as input and retrain the modified methods.

For SSML and WGSE, we directly modify the channel number of the input layer to 61. The original version of Spk2ImgNet segments the input 41 frames of spikes to 5 × 13-frame sub-streams with overlap. We modify Spk2ImgNet to use 61 frames of spikes to 5 × 21-frame sub-streams with overlap, and we modify the channel number of the input layer to 21, correspondingly.

The quantitative results of these three methods using 41 or 61 frames are shown in Table. 7. In summary, for existing data-driven methods whose N_{IF} is modified to 61, the proposed BSF still maintains an advantage.

D. Visual results on the SCIR-REDS dataset. We supplement visual results on the REDS-SCIR dataset ($\eta = 0.50$) in Fig. 13 and Fig. 14. For each method in these two figures, we show its reconstructed image and the error between the reconstructed image and its corresponding ground truth. The reconstructed images in the figures are normalized by the quantum factor α and light intensity factor η: $I_{\text{norm}} = I_{\text{pred}}/(\alpha \cdot \eta)$. On the top right corner of the whole figure is the color bar used for the error maps.

Part	Method	Params (M)	MACs (G)	Time (ms)	Memory (GB)	$\eta = 1.00$		$\eta = 0.75$		$\eta = 0.50$	
						PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
(B)	FireNet [2] ♣	0.038	23.54	40.76	1.65	34.38	0.922	33.87	0.911	32.62	0.884
	ETNet [3] ♣	22.179	338.48	243.40	1.87	33.24	0.918	32.85	0.909	31.96	0.889
(C)	SSML [1]	2.385	386.02	408.32	4.19	32.60	0.920	32.09	0.907	31.00	0.879
	SSML [1] ♣					33.94	0.923	33.27	0.909	32.01	0.883
(D)	Spk2ImgNet [5]	3.904	1016.38	167.58	2.68	35.21	0.953	34.70	0.945	33.75	0.926
	Spk2ImgNet [5] ♣					39.16	0.966	38.27	0.958	36.59	0.940
	WGSE [4]	3.806	425.22	88.65	4.78	35.21	0.950	34.98	0.947	34.11	0.931
	WGSE [4] ♣					38.97	0.964	38.23	0.957	36.75	0.940
	BSF (Ours)	2.477	726.89	207.52	4.41	**39.76**	**0.970**	**39.09**	**0.964**	**37.76**	**0.951**

Table 6. The number of parameters, computational complexity, inference time, GPU memory cost, and quantitative performance on the SCIR-REDS dataset of our and other methods. ♣ means the network is retrained on REDS-SCIR with the *same settings* as ours

Method	N_{IF}	$\eta = 1.00$			$\eta = 0.75$			$\eta = 0.50$		
		PSNP ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
SSML [1] ♣	41	33.94	0.923	0.075	33.27	0.909	0.088	32.01	0.883	0.116
	61	34.16	0.922	0.076	33.46	0.909	0.089	32.15	0.883	0.117
Spk2ImgNet [5] ♣	41	39.16	0.966	0.024	38.27	0.958	0.032	36.59	0.940	0.051
	61	39.27	0.970	0.024	38.55	0.961	0.032	37.12	0.946	0.046
WGSE [4] ♣	41	38.97	0.964	0.027	38.23	0.957	0.034	36.75	0.940	0.049
	61	38.91	0.963	0.027	38.17	0.956	0.034	36.76	0.940	0.049
BSF (Ours)	61	39.76	0.970	0.021	39.09	0.964	0.027	37.76	0.951	0.040

Table 7. Performance of data-driven spike camera image reconstruction methods whose number of input spike frames (N_{IF}) is modified to 61 on REDS-SCIR dataset. ♣ means the network is retrained on REDS-SCIR with the *same settings* as ours

MGA			s_{p}	k_{p}	PSNR ↑		
Pym	DCN	CAPA			$\eta = 1.00$	$\eta = 0.75$	$\eta = 0.50$
3	✓	✗	✗	✗	39.380	38.768	37.533
3	✓	✓	3	3	39.759	39.088	37.764
3	✓	✓	5	3	39.704	39.041	37.714
3	✓	✓	7	3	39.693	39.030	37.714
3	✓	✓	9	3	39.700	39.037	37.721

Table 8. Ablation studies on the patchification size s_{p} of the cross-attentional patch-level alignment (CAPA) in multi-granularity alignment (MGA) module.

