All-in-focus Imaging from Event Focal Stack

Hanyue Lou[†1,2] Minggui Teng[†1,2] Yixin Yang[1,2] Boxin Shi[*1,2]

[1] National Key Laboratory for Multimedia Information Processing, School of Computer Science, Peking University

[2] National Engineering Research Center of Visual Technology, School of Computer Science, Peking University

{hylz, minggui_teng, yangyixin93, shiboxin}@pku.edu.cn

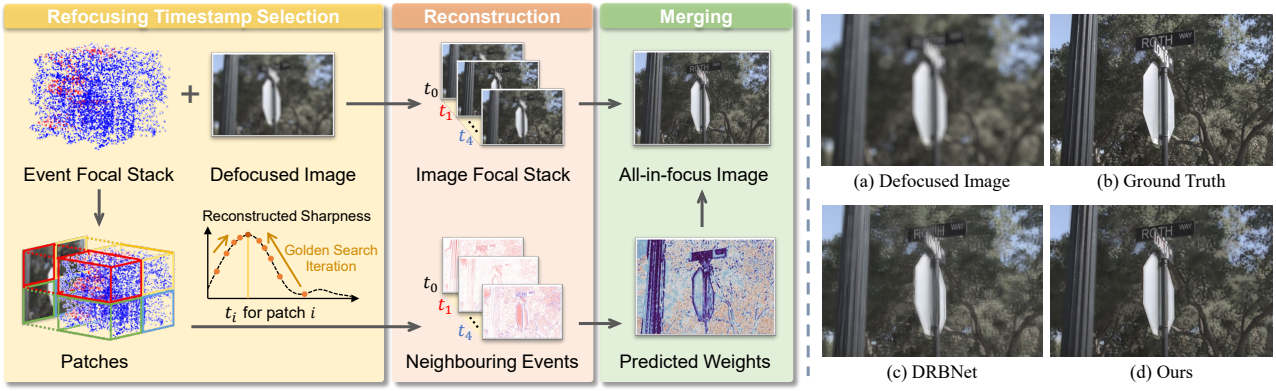


Figure 1. Left: We propose the event focal stack composed of event streams, which can be used to reconstruct an image focal stack and predict the merging weights for all-in-focus image recovery. Our pipeline consists of three steps: selecting the refocusing timestamps, reconstructing the corresponding image focal stack, and merging the stack into an all-in-focus image with weights predicted from the images and neighbouring events. Right: Given a defocused image (a) and the corresponding event focal stack, our method recovers an all-in-focus image (d) with closer clarity to the ground truth (b) than DRBNet [31] (c).

Abstract

Traditional focal stack methods require multiple shots to capture images focused at different distances of the same scene, which cannot be applied to dynamic scenes well. Generating a high-quality all-in-focus image from a single shot is challenging, due to the highly ill-posed nature of the single-image defocus and deblurring problem. In this paper, to restore an all-in-focus image, we propose the event focal stack which is defined as event streams captured during a continuous focal sweep. Given an RGB image focused at an arbitrary distance, we explore the high temporal resolution of event streams, from which we automatically select refocusing timestamps and reconstruct corresponding refocused images with events to form a focal stack. Guided by the neighbouring events around the selected timestamps, we can merge the focal stack with proper weights and restore a sharp all-in-focus image. Experimental results on both synthetic and real datasets show superior performance over state-of-the-art methods.

[†] Contributed equally to this work as first authors

[*] Corresponding author

Project page: https://hylz-2019.github.io/EFS

1. Introduction

The lens aperture of a camera controls the amount of incoming luminous flux. A larger aperture maintains the signal-to-noise ratio with shorter exposure time, which is useful for shooting high-speed scenes or capturing images in low-light conditions with less noise. However, large aperture settings also make the depth of field (DoF) shallow, which results in defocus blur. This is preferable in certain scenarios, such as in portrait photography a shallow DoF can be used to emphasize the subject. Yet, all-in-focus images preserve information from all distances and are desired in more situations, *e.g.*, microscopy imaging [25]. Besides, all-in-focus imaging also benefits various high-level vision tasks, *e.g.*, object detection [29] and semantic segmentation [10].

