Supplementary Materials: Spike-guided Motion Deblurring with Unknown Modal Spatiotemporal Alignment

Jiyuan Zhang[1,2] Shiyan Chen[1,2] Yajing Zheng[1,2*] Zhaofei Yu[1,2,3*] Tiejun Huang[1,2,3]
[1]School of Computer Science, Peking University
[2]National Key Laboratory for Multimedia Information Processing, Peking University
[3]Institute for Artificial Intelligence, Peking University
{jyzhang,2301112005}@stu.pku.edu.cn, {yj.zheng,yuzf12,tjhuang}@pku.edu.cn

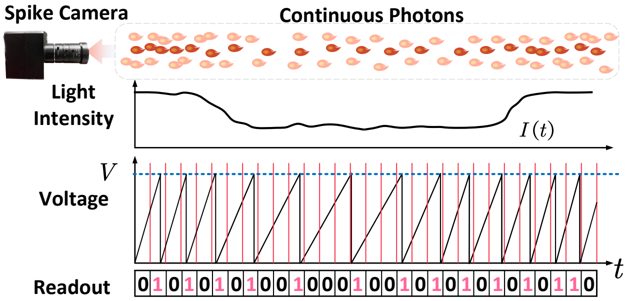

Figure 1. Illustration of the spike generation principle.

1. Illustration of Spike Generating Process

The detailed spike-generating process of a spike camera is presented in Fig. 1. The pixel array on a spike camera independently receives continuous photons. As shown in the figure, the photon arriving velocity reflects the light intensity. The voltage is always increased by the electrical current converted from photons, and reset whenever reaching the threshold. The back circuit reads out spike signals in the very short interval τ.

2. Details of Network Structures

In the first stage, we adopt NAFNet [2] as the basic image deblurring module M_1. The overall network channels are halved from 64 to 32.

In the second stage, the rough grayscale intensity $\tilde{\mathbf{L}}_{\mathbf{S}}$ is inferred by the shallow convolutional module M_{S2I} with spike $\mathbf{S}$ as input. M_{S2I} simply consists of 6 Conv2d layers with the middle channel of 128. In the stage-2 model M_2, spike features $\mathbf{F}_{\mathbf{S}} = \{F^h_{\mathbf{S}}\}$ and image features $\mathbf{F}_{\mathbf{L}} = \{F^h_{\mathbf{L}}\}$, where $h \in [1, 2, ..., H]$ are obtained by the convolutional feature extractors. H is set to 3. The first extractor consists of a head Conv layer and a residual Conv block, while others consist of two consecutive Conv layers. Feature channels in stage 2 are 64.

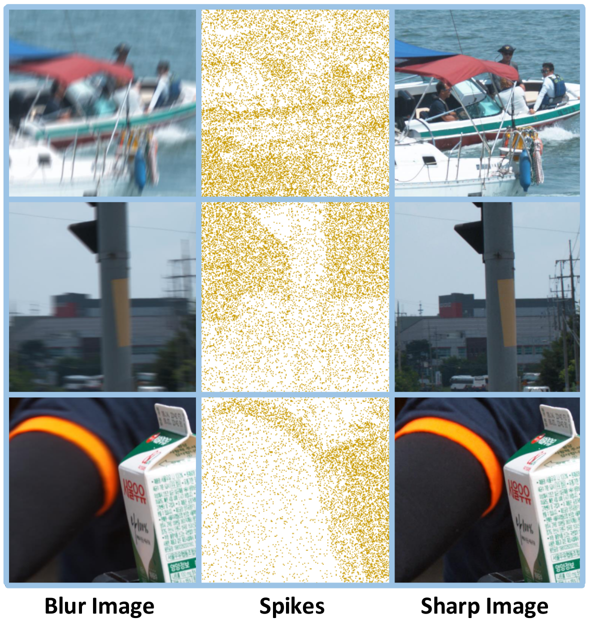

Figure 2. Samples in the X4K1000FPS with unaligned spikes.

In the third stage, each encoder and decoder is an attention-base dense block (ADB) consisting of a channel attention layer (CA) and a residual dense block (RDB). There are 9 ADBs for encoding, 9 ADBs for decoding, and 1 ADB between the encoding and decoding modules. There is a Conv layer for downsampling after the 3rd, 6th, and 9th encoder. A RDB [7] contains 6 Conv layers and the growing rate is 32.

3. Details of Datasets

To provide a clearer description of datasets, Fig. 2 presents visualizations of the spatiotemporally misaligned

[*] Corresponding author.