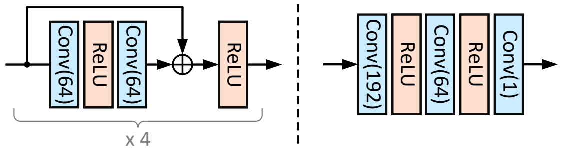
MGA			s_{p}	k_{p}	PSNR ↑		
Pym	DCN	CAPA			$\eta = 1.00$	$\eta = 0.75$	$\eta = 0.50$
3	✓	✗	✗	✗	39.380	38.768	37.533
3	✓	✓	3	3	39.759	39.088	37.764
3	✓	✓	3	5	39.712	39.047	37.726
3	✓	✓	3	7	39.731	39.069	37.749
3	✓	✓	3	9	39.694	39.020	37.685

Table 9. Ablation studies on the local sampling size k_{p} of the cross-attentional patch-level alignment (CAPA) in multi-granularity alignment (MGA) module.

8.2. Additional Results of Ablation Studies

A. Ablations on the patchification size s_{p} of CAPA. In the cross-attentional patch-level alignment (CAPA) of the multi-granularity alignment (MGA) module, features are first patched into patches at $s_{\text{p}} \times s_{\text{p}}$ size without overlap. We implement ablation studies on the s_{p}. As shown in Table. 8, when s_{p} is 3, CAPA achieves its maximum performance gain. When s_{p} is greater than 3, the performance gain decreases. Thus, the $s_{\text{p}} = 3$ we choose is appropriate.

B. Ablations on the local sampling size k_{p} of CAPA. In the CAPA of the MGA module, the key features and value features are locally sampled with $k_{\text{p}} \times k_{\text{p}}$ size to realize a local search strategy. We implement ablation studies on the k_{p}. As shown in Table. 9, the performance of CAPA is insensitive to changes in k_{p}. Thus, we choose $k_{\text{p}} = 3$ that has less computational cost.



(a) Encoder (b) Reconstruction Layer

Figure 12. Illustration of the structure of (a) encoder and (b) reconstruction layer in our method. In parentheses is the number of the output channel.

8.3. Additional Details of Network Architecture

The detailed architecture of the encoder and reconstructive layer in the proposed method are shown in (a) and (b) of

Fig. 12, respectively. The Encoder consists of 4 layers of residual blocks, which are used for embedding the spike representation. The reconstruction we use is similar to that in Spk2ImgNet [5].

9. Fast Algorithm for DSFT

For spike representation, we propose to use multi-order DSFT to reduce the influence of multiple factors that make spikes fluctuate. For the computing of (1,1)-order and higher-order DSFT, we design two fast algorithms with $\mathcal{O}(T)$ complexity, respectively, where T is the spikes' temporal length. We show these two algorithms in a pseudo-code-like manner. Algo. 1 is designed for computing (1,1)-order DSFT, and Algo. 2 is designed for higher-order DSFT.

References

[1] Shiyan Chen, Chaoteng Duan, Zhaofei Yu, Ruiqin Xiong, and Tiejun Huang. Self-supervised mutual learning for dynamic scene reconstruction of spiking camera. In *IJCAI*, pages 2859–2866, 2022. 3, 4

[2] Cedric Scheerlinck, Henri Rebecq, Daniel Gehrig, Nick Barnes, Robert Mahony, and Davide Scaramuzza. Fast image reconstruction with an event camera. In *WACV*, pages 156–163, 2020. 4

[3] Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Event-based video reconstruction using transformer. In *ICCV*, pages 2563–2572, 2021. 4

[4] Jiyuan Zhang, Shanshan Jia, Zhaofei Yu, and Tiejun Huang. Learning temporal-ordered representation for spike streams based on discrete wavelet transforms. In *AAAI*, pages 137–147, 2023. 3, 4

[5] Jing Zhao, Ruiqin Xiong, Hangfan Liu, Jian Zhang, and Tiejun Huang. Spk2ImgNet: Learning to reconstruct dynamic scene from continuous spike stream. In *CVPR*, pages 11996–12005, 2021. 3, 4, 5