An all-in-focus image could be obtained by deblurring a defocused image, but the defocus kernel, determined by the aperture shape and depth of the scene, is usually spatially-varying and difficult to be estimated accurately [48]. Conventional two-stage methods [9, 13, 36] first estimate the pixel-wise or patch-wise defocus kernels with image priors and then apply non-blind image deconvolution to each

pixel or patch. Recently, benefiting from the data-driven strategy, end-to-end deep learning methods [18, 31, 32, 38] outperform conventional two-stage restoration methods, by observing defocused and all-in-focus image pairs during training. Although they have demonstrated high potential in removing defocus blur, the deblurred results still cannot avoid ringing artifacts or remain blurry in high-frequency regions due to inaccurate defocus kernel estimation especially for weakly textured and defocused regions (an example is shown in Figure 1 right (c)).

To overcome the ill-posedness of estimating the defocus kernel from a single image, merging a focal stack, *i.e.*, a sequence of images taken at different focus distances, can generate an all-in-focus image reliably [11, 40, 47]. However, capturing a focal stack requires a static scene and multiple exposures. Moreover, the selection of focus distances is a key factor in capturing the focal stack, which requires elaborate design.

Neuromorphic event cameras [5, 35] are novel sensors that can detect brightness changes and trigger an event whenever its log variation exceeds a preset threshold. Thanks to their high temporal resolution featured with microsecond-level sensitivity, they can capture approximately continuous signals for intensity variations of a scene, and support applications like generating high-speed videos from event streams [28, 41–43]. These characteristics motivate us to think about: *Can we use "focal stacks" composed of event streams for all-in-focus imaging?*

In this paper, we propose *event focal stack* (EFS) for the first time. It is composed of event streams obtained from a continuous focal sweep with an event camera, which can be used to reconstruct an image focal stack (given an RGB image focused at an arbitrary distance) and predict the merging weights for all-in-focus image recovery, as shown in Figure 1 left. EFS encodes scene texture information from continuous different depths in temporal log-gradient domain, so we first select a refocusing timestamp for each patch of the scene, which corresponds to sharper edges and richer texture information at that time. By fusing a defocused image and the EFS recorded between the defocused timestamp and refocusing timestamp, we generate a refocused image for each refocusing timestamp, forming an image focal stack. Guided by neighbouring events around refocusing timestamps, we can predict the merging weight for each image needed for composing a focal stack, and finally restore an all-in-focus image (an example is shown in Figure 1 right (d)). Contributions of this paper are demonstrated by exploring the following benefits of the proposed EFS:

- reliable selection of refocusing timestamps by decoding continuous scene gradient changes from events;
- consistent link between defocused (given) and refocused images (estimated) composing an image focal stack; and
- robust guidance for merging weight prediction and all-in-focus reproduction with event triggered neighbouring the selected timestamps.

We quantitatively and qualitatively evaluate our method on both synthetic and real datasets and demonstrate its superior quality in recovering all-in-focus images over state-of-the-art methods.

2. Related Work

In this section, we briefly review all-in-focus image recovery methods in two categories: image-based methods and computational photography methods. The inputs for image-based methods are obtained using conventional cameras with a single shot, while computational photography methods use a specific capture pipeline or unconventional lenses or sensors. The event-based video reconstruction methods, which are partially related to image focal stack generation from events, are also reviewed.

Image-based methods. Conventional defocus deblurring methods [9, 13, 36] usually contain two steps: estimating the defocus map and applying non-blind deconvolution for deblurring. The quality of deblurred results highly depends on the accuracy of the defocus map. To boost the performance of defocus map estimation, Park *et al.* [27] fused multi-scale image features and hand-crafted features to improve the accuracy of the defocus map. Lee *et al.* [17] proposed a domain adaptation method to transfer features of a synthetic defocused image to the real blurred one for reconstructing a more realistic defocus map. Zhao *et al.* [46] proposed an adversarial promoting learning framework to estimate defocus maps in a weakly-supervised manner.