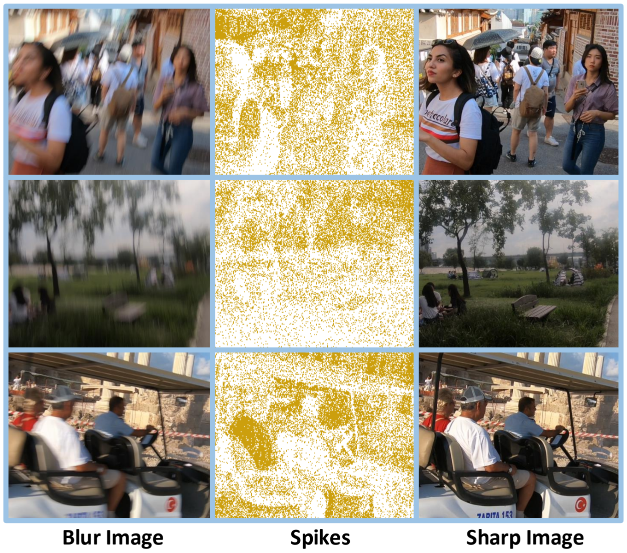

Figure 3. Samples in the REDS with unaligned spikes.

Method	Input Data	PSNR ↑	SSIM ↑
HINet [1]	Image	31.10	0.902
NAFNet [2]	Image	30.73	0.894
EFNet [4]	Image+Spike	32.79	0.926
REFID [5]	Image+Spike	33.63	0.926
UaSDN(Ours)	Image+Spike	**34.58**	**0.945**
UaSDN-A(Ours)	Image+Spike	**35.55**	**0.955**

Table 1. Comparison of various motion deblurring methods on REDS [3] when spikes and images are precise aligned.

spikes generated using the X4K1000FPS dataset. In X4K1000FPS, 66,120 samples are used for training, and 105 samples are used for testing. Fig. 3 presents visualizations of REDS datasets with spikes. In REDS with spikes, 23,520 samples are used for training, and 2940 samples are used for testing. (Spikes are rendered with color in the figures for better visualization.)

4. Training Details

Models are trained with PyTorch on 2 NVIDIA GeForce RTX 4090 with a batch size of 8. We use random cropping, random flipping, and random rotation($90°, 180°, 270°$) as data augmentations. The batch size is 256 × 256 and 128 × 128 for images and spikes, respectively. AdamW is adopted as the optimizer with a learning rate of 1e-4 and weight decay of 1e-4. We train all models for 100,000 iterations with the cosine scheduler. The lower bound of the learning rate is set to 1e-6.

Method	Input Data	PSNR ↑	SSIM ↑
EDVR [6]	Image	34.80	0.949
UaSDN(+EDVR)	Image+Spike	**36.80**	**0.966**

Table 2. Comparison with the SOTA video-based method on REDS [3].

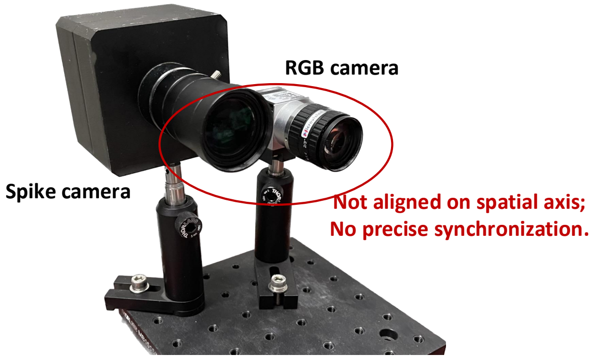

Figure 4. Illustration of the built simple spike-RGB camera system for capturing real-world scenes.

5. Comparison Results on REDS under Precise Alignment

In the ablation experiments of the manuscripts, we test the performance advantages of our method when spikes and images are strictly aligned on the X4K1000FPS dataset. In this section, we present the experimental results of the REDS dataset. As shown in Tab. 1, compared to the other four methods, our approach achieves a PSNR of 34.58dB and an SSIM of 0.945. Compared to the retrained EFNet and REFID methods with spikes as auxiliary information, our method significantly improves by 1.70dB and 0.95dB.

In addition, we consider that when spikes and images are aligned, our network structure can be simplified by removing the modules specifically used for feature alignment to lighten the network and focus on learning scene textures from spikes. Specifically, we remove the first and second stages of the model and eliminate the flow alignment part of the third stage, resulting in a lightweight version named UaSDN with Alignment (UaSDN-A). As shown in Tab. 1, UaSDN-A achieves better results when aligned, specifically a PSNR of 35.55dB and an SSIM of 0.955. The ablation result demonstrates that our method can achieve optimal performance on motion deblurring whether the modalities are aligned or not.

