

# Event-guided Deblurring of Unknown Exposure Time Videos

Taewoo Kim<sup>1</sup>, Jeongmin Lee<sup>1</sup>, Lin Wang<sup>2</sup>, and Kuk-jin Yoon<sup>1</sup>

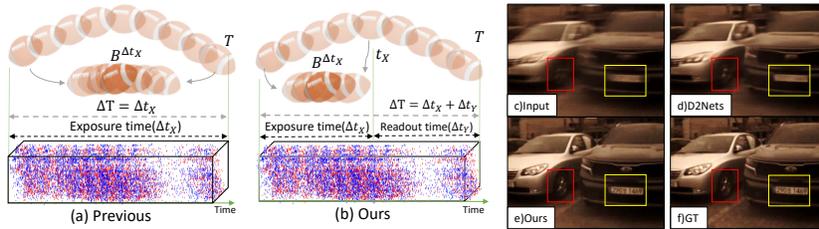
<sup>1</sup> Korea Advanced Institute of Science and Technology  
{intelpro, jeanmichel, kjyoon}@kaist.ac.kr

<sup>2</sup>AI Thrust, HKUST Guangzhou and Dept. of CSE, HKUST  
linwang@ust.hk

**Abstract.** Motion deblurring is a highly ill-posed problem due to the loss of motion information in the blur degradation process. Since event cameras can capture apparent motion with a high temporal resolution, several attempts have explored the potential of events for guiding deblurring. These methods generally assume that the exposure time is the same as the reciprocal of the video frame rate. However, this is not true in real situations, and the exposure time might be unknown and dynamically varies depending on the video shooting environment (e.g., illumination condition). In this paper, we address the event-guided motion deblurring assuming dynamically variable unknown exposure time of the frame-based camera. To this end, we first derive a new formulation for event-guided motion deblurring by considering the exposure and readout time in the video frame acquisition process. We then propose a novel end-to-end learning framework for event-guided motion deblurring. In particular, we design a novel Exposure Time-based Event Selection (ETES) module to selectively use event features by estimating the cross-modal correlation between the features from blurred frames and the events. Moreover, we propose a feature fusion module to fuse the selected features from events and blur frames effectively. We conduct extensive experiments on various datasets and demonstrate that our method achieves state-of-the-art performance. Our project code and dataset are available at: <https://intelpro.github.io/UEVD/>

## 1 Introduction

Motion blur often occurs due to the non-negligible exposure time of the frame-based cameras. Any motion during the video recording makes the sensor observe an averaged signal from different points in the scene [44,23]. Motion deblurring is a task aiming at restoring sharp frame from the motion-blurred ones. This task is a highly ill-posed problem due to the loss of motion information in the blur degradation process, especially in the complex real-world scene [6,1,9]. Recently, deep learning (DL)-based approaches have achieved great success in modeling general motion blur and recovering sharp frames from the motion-blurred frames [49,20,33,55]. However, they are limited to specific scenarios and may fail to recover the sharp frames for the severe motion blur. Event cameras are



**Fig. 1.** (a-b) The motivation of our work. The figure shows the image formation setting assumed in (a) previous event-guided methods and (b) our work. Here,  $\Delta T$  represents the shutter period. From above to bottom, continuous latent images, video frame acquisition, corresponding event streams ( $E^{\Delta T}$ ), respectively. Previous methods generally assume that the exposure time is the same as the reciprocal of the video frame rate, while ours considers the actual exposure time for selectively using events. (c-f) Results of deblurring on the unknown exposure time video frame. Our method restores a sharper frame than the existing event-guided deblurring methods, *e.g.*, D2Nets [32], trained on the same dataset.

bio-inspired sensors that encode the per-pixel intensity change asynchronously with high temporal resolution.

Many endeavors have been engaged in reconstructing image/video from event streams [31,38,19,41,59,40,39]. However, the reconstructed results from the events may lose texture details. Consequently, several attempts have leveraged events for guiding motion deblurring [17,9,53,27,38,32], trying to take advantage of frame-based and event cameras. As shown in Fig. 1(a), these methods generally assume that the exposure time is the same as the reciprocal of the video frame rate and perform the deblurring guided by the events within the exposure time. However, this assumption may not be valid in real situations as the video frame acquisition process generally consists of two phases: exposure phase ( $X$ ) and readout phase ( $Y$ ) [5,55], as shown in Fig. 1(b). In the exposure phase, the camera’s shutter opens and receives lights. In the readout phase, the camera clears charge from the serial register, and the pixel value is digitalized. The total time, reciprocal of the frame rate, is called the shutter period, not exposure time. Since the motion blur of the frame-based camera occurs only in the exposure phase rather than the readout phase, it is crucial to use the events during the exposure phase within the shutter period. However, the exposure time is not always known, and furthermore, it can dynamically vary depending on the imaging environments when the auto-exposure function turns on.

For that reason, we assume the exposure time is unknown when performing the event-guided motion deblurring, as shown in Fig. 1(b), to consider more practical situations. This assumption can lead to significant performance changes. If we apply the existing event-guided deblurring methods, *e.g.*, [32], for unknown exposure time video frames, the performance degrades, as shown in Fig. 1(d).<sup>1</sup>

<sup>1</sup> This result is obtained by using all the events during the shutter period without knowing the actual exposure time.

Therefore, it is necessary to infer the actual exposure time to use events rightly for deblurring. Accordingly, we propose an end-to-end learning framework. As the exposure time is assumed to be unknown, we first propose an event selection module called the Exposure Time-based Event Selection(ETES). The proposed module extracts the relevant events within the shutter period by estimating the temporal correlation between events and blur frames features. As such, only the event features corresponding to the exposure time are automatically selected for guidance. Second, we propose a new module for events-frame feature fusion. Such a fusion module leads to more robust feature representation learning. Lastly, as a lack of publically available real-world event datasets for event-guided motion deblurring, we collect color images and event data in diverse real-world scenes using a DAVIS-346 color event camera. We then make a dataset by simulating dynamically variable exposure time using collected frames and real events for performing motion deblurring without exposure time information.

In summary, our contributions are four-fold. (I) We study and formulate an event-guided deblurring for unknown exposure time videos. (II) Based on the formulation, we design a novel event feature selection method within exposure time and propose a feature fusion module to use complementary information of events and frame features. (III) We build a novel large-scale dataset for event-guided motion deblurring, including RGB images and the real events in various scenes. (IV) We conduct various experiments on the synthetic event and our real-world event datasets and demonstrate that our method achieves new state-of-the-art performance.

## 2 Related Works

**Image and Video Deblurring.** DL has been broadly applied to image and video deblurring. Earlier works, *e.g.*, [34], utilized convolutional neural networks (CNNs) with frame alignment and merged multiple frames based on the homography for video deblurring. The baseline networks have been improved by applying more sophisticated network structures or learning methods, *e.g.*, recurrent neural networks (RNN) [44,7,36,21,56,28], multi-scale architecture [20,2,4], adversarial training [50,14,15,51,52], multi-stage approaches [49,48,3], video frame alignment [11,57,42,24,16] in an end-to-end learning manner.