To avoid the reliance on defocus map estimation in two-step approaches, recently, end-to-end defocus deblurring networks have demonstrated higher robustness and performance. Lee *et al.* [18] proposed an Iterative Filter Adaptive Network (IFAN) to handle spatially-varying and large defocus blur via predicting filters for defocused features. Son *et al.* [38] proposed a Kernel-sharing Parallel Atrous Convolutional (KPAC) block to deal with defocus blur with slightly varying shapes, which simulates the varying scales of inverse kernels. Ruan *et al.* [31] proposed a neural network trained on both light field generated and real defocused images to enhance the defocus deblurring performance. However, it is hard to recover the high-frequency regions from the defocused image, and the artifacts become obvious when applying deconvolution on a single defocus image. Thus, it is desirable to obtain all-in-focus images using a more robust method, which can record the continuous scene and depth information.

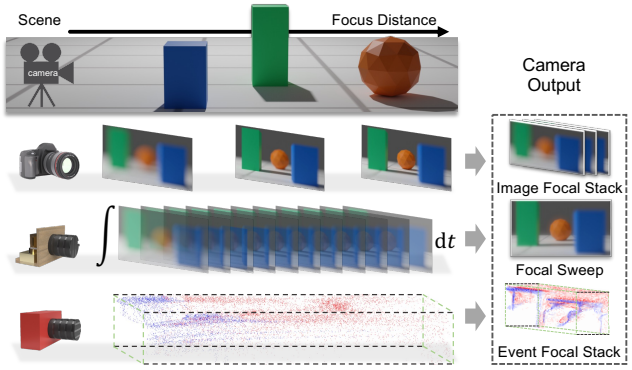


Figure 2. An illustration of image focal stack [47], focal sweep [16], and event focal stack. When focus distance sweeps from near to far, each method captures information at different depths, shown below the scene. Note that EFS continuously records the intensity changes and encodes texture information (as image gradients) from all depths to which the camera focuses.

Computational photography methods. Computational photography based defocus deblurring methods utilize specific capture pipelines (*e.g.*, focal stack [11, 47], focal sweep [16]) or unconventional lens (*e.g.*, coded aperture [19], wavefront coding [7], lattice lens [20]) to relieve the ill-posedness of defocus deblurring. Recently, Abuolaim *et al.* [1, 2] have illustrated that the difference between the two views of a dual-pixel image is related to the defocus amount and can be utilized to further improve the defocus deblurring performance. Although additionally useful cues for all-in-focus image recovery (than single image-based methods) have been encoded and decoded via various computational photography systems, existing methods still do not use continuous scene depth information, due to limitations from frame-based cameras.

Event-based video reconstruction. Reconstructing intensity frames from events can be achieved using hand-crafted features and regularization [26, 34]. More recent approaches adopt end-to-end generation methods. Rebecq *et al.* [28] synthesized video frames with a U-Net-like E2VID model. Weng *et al.* [45] presented a hybrid CNN-transformer network for intensity frame reconstruction. Zhu *et al.* [49] proposed a bio-inspired SNN to improve the image reconstruction quality. Inspired by the ability of event streams to capture continuous intensity changes, this paper explores how to perform focal sweeps with event cameras to conquer the bottlenecks of existing all-in-focus image recovery methods.

3. Proposed Method

In this section, we first introduce the event camera formation preliminaries in Section 3.1. We then formulate the event focal stack and our model for reconstructing refocused images in Section 3.2, and propose our all-in-focus imaging framework in Section 3.3. Our implementation details are illustrated in Section 3.4.

3.1. Event camera formulation preliminaries

An event signal (x, y, t, p) with polarity p is triggered whenever the log irradiance changes at pixel (x, y) and time t exceeds a preset threshold c:

$$|\log(\mathbf{I}^{t}_{x,y}) - \log(\mathbf{I}^{t-\Delta t}_{x,y})| \geq c, \tag{1}$$

in which $\mathbf{I}^{t}_{x,y}$ and $\mathbf{I}^{t-\Delta t}_{x,y}$ represent the pixel irradiance of (x, y) at time t and $t - \Delta t$, and the previous event of pixel (x, y) is triggered at $t - \Delta t$. Polarity $p \in \{1, -1\}$ indicates whether the intensity changes increase or decrease. Equation (1) applies to each pixel (x, y) independently, so pixel indices are omitted henceforth.