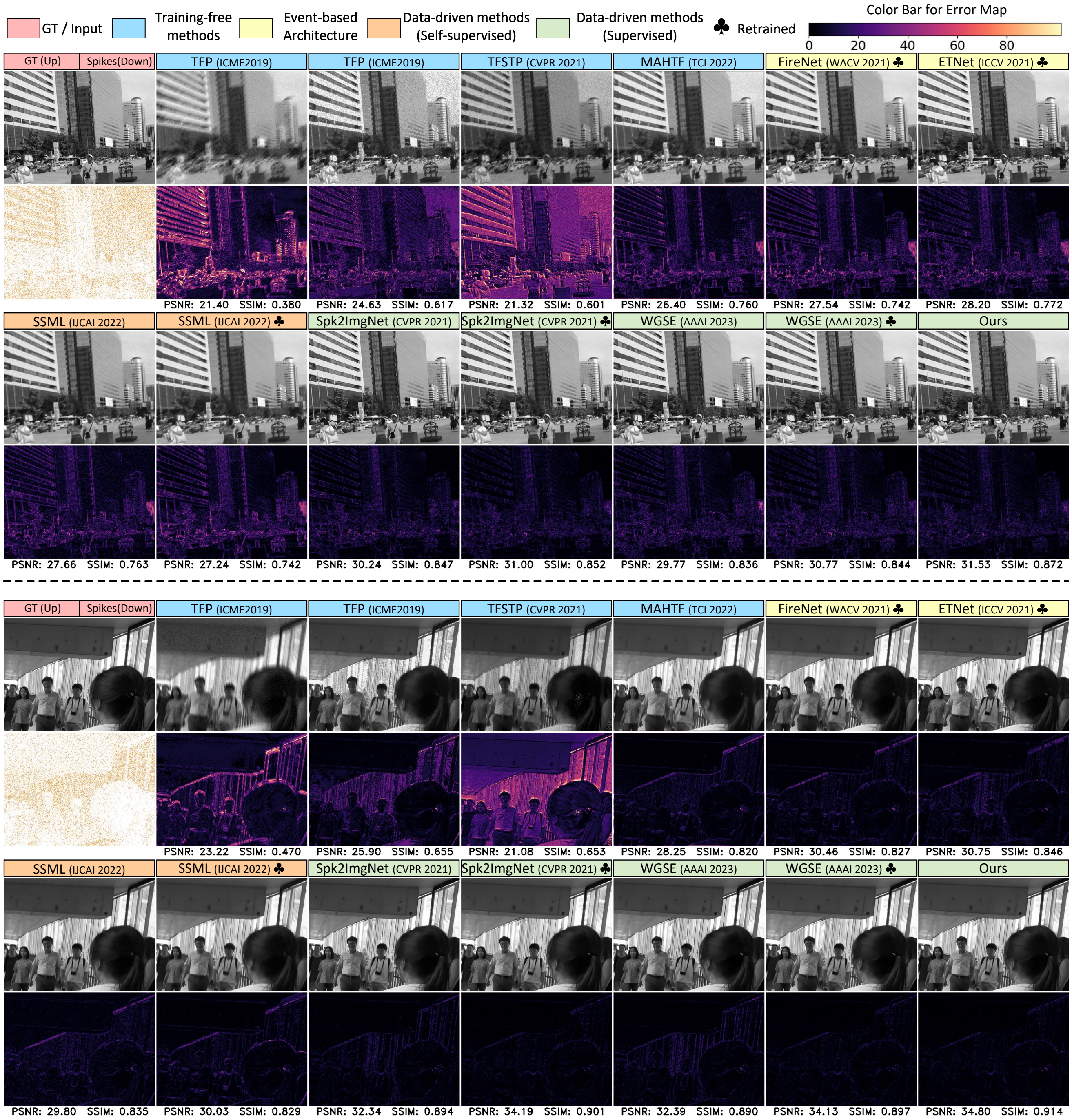


Figure 13. Visual comparison on the REDS-SCIR dataset. In the visualization of spikes, an orange point means a spike. In the visualization of each method, on the top is the reconstructed image, and on the bottom is the error map between the reconstructed image and its ground truth. On the top right corner of the whole figure is the color bar used for error maps. Please zoom in for more details.