**Event-guided Motion Deblurring.** The event cameras show higher temporal resolution and HDR properties. To leverage the advantages of event cameras, recent works focus on event-guided deblurring. Pan *et al.* [27,26] first proposed a deblurring framework by formulating an event-based double integral model(EDI). Although they show the effectiveness of formulation, they often fail to reconstruct the scene details due to the noisy contrast threshold of the events. To solve the aforementioned issues, Jiang *et al.* [9] introduced a DL-based deblurring framework by using an RNN-based network architecture and a directional event filtering module. More recently, Lin *et al.* [17] proposed a CNN-based framework driven by an event-based physical model for deblurring and frame interpolation. Shang *et al.* [32] proposed an event-guided deblurring

framework to exploit the non-consecutively blurry video frames. Concurrently, Xu *et al.* [47] proposed a self-supervised learning framework that utilizes real events to alleviate performance degradation due to the domain gap between real and synthetic data. These works generally assume the exposure time is the same as the shutter period. However, as aforementioned, this assumption is not valid in many real situations. *Unlike these works, we propose a novel framework for unknown exposure time videos.*

### 3 Method

#### 3.1 Formulation

**Event Selection** A video frame acquisition consists of the exposure phase  $X$  and readout phase  $Y$ , as depicted in Fig. 1. We denote the duration of the exposure phase as  $\Delta t_X$  and the readout phase as  $\Delta t_Y$ . The summation of two phases (shutter period),  $\Delta T$ , represents the time to acquire one video frame. By the nature of frame-based cameras, motion blur only occurs during the exposure phase  $X$ . In contrast, events are generated in the exposure phase  $X$  and the readout phase  $Y$ . Therefore, it is imperative to use events during the exact duration of the exposure phase  $\Delta t_X$  only. The existing event-guided deblurring methods [27,17] generally assume that the exposure time is equal to the shutter period;  $\Delta T = \Delta t_X$ . By contrast, our goal is to estimate the temporal correlation between the motion-blurred frame and the event during  $\Delta T$  to handle the unknown exposure time  $\Delta t_X$ . A motion-blurred frame can be expressed as the temporal average of  $N$  latent frames during the exposure time  $\Delta t_X$  as

$$B^{\Delta t_X}(x, y) \simeq \frac{1}{N} \sum_{i=1}^N L_{\tau_i}(x, y), \quad (1)$$

where  $B^{\Delta t_X}(x, y)$  denotes a blurred frame, and  $L_{\tau_i}(x, y)$  is the  $i$ -th latent frame at  $\tau_i$ . For event cameras, an event is generated when the log intensity change exceeds a contrast threshold  $\beta$ .

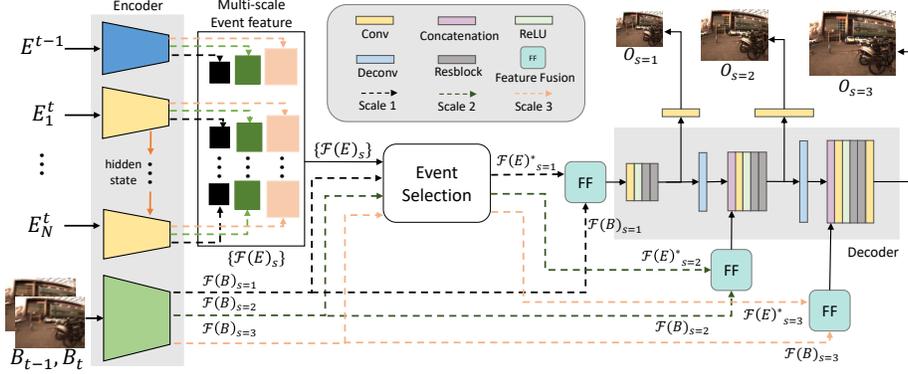
$$E^t(x, y) = \begin{cases} +1, & \text{if } \log\left(\frac{I^t(x, y)}{I^{t-1}(x, y)}\right) \geq \beta \\ -1, & \text{if } \log\left(\frac{I^t(x, y)}{I^{t-1}(x, y)}\right) \leq -\beta \end{cases} \quad (2)$$

where  $I^t(x, y)$  is the intensity value at timestamp  $t$ . Given two consecutive frames,  $I^{t_1}(x, y)$  and  $I^{t_2}(x, y)$ , the events  $E^t(x, y)$  are generated by intensity changes between them. Accordingly, we can derive the relationship between two intensity images based on event generation.

$$I^{t_2}(x, y) \simeq I^{t_1}(x, y) \cdot \exp\left(\sum_{t_1}^{t_2} \beta \cdot E^t(x, y)\right) = I^{t_1}(x, y) \tilde{R}(x, y) \quad (3)$$

By combining Eq.(1) and Eq.(3), we can represent a blurred frame as follows:

$$B^{\Delta t_X}(x, y) \simeq I^t(x, y) \left(\frac{1}{N} \sum_{i=1}^N \tilde{R}_i(x, y)\right) = I^t(x, y) S(x, y) \quad (4)$$



**Fig. 2.** Overview of the proposed framework. For the encoder, blue, yellow, and green boxes represent an event encoder for the past part, a shared RNN-based event encoder for the current part, and a blur-frame encoder, respectively.

where  $I^t(x, y)$  denotes a latent frame, which is the result of deblurring, and  $S(x, y)$  is the summation of residual matrix  $\tilde{R}_i(x, y)$ . According to Eq.(4), we need to estimate a  $S(x, y)$  corresponding to the exposure time of the blurred frame. However, since the exact exposure time is assumed to be unknown, we aim at estimating the temporal correlation between the blurred frames and events during  $\Delta T$ . That is, to estimate  $S(x, y)$ , we find a function  $f_{\theta^*}$  that approximates a set of events  $\{E^{\Delta t_x}\}$  based on  $\{E^{\Delta T}\}$  and the blurred frame  $B^{\Delta t_x}$  as

$$f_{\theta^*} \sim \{E^t \mid \psi(E^t, B^{\Delta t_x}) > 0\}; \quad t \in \{0, T\} \quad (5)$$

where  $\psi(E^t, B^{\Delta t_x})$  is a conditional function obtained by calculating the temporal correlation between  $E^t$  and  $B^{\Delta t_x}$ . With respect to  $E^t$ , a function  $\psi(E^t, B^{\Delta t_x})$  is true when the intensity change corresponding to  $E^t$  exists in the motion-blurred frame  $B^{\Delta t_x}$  and vice versa. As such, only the events within the exposure time are selected as guidance.