Discussion on comparison with video-based method Currently, in image restoration tasks, state-of-the-art (SOTA) methods often rely on using consecutive frames in a video to provide more visual information as deblurring clues. On the REDS dataset, one of the SOTA meth-

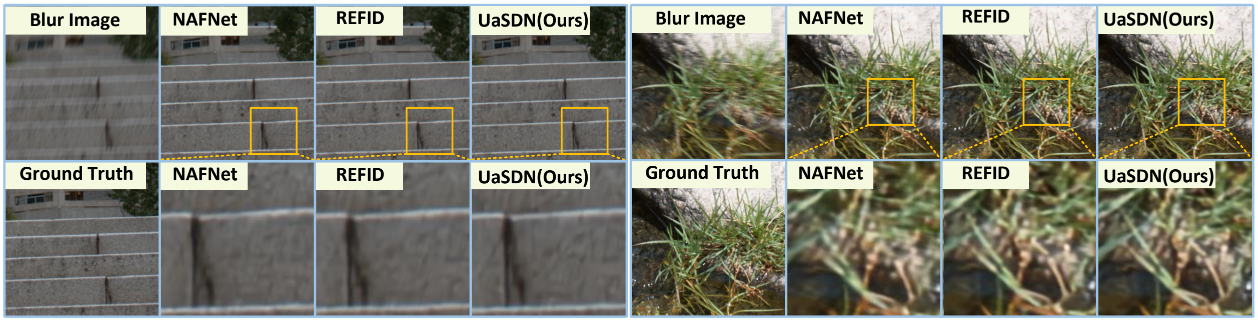

Figure 5. Visualized results under the situation of the precise alignment. The comparison of our method (UaSDN) on X4K100FPS dataset with four other methods: HINet, NAFNet, EFNet, and REFID.

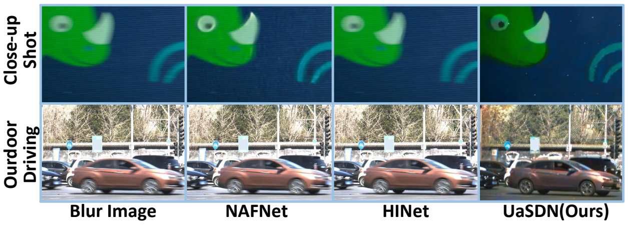

Figure 6. Visualized results on real-world data captured with an RGB camera and a spike camera.

ods is EDVR [6], which uses a sequence of 5 consecutive blurry frames as input. However, in practical applications like driving scenarios, such methods are not feasible to have access to images at time T_{i+1}, T_{i+2}, etc., when deblurring the image $\mathbf{B}_i$ at time T_i. Therefore, methods relying on multi-frame inputs are impractical in real-world scenarios. Nevertheless, To demonstrate the effectiveness of our spike-guided deblurring scheme, we replaced the first stage of the proposed UaSDN with the EDVR network and trained the complete UaSDN on REDS. The experimental results are shown in Tab. 2. Compared to EDVR, out UaSDN improves the performance of deblurring on REDS dataset and reaches the PSNR of 36.78dB and the SSIM of 0.966. The significant improvement in PSNR is 2.00 dB, which demonstrates that our method can effectively utilize information in spikes to assist motion deblurring.

6. Performance on Real World Spikes

To assess the deblurring capability of our method in real-world scenarios, we set up a simple spike-RGB hybrid camera system, as illustrated in Fig. 4. In the setup, no strict spatial alignment is required between the spike camera and the RGB camera, and precise synchronization in time is not necessary, making the system easy to deploy. Fig. 6 presents more real-world test cases. The real-world test demonstrated the generalization ability of our model.

K	1	2	3	4	5
PSNR(dB)	34.97	**35.78**	35.81	35.92	35.84
InferTime(ms)	80.4	**90.7**	101.1	111.6	123.0

Table 3. Performance with different number of spike segments

7. Visualized Results in Precise Alignment.

Fig.5 illustrates the visual comparison results in precise alignment, demonstrating that our model better restores scene details compared to other methods.

8. Ablation study on the number of spike segments.

We thoroughly test the number of spike segments $\mathbf{K} \in [1, 2, 3, 4, 5]$. Tab. 3 shows that all models perform well in terms of PSNR and inference time. The PSNR increases from K=1 to 2, but for K>2, the increase was minimal while the computational overhead increased. Therefore, we choose K=2 as the final setting.

References

[1] Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen. Hinet: Half instance normalization network for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 182–192, 2021. 2

[2] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European Conference on Computer Vision*, pages 17–33. Springer, 2022. 1, 2

[3] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019. 2

[4] Lei Sun, Christos Sakaridis, Jingyun Liang, Qi Jiang, Kailun Yang, Peng Sun, Yaozu Ye, Kaiwei Wang, and Luc Van Gool.

Event-based fusion for motion deblurring with cross-modal attention. In *European Conference on Computer Vision*, pages 412–428. Springer, 2022. 2

[5] Lei Sun, Christos Sakaridis, Jingyun Liang, Peng Sun, Jiezhang Cao, Kai Zhang, Qi Jiang, Kaiwei Wang, and Luc Van Gool. Event-based frame interpolation with ad-hoc deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18043–18052, 2023. 2

[6] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019. 2, 3

[7] 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*, 43(7):2480–2495, 2020. 1