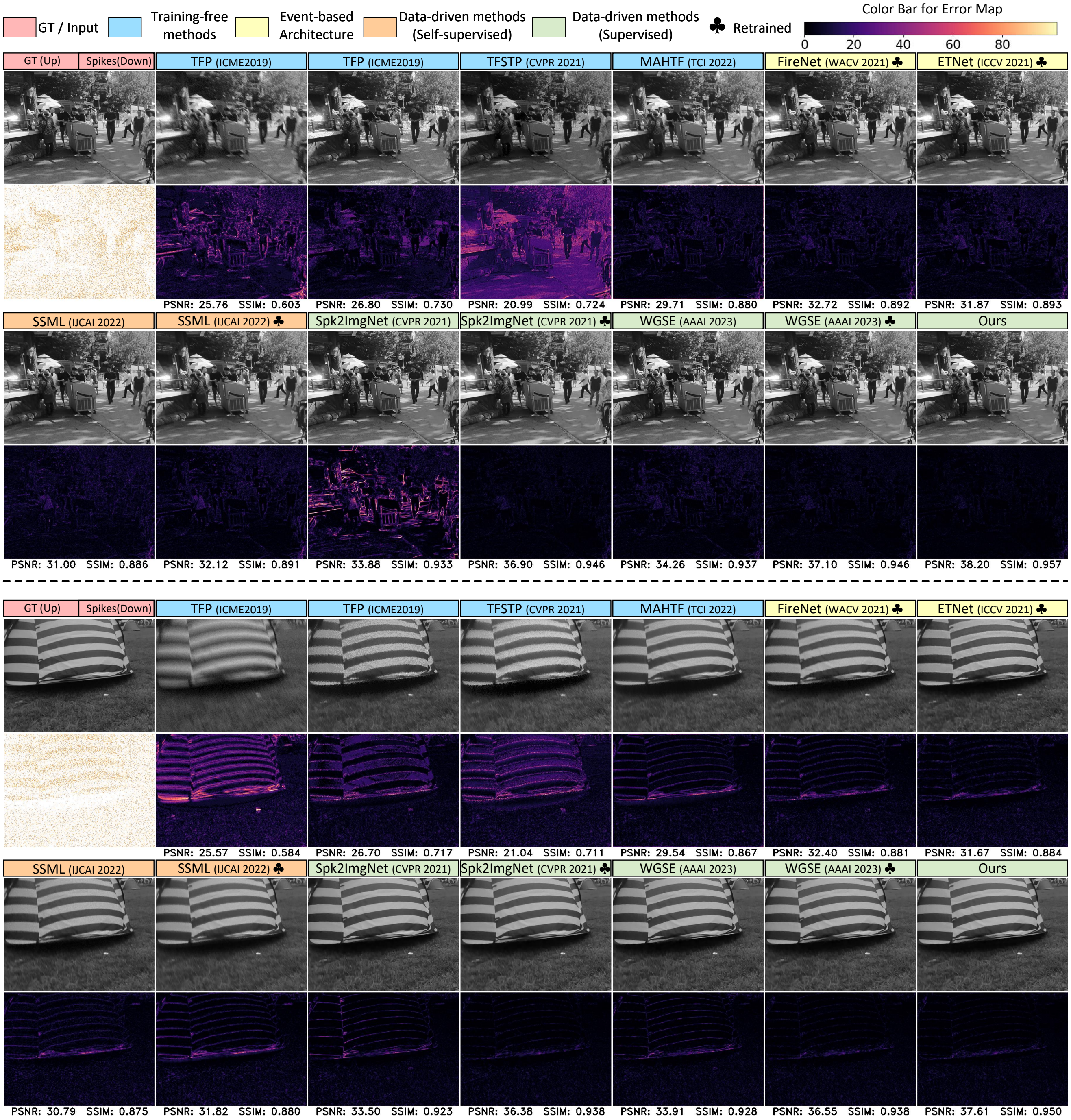


Figure 14. Visual comparison on the REDS-SCIR dataset. In the visualization of spikes, an orange point means a spike. In the visualization of each method, on the top is the reconstructed image, and on the bottom is the error map between the reconstructed image and its ground truth. On the top right corner of the whole figure is the color bar used for error maps. Please zoom in for more details.

Algorithm 1 $\mathcal{O}(T)$ algorithm for (1,1)-order differential of spike firing time (DSFT) $\mathbf{D}_{\text{SFT}}^{(1,1)}$

```
Input:
    The input spike streams $\mathbf{S} \in \mathbb{B}^{T \times H \times W}$.
Output:
    (1,1)-order DSFT $\mathbf{D}_{\text{SFT}}^{(1,1)}$.
Allocate three matrices pre_idx, cur_idx, $\mathbf{D}_{\text{SFT}}^{(1,1)}$ with $T \times H \times W$ size and assign $-1$ to all of their elements;
for $i$ in $1, \ldots, T$ do do
    if $i > 1$ then
        $\texttt{pre\_idx}[i, :, :] = \texttt{cur\_idx}[i-1, :, :]$;
        $\texttt{cur\_idx}[i, :, :] = \texttt{cur\_idx}[i-1, :, :]$;
    end if
    $\texttt{cur\_idx}[i, \mathbf{S}[i, :, :] == 1] = i$;
end for
$\texttt{diff} = \texttt{cur\_idx} - \texttt{pre\_idx}$;
for $i$ in $T, \ldots, 1$ do do
    $\mathbf{D}_{\text{SFT}}^{(1,1)}[i, \texttt{diff}[i, :, :] \neq 0] = \texttt{diff}[i, \texttt{diff}[i, :, :] \neq 0]$;
    if $i < T - 1$ then
        $\mathbf{D}_{\text{SFT}}^{(1,1)}[i, \texttt{diff}[i, :, :] == 0] = \mathbf{D}_{\text{SFT}}^{(1,1)}[i+1, \texttt{diff}[i, :, :] == 0]$;
    end if
end for
return $\mathbf{D}_{\text{SFT}}^{(1,1)}$;
```