**Event-guided deblurring** Through the above formulation, we can select event streams  $\{E^{\Delta t_x}\}$  among  $\{E^{\Delta T}\}$  for the unknown exposure time  $\Delta t_x$ . Accordingly, we aim at recovering an intermediate sharp video frame from the motion-blurred ones  $B^{\Delta t_x}$  using  $\{E^{\Delta t_x}\}$ .

### 3.2 The Proposed Framework

Based on the formulation, we propose a novel end-to-end learning framework. To feed event streams to DNNs, we need to embed them to the fixed size tensor-like format. The voxel grid [58] is a well-designed event representation as it preserves the spatio-temporal information of events. We use 16 temporal bins of the voxel grid for all the experiments.

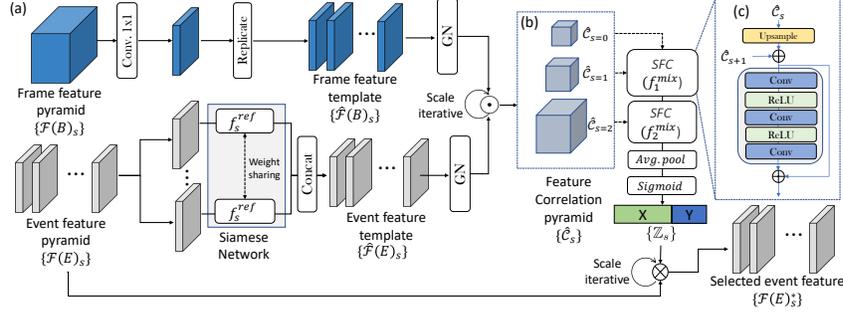
**Overview** An overview of the proposed framework is illustrated in Fig. 2.

The network uses the past and current inputs together to reinforce the spatio-temporal dependency of the videos. The overall framework consists of *two* major components: *event selection* and *feature fusion*. First, we encode the embedded events via an RNN-based encoding network for event feature selection. Then, we propose a novel ETES module to select the event features corresponding to the unknown exposure time *without any supervision*. Second, in Sec. 3.4, we propose a new feature fusion module that efficiently exploits the complementary information of selected events and frames. After the two steps, our network processed fused feature in a coarse-to-fine manner using a pyramid structure.

### 3.3 Event Selection

**Recurrent Encoding for Embedded Events.** To extract features from events, recent works widely adopt 2D CNNs [58,17]; however, they are less effective in preserving temporal information for the event selection under unknown exposure time (Eq.5). Specifically, we first divide the events into the past shutter period part  $\{E^{t-1}\}$  and the current shutter period part  $\{E^t\}$ . We then use the current shutter-period events to infer unknown exposure time, and past information is only used for better deblurring. Therefore, we use different event encoders to allocate more channels to the current events for correctly estimating exposure time. For the past part  $\{E^{t-1}\}$ , we extract a feature pyramid  $\{\mathcal{F}(E^{t-1})_s\} \in \mathbb{R}^{C_s^U \times H_s \times W_s}$ , using a 2D CNN block with the scale index  $s \in \{1, 2, 3\}$ . Especially for the current part  $\{E^t\}$ , we devise a new RNN-cell based event encoder with shared weights to extract features for preserving temporal information inspired by recent frame-based video deblurring works [56,21] as shown in Fig. 2. Considering the temporal information of the current part, we first divide the voxel grid into  $N$  temporal units with an equal time interval  $\Delta t_{unit}$ . Thus, we get temporally divided event units  $E_n^t \in \mathbb{R}^{2 \times H \times W}$  with the temporal index  $n \in \{1, \dots, N\}$ . We then recursively update the hidden state of the RNN cell to reinforce temporal coherence between consecutive event units. As such, we generate  $N$  hierarchical feature maps at each scale index  $s$  for the current part  $\{\mathcal{F}(E_1^t)_s, \dots, \mathcal{F}(E_N^t)_s\}$ . The extracted features of the current and past parts are concatenated to form a feature pyramid, denoted by  $\{\mathcal{F}(E)_s\} \in \mathbb{R}^{(N+1) \times C_s^U \times H_s \times W_s}$ , where  $C_s^U, H_s, W_s$  denotes the numbers of unit channels, height, and width at scale index  $s$ , respectively. For brevity, we denote  $N + 1$  as  $T$ . The detailed network structures and RNN-based encoding methods are given in the supple. material.

**Exposure Time-based Event Selection (ETES) Module.** Through encoding, we obtain the event and frame feature pyramids  $\{\mathcal{F}(E)_s\}$  and  $\{\mathcal{F}(B)_s\}$ , respectively. To approximate Eq.(5), we now aim to select the beneficial event features corresponding to the dynamically varying unknown exposure time of the frame-based camera. However, there exist two crucial challenges. The first one is how to pre-process event features (with complete temporal information) and frame features (with missing temporal information). The second is how to better discover the cross-modal relationship between events and frames by aggregating



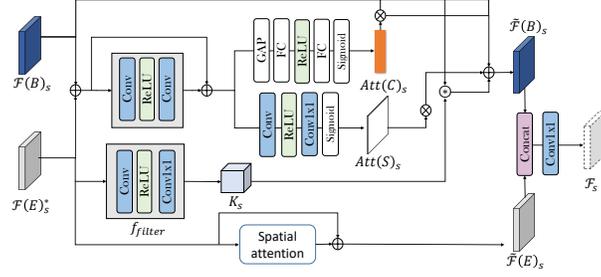
**Fig. 3.** The network structure of the proposed Exposure-Time based Event Selection Module. “GN” denotes the group normalization [46].

the feature pyramid obtained from two different modalities. To this end, we propose a novel ETES module *without any supervision*, as depicted in Fig. 3. The main idea is to temporally mine the essential channels of event features based on the multi-scale cross-modal correlation. That is, we aim to suppress the event feature corresponding to the duration of the readout phase by calculating the similarity between the blur frame feature and the event feature along with the temporal flow. As depicted in Fig. 3(a), we first pre-process the event and frame features to calculate cross-modal correlation at multiple visual scales. For the frame features at scale  $s$ , we first compress them by applying the point-wise convolution to reduce spatial information loss. Then, we replicate the compressed frame features to have the same temporal dimension as the event features. The operations are formulated as:

$$\hat{\mathcal{F}}(B)_s = \xi(\text{Conv}_{1 \times 1, s}(\mathcal{F}(B)_s)), \quad (6)$$

where  $\text{Conv}_{1 \times 1, s}$  is the point-wise convolution at scale  $s$  such that  $\mathbb{R}^{C_s^U \times H_s \times W_s} \rightarrow \mathbb{R}^{C_{s=1}^U \times H_s \times W_s}$ . Here,  $\xi$  denotes the replication operation along the temporal dimension to form frame template features  $\hat{\mathcal{F}}(B)_s$  such that  $\mathbb{R}^{C_{s=1}^U \times H_s \times W_s} \rightarrow \mathbb{R}^{T \times C_{s=1}^U \times H_s \times W_s}$ . This allows calculating the cross-modal correlation in unit time interval  $\Delta t_{unit}$ . For the event features at scale  $s$ , we leverage a siamese network [13] to modulate them to effectively maintain the temporal information within  $\Delta t_{unit}$ , as shown in Fig. 3(a). We apply this w.r.t. each temporal unit of the event features,  $f_s^{ref} : \mathbb{R}^{C_s^U \times H_s \times W_s} \rightarrow \mathbb{R}^{C_{s=1}^U \times H_s \times W_s}$ . The features are concatenated along the temporal dimension to form event template features  $\hat{\mathcal{F}}(E)_s$ . We finally apply the group normalization [46] to these two template features to mitigate the extreme modality differences. Through pre-processing both the event and frame features, we can get two feature templates at each scale  $s$ . As shown in Fig. 3(b), it is imperative to explore the correlations from the feature pyramids to select the most beneficial events. For this reason, we aggregate these two feature templates via the Hadamard product  $\odot$  for all scales as

$$\hat{C}_s = \text{ReLU}(\hat{\mathcal{F}}(E)_s \odot \hat{\mathcal{F}}(B)_s) \quad (7)$$



**Fig. 4.** Proposed feature fusion module. In the figure,  $\otimes$  denotes dynamic convolution operation with a generated convolution filter.

where the feature correlation  $\hat{C}_s \in \mathbb{R}^{T \times C_{s=1}^U \times H_s \times W_s}$  with ReLU for removing noisy correlation. Finally, we get a feature correlation pyramid  $\{C_s\}$  seen from multiple visual scales. As illustrated in Fig. 3(c), we merge the collection of correlation features by designing a scale-fusion convolutional (SFC) block. In particular, SFC aims to form a multi-scale temporal activation map  $\{Z_s\} \in \mathbb{R}^{T \times C_s^U \times 1 \times 1}$  in three steps. SFC upsamples the correlation features at scale  $s$ , followed by an element-wise addition with the features at scale  $s + 1$ . It then propagates the most beneficial correlation information in a top-down manner. In such a way, it effectively enables the merge of lower-level to a higher-level cross-modal correlation between the frame and event. Lastly, we squeeze the output tensor on the spatial dimension by global average-pooling followed by a sigmoid activation function. The output tensor is the condensed temporal activation map  $Z_{s=1} \in \mathbb{R}^{T \times C_{s=1}^U \times 1 \times 1}$ . We then interpolate  $Z_{s=1}$  to get the temporal activation map at each scale  $s$   $Z_s \in \mathbb{R}^{T \times C_s^U \times 1 \times 1}$  as

$$\mathcal{F}(E)_s^* = Z_s \otimes \mathcal{F}(E)_s \quad (8)$$

where  $\otimes$  denotes channel-wise multiplication. As such, the ETES module filters and selects the event features w.r.t. unknown exposure time of the frame, as shown in Fig. 7. Each of the selected event feature pyramids  $\{\mathcal{F}(E)_s^*\}$  is fed into the feature fusion module of each scale, as depicted in Fig. 2.

### 3.4 Feature Fusion Module

The event features  $\mathcal{F}(E)_s^*$  are selected via the event selection module. We notice that the blur frame features  $\mathcal{F}(B)_s$  contain rich semantic and texture information, while the selected event features contain clear edge and motion cues. Therefore, it is meaningful to leverage the information from event features to complement the frame features. It is possible to use existing feature fusion methods directly, *e.g.*, RGB-D [35,54]. However, as events are remarkably different from the frame, naively using these methods rather degrades the deblurring performance. Consequently, we propose a novel event-frame feature fusion module

for deblurring, as depicted in Fig. 4. Our idea is to leverage the event features to mine the discriminative channels and spatial information from the blur frame features. That is, we first calibrate the frame and event features via element-wise summation. As a result, the frame features of blurry regions are highlighted through calibration due to the motion cues of the event features. We then employ the global average pooling(GAP) to obtain global statistics, which are fed into a fully connected(FC) layer to obtain a channel attention vector for the frame features  $Att(C)_s \in \mathbb{R}^{T \times C_s \times 1 \times 1}$ . Meanwhile, we extract spatial attention maps  $Att(S)_s$  to attain the spatial statistics from calibrated features. As such, only the important blur frame features related to deblurring are highlighted, and the unnecessary information is suppressed. Moreover, we design a filter generation block  $f_{\text{filter}}$  (see Fig. 4) to generate a position-specific convolution filter  $K_s = f_{\text{filter}}(\mathcal{F}(E)_s^*)$  from the selected event features  $\mathcal{F}(E)_s^*$ , inspired by kernel prediction networks [8,22,18,17]. The filtered features are formulated as:

$$\tilde{\mathcal{F}}(B)_s = \mathcal{F}(B)_s + Att(C)_s \otimes \mathcal{F}(B)_s + Att(S)_s \cdot \mathcal{F}(B)_s + K_s \otimes \mathcal{F}(B)_s$$

where  $\otimes$  denotes convolution operation and  $\otimes$  denotes channel-wise multiplication. For event features, we only apply spatial attention [45] from the selected event feature  $\mathcal{F}(E)_s^*$ . The fused features  $\mathcal{F}_s$  at each scale are obtained by concatenating the enhanced event features  $\tilde{\mathcal{F}}(E)_s$  and the filtered frame features  $\tilde{\mathcal{F}}(B)_s$ , followed by a  $1 \times 1$  convolution. Lastly, the fused feature  $\{\mathcal{F}_s\}$  are separately fed into the decoder to reconstruct a sharp video, as shown in Fig. 2. The outputs of the decoder consist of sharp video frames at each scale, represented as  $\{O_s\}$ . We use the charbonnier loss [10] for optimization, and the total loss is:

$$\mathcal{L}_{total} = \sum_{s=0}^2 \lambda_s \sqrt{\|O_{gt,s} - O_s\|^2 + \varepsilon^2} \quad (9)$$

We empirically set to  $\varepsilon = 10^{-3}$  for all experiments.

## 4 Experiments

### 4.1 Datasets and Implementation Details

**Synthetic event datasets** We train and test our framework on the GoPro dataset [20], widely used for deblurring. We then test with the test split of the Adobe-240fps datasets [34] using the trained model on the GoPro dataset. For both datasets, we follow an official data split, and events are generated from the high frame rate video using the event simulator (ESIM) [30]. We synthesize blurry video by averaging the video frames. To mimic the real video frame acquisition, we follow the method in [55]. We discard several video frames to simulate the readout time. We denote the number of video frames of the exposure phase as  $m$  and that of the readout phase as  $n$ . We downsample the original video from 240fps to 15fps with  $m + n = 16$ . We set the frame number  $m$  of the exposure phase as an odd number  $m = \{9, 11, 13, 15\}$ . During training, we add

**Table 1.** Quantitative evaluation on a synthetic event dataset. Asterisk(\*) means retraining our training dataset. † denotes the event-guided method. The **Bold** and underline denote the best and the second-best performance, respectively. We trained our method on the GoPro dataset and directly applied it to the Adobe dataset. The same notation and typography are applied to the following tables.