Algorithm 2 $\mathcal{O}(T)$ algorithm for higher order differential of spike firing time (DSFT) $\mathbf{D}_{\text{SFT}}^{(n_1,n_2)}$, $n_1 + n_2 > 2$

```
Input:
    The input spike streams and (1,1)-order DSFT: $\mathbf{S} \in \mathbb{B}^{T \times H \times W}$, $\mathbf{D}_{\text{SFT}}^{(1,1)} \in \mathbb{Q}^{+, T \times H \times W}$.
Output:
    higher order DSFT $\mathbf{D}_{\text{SFT}}^{(1,2)}, \mathbf{D}_{\text{SFT}}^{(2,1)}, \mathbf{D}_{\text{SFT}}^{(2,2)}$.
Allocate a matrix flag with $H \times W$ size and assign $-2$ to all the elements;
Allocate two matrices dmls, drls with $T \times H \times W$ size and assign $-1$ to all the elements;
/* dmls: dsft_mask_left_shift, which means the difference between increasing the current dsft to the left by one interval. dmrs: left -> right */
for $i$ in $T, \ldots, 1$ do do
    $\texttt{flag} = \texttt{flag} + (\mathbf{S}[i, :, :] == 1)$;
    $\texttt{cp\_crd} = (\texttt{flag} < 0)$;   /* cp_crd: copy_pad_coordinates */
    $\texttt{dmls}[i, \texttt{cp\_crd} == 1] = \mathbf{D}_{\text{SFT}}^{(1,1)}[i, \texttt{cp\_crd} == 1]$;
    if $i < T - 1$ then
        $\texttt{up\_crd} = (\mathbf{S}[i, :, :] == 1) * (1 - \texttt{cp\_crd})$;   /* up_crd: update_coordinates */
        $\texttt{dmls}[i, \texttt{up\_crd} == 1] = \mathbf{D}_{\text{SFT}}^{(1,1)}[i, \texttt{up\_crd} == 1]$;
        $\texttt{non\_up\_crd} = (\mathbf{S}[i, :, :] \neq 1) * (1 - \texttt{cp\_crd})$;   /* non_up_crd: non_update_coordinates */
        $\texttt{dmls}[i, \texttt{non\_up\_crd} == 1] = \texttt{dmls}[i+1, \texttt{up\_crd} == 1]$;
    end if
end for
Assign -2 to all the elements of flag;
for $i$ in $1, \ldots, T$ do do
    $\texttt{flag} = \texttt{flag} + (\mathbf{S}[i, :, :] == 1)$;
    $\texttt{cp\_crd} = (\texttt{flag} < 0)$;
    $\texttt{dmrs}[i, \texttt{cp\_crd} == 1] = \mathbf{D}_{\text{SFT}}^{(1,1)}[i, \texttt{cp\_crd} == 1]$;
    if $i > 1$ then
        $\texttt{up\_crd} = (\mathbf{S}[i-1, :, :] == 1) * (1 - \texttt{cp\_crd})$;
        $\texttt{dmrs}[i, \texttt{up\_crd}] = \mathbf{D}_{\text{SFT}}^{(1,1)}[i, \texttt{up\_crd} == 1]$;
        $\texttt{non\_up\_crd} = (\mathbf{S}[i-1, :, :] \neq 1) * (1 - \texttt{cp\_crd})$;
        $\texttt{dmrs}[i, \texttt{non\_up\_crd} == 1] = \texttt{dmrs}[i-1, \texttt{up\_crd} == 1]$;
    end if
end for
$\mathbf{D}_{\text{SFT}}^{(1,2)} = \mathbf{D}_{\text{SFT}}^{(1,1)} + \texttt{dmls}$;
$\mathbf{D}_{\text{SFT}}^{(2,1)} = \mathbf{D}_{\text{SFT}}^{(1,1)} + \texttt{dmrs}$;
$\mathbf{D}_{\text{SFT}}^{(2,2)} = \mathbf{D}_{\text{SFT}}^{(1,1)} + \texttt{dmls} + \texttt{dmrs}$;
return $\mathbf{D}_{\text{SFT}}^{(1,2)}, \mathbf{D}_{\text{SFT}}^{(2,1)}, \mathbf{D}_{\text{SFT}}^{(2,2)}$;
```