Method	GoPro -15fps								Adobe-15fps							
	GoPro-9-7		GoPro-11-5		GoPro-13-3		GoPro-15-1		Adobe-9-7		Adobe-11-5		Adobe-13-3		Adobe-15-1	
	PSNR	SSIM														
Nah <i>et al.</i> [20]	28.89	0.930	27.76	0.914	26.70	0.895	25.74	0.876	28.29	0.914	27.35	0.900	26.59	0.888	25.96	0.878
DMPHN [49]	31.21	0.946	30.39	0.936	28.75	0.914	26.83	0.880	29.17	0.920	28.22	0.905	27.21	0.888	26.31	0.872
Kupyn <i>et al.</i> [15]	30.18	0.932	29.10	0.918	28.11	0.903	27.21	0.888	29.81	0.924	28.85	0.913	28.09	0.903	27.51	0.894
DBGAN [51]	32.15	0.955	31.44	0.945	29.25	0.921	26.99	0.881	29.91	0.927	28.82	0.912	27.70	0.895	26.69	0.878
BANet [37]	33.02	0.961	32.38	0.956	30.89	0.941	28.93	0.915	31.01	0.943	29.96	0.929	28.84	0.913	27.90	0.898
MPRNet [48]	33.77	0.967	32.65	0.959	31.24	0.946	29.66	0.927	31.20	0.945	30.17	0.933	29.05	0.919	28.16	0.906
MIMO-UNet+ [4]	33.34	0.964	32.39	0.956	30.74	0.940	28.52	0.908	30.83	0.939	29.76	0.925	28.57	0.908	27.51	0.892
HNNet [3]	33.60	0.965	32.86	0.960	31.26	0.945	29.09	0.917	31.06	0.944	30.08	0.932	28.90	0.915	27.91	0.901
ESTRNN* [56]	29.97	0.929	28.93	0.916	27.96	0.901	27.04	0.885	28.36	0.907	27.45	0.893	26.72	0.881	26.10	0.870
CDVD-TSP [25]	29.13	0.926	28.53	0.917	27.77	0.905	26.89	0.890	27.57	0.907	27.21	0.900	26.80	0.892	26.43	0.885
Nah <i>et al.</i> †* [20]	34.39	0.966	34.37	0.965	34.04	0.963	33.66	0.961	33.63	0.962	33.64	0.962	33.10	0.958	32.25	0.950
DMPHN†* [49]	33.68	0.961	33.63	0.961	33.36	0.959	33.03	0.956	33.16	0.959	33.02	0.959	32.55	0.955	32.01	0.949
LEDVDI†* [17]	33.39	0.958	33.57	0.959	33.68	0.959	32.81	0.953	32.92	0.958	32.92	0.958	33.18	0.953	31.91	0.949
D2Nets†* [32]	29.52	0.923	29.39	0.921	29.56	0.921	29.04	0.911	28.41	0.909	28.24	0.905	28.46	0.906	28.14	0.900
<b>Ours-light</b> †	34.91	0.969	34.61	0.968	34.43	0.966	34.00	0.964	34.19	0.965	33.72	0.963	33.87	0.964	33.20	0.959
<b>Ours</b> †	<b>36.22</b>	<b>0.976</b>	<b>35.93</b>	<b>0.974</b>	<b>35.67</b>	<b>0.973</b>	<b>35.28</b>	<b>0.971</b>	<b>35.52</b>	<b>0.973</b>	<b>35.24</b>	<b>0.971</b>	<b>35.11</b>	<b>0.970</b>	<b>34.67</b>	<b>0.968</b>

random noise,  $\epsilon \sim [-0.6n, 0.6n]$ , to the readout interval for better generalization. Accordingly, we can simulate the random video frame acquisition process in training. As such, we get a synthetic dataset that simulates various exposure times. We denoted this dataset as “dataset-m-n”.

**Real-world event datasets** To evaluate our method on real-world events, we collected 53,601 sharp images of 59 different scenes with slow-motion using the DAVIS 346 color event camera that provides aligned events and RGB data ( $346 \times 260$  resolution). We attain the blurry video by averaging the sharp frames with the setting  $m + n = 10$  for network training. We set the frame number of the exposure phase as an odd number  $m = \{3, 5, 7, 9\}$ . Similarly, we add a random noise  $\epsilon \sim [-0.6n, 0.6n]$  to the read-out interval in the training phase. Finally, we generated training sets consisting of blur images and corresponding sharp ground truths images with events for 43 scenes. For testing, *we set the test set configuration differently from the composition of the training sets* by setting  $m + n = 14$  and  $m = \{9, 11, 13\}$  to confirm the generalization ability for the unseen video frame acquisition process. Finally, we generated a test set consisting of 3,588 blur and GT images with events for 16 scenes. In this manner, we conduct a quantitative evaluation with other methods. In addition, we collected real-world blurry videos by shooting various scenes with fast motion to evaluate our method on the real-world blurry videos qualitatively. For the real-world blurry video shooting, we set the exposure time as  $\{15, 25, 35, 45, 55\}$ ms or auto-exposure and the shutter period as 60ms. Then, we conducted experiments on real-world blurry videos at various unknown exposure times.

**Implementation Details** Our frameworks are implemented using PyTorch [29]. For all datasets, we utilize the batch size of 8 and ADAM [12] optimizer to update weight using a multi-step scheduler with an initial learning rate  $1e^{-4}$  and

**Table 2.** Quantitative evaluation and complexity analysis on real event dataset. The inference time and FLOPs are measured using TITAN RTX GPU on  $346 \times 260$  resolution images of test sets. † means event-guided deblurring methods evaluated using the official pretrained model. As their official models only provide the result of a grayscale image input, we evaluate the performance of each model on the grayscale image input.

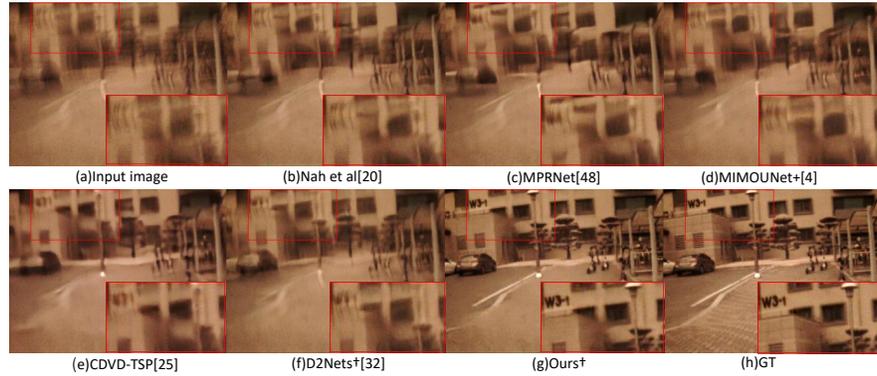
Methods	Real-world event dataset								Complexity FLOPs(G)/Runtime(ms)
	9-5		11-3		13-1		Avg.		
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	
Nah <i>et al.</i> * [20]	29.74	0.8420	28.74	0.8207	27.96	0.8037	28.81	0.8221	245.40/154.65
DMPHN* [49]	29.76	0.8392	28.80	0.8188	28.03	0.8021	28.87	0.8200	56.99/27.99
CDVD-TSP* [25]	32.95	0.9077	31.73	0.8878	30.66	0.8683	31.78	0.8880	281.15/204.46
MPRNet* [48]	30.42	0.8596	29.22	0.8345	28.25	0.8136	29.30	0.8359	1247.17/151.9
MIMO-UNet+* [4]	30.32	0.8529	29.24	0.8309	28.37	0.8129	29.31	0.8322	112.44/44.0
e-SLNet‡ [38]	20.82	0.6379	21.39	0.6603	22.09	0.6872	21.43	0.6546	114.88/55.87
REDS‡ [47]	26.12	0.7399	30.20	0.8448	31.34	0.8542	29.22	0.8130	116.89/47.15
Nah <i>et al.</i> †* [20]	35.10	0.9326	32.84	0.9057	32.41	0.9016	33.45	0.9133	247.00/156.32
DMPHN†* [49]	33.87	0.9069	33.09	0.8958	33.02	0.8957	33.33	0.8995	57.88/27.22
LEDVDI†* [17]	34.77	0.9258	33.83	0.9138	32.96	0.9047	33.86	0.9148	62.80/25.24
D2Nets†* [32]	31.36	0.8753	30.87	0.8663	29.90	0.8481	30.71	0.8632	283.86/243.9
Ours-light†	35.53	0.9342	34.58	0.9232	34.64	0.9248	34.92	0.9274	60.72/32.14
Ours†	<b>36.98</b>	<b>0.9487</b>	<b>36.10</b>	<b>0.9407</b>	<b>35.98</b>	<b>0.9404</b>	<b>36.35</b>	<b>0.9433</b>	237.77/75.07

decay rate  $\gamma = 0.5$ .  $\lambda_s$  of Eq.(9) are set to  $\{1, 0.1, 0.1\}$  for each scale. For data augmentation, we apply random cropping( $256 \times 256$ ) to the event and frame for the same position. We adopt the dynamic convolution operation from the STFAN [57] implementation. For quantitative evaluation, we use common evaluation metrics PSNR and SSIM [43].

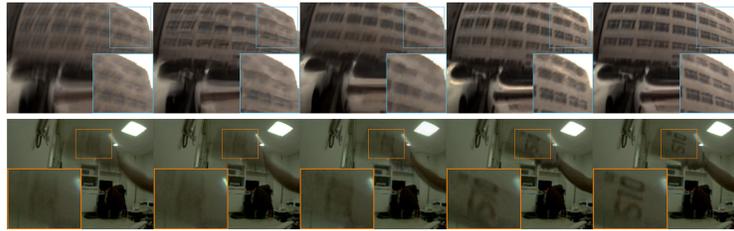
## 4.2 Experimental Results

We compare with Nah *et al.* [20] and DMPHN [49] by feeding the RGB frame with embedded events to the networks (denoted as Nah *et al.*†, DMPHN†). For comparison of the SOTA event-guided video deblurring method D2Nets† [32], we used the official training code. In addition, we reimplement the other SOTA event-guided video deblurring methods [17](denoted as LEDVDI†) based on the code provided by the authors. We keep the original network architecture and with modification of the event representation [58].

**Synthetic event datasets** For a fair comparison, we *retrain* the one frame-based video deblurring method [56] and four event-guided methods(D2Nets†, LEDVDI† and Nah *et al.*† and DMPHN†) on our training dataset. Also, we compare with the SOTA frame-based methods using official pre-trained models provided by the authors [25,20,49,51,15,37,48,4,3]. As clearly shown in Tab. 1, our method surpasses the frame-based and event-guided methods by a large margin on the two datasets. Compared to the frame-based method, the avg. PSNR score of our method improves from 3.95dB to 8.50dB in the GoPro-15fps, from 5.49dB to 8.09dB in the Adobe-15fps. As we maximize the number of video frames corresponding to the exposure phase, the performance gap between our method and frame-based competitors widens from 5.62dB to 9.54dB in the GoPro-15-1. This indicates our method achieves better results on challenging



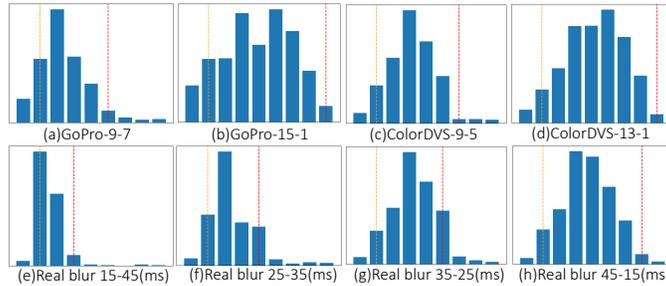
**Fig. 5.** Visual comparison on the test split of real-world event datasets.



**Fig. 6.** Deblurring results on **real-world** unknown exposure time blurry video frames. From left to right: inputs, MIMO-UNet+ [4], D2Nets<sup>†</sup> [32], LEDVDI<sup>†</sup> [17], Ours<sup>†</sup>.

motion blur frames. Compared to the event-guided method, our method still shows better results from 1.66 dB to 6.40dB in the GoPro-15fps and 1.98 to 6.82dB in the Adobe-15fps.

**Real-world event datasets** As mentioned above, the real-world datasets contain various scenes different from the synthetic event dataset, so we re-train the SOTA frame-based methods [25,49,20,48,4] and event-guided methods (LEDVDI<sup>†</sup>, Nah *et al.*<sup>†</sup>, DMPHN<sup>†</sup>, D2Nets<sup>†</sup>) on our real-world event datasets. In addition, we evaluate the SOTA event-guided deblurring methods(e-SLNet<sup>‡</sup> [38], REDS<sup>‡</sup> [47]) using official pretrained model provided by the authors. In Tab. 2, we reported the deblurring performance of unseen and unknown exposure time videos compared with other methods. Compared to the frame-based methods, there is a performance gap(**4.57** to **7.54** dB) on avg. PSNR and SSIM(**0.0553** to **0.1233**). Moreover, our network records a significant performance improvement compared to the event-guided methods(retrained in our datasets) on avg. PSNR(**2.49** to **5.64** dB) and SSIM(**0.0285** to **0.0801**) with comparable test cost and running time. The “ours-light” network, the light version of our original network with reduced channels and res-blocks, has higher performance (1.06dB to 4.21dB) with low computational cost. Ours-light network shows the slight difference of test costs and running time with two light event-guided methods



**Fig. 7.** The visualization of the average temporal activation map of the ETES module on various datasets, including real-shot blurry videos. The horizontal and vertical axes represent the temporal axis and the average amount of channel activation, respectively. The yellow and red dotted lines indicate the start and end of the exposure time, respectively. In the second row, the first and the last numbers indicate the exposure and readout time of real-world blurry videos.

**Table 3.** The ablation study of the individual components.

RE			✓	✓	✓	✓	✓	✓
MS		✓		✓			✓	✓
ETES					✓	✓	✓	✓
FF						✓		✓
PSNR	34.45	34.93	34.96	35.42	35.77	36.09	<u>36.02</u>	<b>36.35</b>

(DMPHN<sup>†</sup> [49], LEDVDI<sup>†</sup> [17]) and faster and lighter than the other two event-guided methods (D2Nets<sup>†</sup> [32], Nah *et al.*<sup>†</sup> [20]) with distinct performance differences. In Fig. 5, only our method effectively restores a detailed structure even under extreme non-linear motion.

**Evaluation results on real-world blurry video frames** Finally, we experiment generalization ability of our methods on real-world blurry videos. For testing, real-shot blurry video frames and corresponding events (in  $1/(\text{shutter speed})$  duration) are given to our model inputs. Then, we checked the results for the temporal activation map of our ETES module according to various unknown exposure times to confirm the inference ability of exposure time for deblurring. Specifically, we plot the average distribution of the activated event channels for each temporal unit of the ETES module by averaging over 200 real-world blurry frames for each video clip in Fig. 7(e), (f), (g), (h). ***We confirm all temporal activations are concentrated near the exposure phase even in the real-blurry video frames without exact exposure time information.*** This indicates that event features are well selected based on cross-modality feature correlations according to real-world blurry videos’ exposure time. Lastly, we performed qualitative comparisons with other methods (D2Nets<sup>†</sup> [32], LEDVDI<sup>†</sup> [17], MIMO-UNet+ [4]) as illustrated in Fig. 6. Only our method can restore the letters written on the box in the second row.

### 4.3 Ablation study

We analyzed the performance contribution of our network modules. All ablation experiments are performed on the real-world event dataset with models trained for  $2.5 \times 10^5$  iterations.

**Recurrent Encoding (RE)** To demonstrate the effectiveness of RE, we compare the model using 2D CNNs and the proposed shared-RNN cell for embedding event streams. From the 1st and 3rd rows of Tab. 3, we observe a performance gain (+0.51db) when using RE. On the other hand, when using the 2D CNNs for encoding events, temporal information lying in the event streams is not well preserved, adversely affecting deblurring. Furthermore, if we use a multi-scale loss function (denoted as “MS”), we observe a performance gain (+0.46db).

**ETES module** is the most crucial module in our method. We validate the deblurring performance with and without this module to verify the effectiveness. From the 3rd and 5th columns of Tab. 3, we observe a significant performance improvement (+0.81db) in terms of the average PSNR. We plot the average distribution of the ETES module’s output according to various unknown exposure times using all video frames of the test sets in the synthetic event datasets (Fig. 7(a), (b)) and real-world event datasets (Fig. 7(c), (d)). Here, we confirm that all activations are within the exposure phase and hardly activate at the readout phase, even in the different configuration in the training phase. This shows the *efficacy* of the ETES module.

**Feature Fusion Module** For cross-modality feature fusion, the simplest way is to use a concatenation of two features to fuse two different modality features. In Tab. 3, by comparing 7rd and 8th columns, we can observe performance improvement (+0.33db). The proposed feature fusion module can better utilize the complementary information from frame and event features.

## 5 Conclusion

This paper studied and formulated a new research problem of event-guided motion deblurring for unknown exposure time videos. To this end, we proposed a novel end-to-end framework. Specifically, we proposed a method of selectively using event features by estimating the feature correlation of different modalities of events and frames. Moreover, extensive experiments demonstrated our method significantly surpasses existing event-guided and frame-based deblurring methods on the various datasets, including real-world blurry videos.

**Acknowledgements** This work was supported by Institute of Information and Communications Technology Planning & Evaluation(IITP) Grant funded by Korea Government (MSIT) (No. 2020-0-00440, Development of Artificial Intelligence Technology that Continuously Improves Itself as the Situation Changes in the Real World and No.2014-3-00123, Development of High Performance Visual BigData Discovery Platform for Large-Scale Realtime Data Analysis) and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (NRF2022R1A2B5B03002636)

## References

1. Argaw, D.M., Kim, J., Rameau, F., Kweon, I.S.: Motion-blurred video interpolation and extrapolation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 901–910 (2021)
2. Brehm, S., Scherer, S., Lienhart, R.: High-resolution dual-stage multi-level feature aggregation for single image and video deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2020)
3. Chen, L., Lu, X., Zhang, J., Chu, X., Chen, C.: Hinet: Half instance normalization network for image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 182–192 (2021)
4. Cho, S.J., Ji, S.W., Hong, J.P., Jung, S.W., Ko, S.J.: Rethinking coarse-to-fine approach in single image deblurring. arXiv preprint arXiv:2108.05054 (2021)
5. Fellers, T., M.W, D.: Digital camera readout and frame rates. <https://hamamatsu.magnet.fsu.edu/articles/readoutandframerates.html> (2020-06-04)
6. Gao, H., Tao, X., Shen, X., Jia, J.: Dynamic scene deblurring with parameter selective sharing and nested skip connections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
7. Hyun Kim, T., Mu Lee, K., Scholkopf, B., Hirsch, M.: Online video deblurring via dynamic temporal blending network. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4038–4047 (2017)
8. Jia, X., De Brabandere, B., Tuytelaars, T., Gool, L.V.: Dynamic filter networks. *Advances in neural information processing systems* **29**, 667–675 (2016)
9. Jiang, Z., Zhang, Y., Zou, D., Ren, J., Lv, J., Liu, Y.: Learning event-based motion deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3320–3329 (2020)
10. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)
11. Kim, T.H., Sajjadi, M.S., Hirsch, M., Scholkopf, B.: Spatio-temporal transformer network for video restoration. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 106–122 (2018)
12. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
13. Koch, G., Zemel, R., Salakhutdinov, R., et al.: Siamese neural networks for one-shot image recognition. In: ICML deep learning workshop. vol. 2. Lille (2015)
14. Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J.: Deblurgan: Blind motion deblurring using conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
15. Kupyn, O., Martyniuk, T., Wu, J., Wang, Z.: Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2019)
16. Li, D., Xu, C., Zhang, K., Yu, X., Zhong, Y., Ren, W., Suominen, H., Li, H.: Arvo: Learning all-range volumetric correspondence for video deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7721–7731 (2021)
17. Lin, S., Zhang, J., Pan, J., Jiang, Z., Zou, D., Wang, Y., Chen, J., Ren, J.S.: Learning event-driven video deblurring and interpolation. In: ECCV (8). pp. 695–710 (2020)

18. Mildenhall, B., Barron, J.T., Chen, J., Sharlet, D., Ng, R., Carroll, R.: Burst denoising with kernel prediction networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2502–2510 (2018)
19. Mostafaviisfahani, S.M., Nam, Y., Choi, J., Yoon, K.J.: E2sri: Learning to super-resolve intensity images from events. *IEEE Transactions on Pattern Analysis & Machine Intelligence* (01), 1–1 (2021)
20. Nah, S., Hyun Kim, T., Mu Lee, K.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3883–3891 (2017)
21. Nah, S., Son, S., Lee, K.M.: Recurrent neural networks with intra-frame iterations for video deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8102–8111 (2019)
22. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive convolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 670–679 (2017)
23. Noroozi, M., Chandramouli, P., Favaro, P.: Motion deblurring in the wild. In: GCPR. pp. 65–77. Springer (2017)
24. Pan, J., Bai, H., Tang, J.: Cascaded deep video deblurring using temporal sharpness prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
25. Pan, J., Bai, H., Tang, J.: Cascaded deep video deblurring using temporal sharpness prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3043–3051 (2020)
26. Pan, L., Hartley, R., Scheerlinck, C., Liu, M., Yu, X., Dai, Y.: High frame rate video reconstruction based on an event camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020)
27. Pan, L., Scheerlinck, C., Yu, X., Hartley, R., Liu, M., Dai, Y.: Bringing a blurry frame alive at high frame-rate with an event camera. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6820–6829 (2019)
28. Park, D., Kang, D.U., Kim, J., Chun, S.Y.: Multi-temporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training. In: European Conference on Computer Vision. pp. 327–343. Springer (2020)
29. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
30. Rebecq, H., Gehrig, D., Scaramuzza, D.: Esim: an open event camera simulator. In: Conference on Robot Learning. pp. 969–982. PMLR (2018)
31. Rebecq, H., Ranftl, R., Koltun, V., Scaramuzza, D.: High speed and high dynamic range video with an event camera. *IEEE transactions on pattern analysis and machine intelligence* (2019)
32. Shang, W., Ren, D., Zou, D., Ren, J.S., Luo, P., Zuo, W.: Bringing events into video deblurring with non-consecutively blurry frames. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4531–4540 (October 2021)
33. Shen, W., Bao, W., Zhai, G., Chen, L., Min, X., Gao, Z.: Blurry video frame interpolation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5114–5123 (2020)
34. Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., Wang, O.: Deep video deblurring for hand-held cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1279–1288 (2017)

35. Sun, P., Zhang, W., Wang, H., Li, S., Li, X.: Deep rgb-d saliency detection with depth-sensitive attention and automatic multi-modal fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1407–1417 (2021)
36. Tao, X., Gao, H., Shen, X., Wang, J., Jia, J.: Scale-recurrent network for deep image deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8174–8182 (2018)
37. Tsai, F.J., Peng, Y.T., Lin, Y.Y., Tsai, C.C., Lin, C.W.: Banet: Blur-aware attention networks for dynamic scene deblurring. arXiv preprint arXiv:2101.07518 (2021)
38. Wang, B., He, J., Yu, L., Xia, G.S., Yang, W.: Event enhanced high-quality image recovery. In: European Conference on Computer Vision. Springer (2020)
39. Wang, L., Chae, Y., Yoon, K.J.: Dual transfer learning for event-based end-task prediction via pluggable event to image translation. In: ICCV (2021)
40. Wang, L., Chae, Y., Yoon, S.H., Kim, T.K., Yoon, K.J.: Evidistill: Asynchronous events to end-task learning via bidirectional reconstruction-guided cross-modal knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 608–619 (2021)
41. Wang, L., Kim, T.K., Yoon, K.J.: Eventsr: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning. In: CVPR. pp. 8315–8325 (2020)
42. Wang, X., Chan, K.C., Yu, K., Dong, C., Change Loy, C.: Edvr: Video restoration with enhanced deformable convolutional networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2019)
43. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13**(4), 600–612 (2004)
44. Wieschollek, P., Hirsch, M., Scholkopf, B., Lensch, H.: Learning blind motion deblurring. In: ICCV. pp. 231–240 (2017)
45. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
46. Wu, Y., He, K.: Group normalization. In: Proceedings of the European conference on computer vision (ECCV). pp. 3–19 (2018)
47. Xu, F., Yu, L., Wang, B., Yang, W., Xia, G.S., Jia, X., Qiao, Z., Liu, J.: Motion deblurring with real events. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2583–2592 (October 2021)
48. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L.: Multi-stage progressive image restoration. arXiv preprint arXiv:2102.02808 (2021)
49. Zhang, H., Dai, Y., Li, H., Koniusz, P.: Deep stacked hierarchical multi-patch network for image deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5978–5986 (2019)
50. Zhang, K., Luo, W., Zhong, Y., Ma, L., Liu, W., Li, H.: Adversarial spatio-temporal learning for video deblurring. *IEEE Transactions on Image Processing* (2018)
51. Zhang, K., Luo, W., Zhong, Y., Ma, L., Stenger, B., Liu, W., Li, H.: Deblurring by realistic blurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2737–2746 (2020)
52. Zhang, K., Luo, W., Zhong, Y., Ma, L., Stenger, B., Liu, W., Li, H.: Deblurring by realistic blurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)

53. Zhang, L., Zhang, H., Zhu, C., Guo, S., Chen, J., Wang, L.: Fine-grained video deblurring with event camera. In: ICMM. pp. 352–364. Springer (2021)
54. Zhang, M., Fei, S.X., Liu, J., Xu, S., Piao, Y., Lu, H.: Asymmetric two-stream architecture for accurate rgb-d saliency detection. In: European Conference on Computer Vision. pp. 374–390. Springer (2020)
55. Zhang, Y., Wang, C., Tao, D.: Video frame interpolation without temporal priors. *Advances in Neural Information Processing Systems* **33** (2020)
56. Zhong, Z., Gao, Y., Zheng, Y., Zheng, B.: Efficient spatio-temporal recurrent neural network for video deblurring. In: European Conference on Computer Vision. pp. 191–207. Springer (2020)
57. Zhou, S., Zhang, J., Pan, J., Xie, H., Zuo, W., Ren, J.: Spatio-temporal filter adaptive network for video deblurring. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2019)
58. Zhu, A.Z., Yuan, L., Chaney, K., Daniilidis, K.: Unsupervised event-based learning of optical flow, depth, and egomotion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 989–997 (2019)
59. Zou, Y., Zheng, Y., Takatani, T., Fu, Y.: Learning to reconstruct high speed and high dynamic range videos from events. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2024–2033 (2021)