As events record continuous intensity changes, given two instantaneous latent images $\mathbf{I}^{t_1}$ and $\mathbf{I}^{t_2}$, let's assume N_e events occurring between t_1 and t_2, denoted as $\{e_k\}_{k=1}^{N_e}$. According to the physical model of the event camera shown in Equation (1), we can bridge $\mathbf{I}^{t_1}$ and $\mathbf{I}^{t_2}$ with corresponding events in log domain as:

$$\log \mathbf{I}^{t_2} = \log \mathbf{I}^{t_1} + \sum_{k=1}^{N_e} c_k \cdot e_k, \tag{2}$$

where c_i denotes the spatial-temporal variant threshold, related to the scene condition [12].

3.2. Event focal stack

As the Thin Lens Law $1/f = 1/u + 1/v$ shows (f is the focal length of the lens, u is the sensor-lens distance, and v is the object distance), we can change u or v to move the focal plane. Conventional image focal stack methods [11, 47] capture multiple images with different focus distances (second row of Figure 2) and then merge them to obtain an all-in-focus image. To recover an all-in-focus image, their methods must capture a focal stack such that all objects in the scene are in focus in at least one of the images in the focal stack. As illustrated in the second row of Figure 2, the blue cuboid is not focused in any image of the captured focal stack, and further leads to defocus blur in the restored image. To avoid losing scene focus information in the desired depth range, the focal sweep technique [16] changes the sensor-lens distance in the exposure time, and captures an integrated defocused image, which can be seen as an all-in-focus image convolved with an integrated *Point Spread Function* (PSF), denoted as IPSF:

$$\text{IPSF}(r, u) = \int_0^T \text{PSF}(r, u, v(t))\mathrm{d}t, \tag{3}$$

in which r represents the distance of an image point from the center of the PSF, $v(t)$ denotes the sensor-lens distance

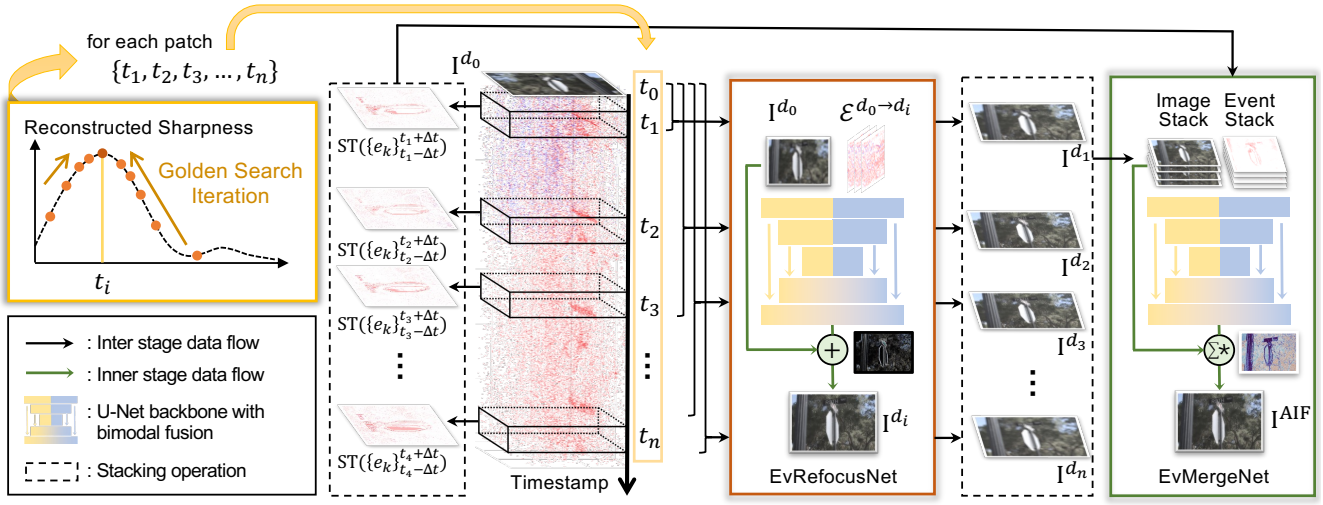


Figure 3. The pipeline of our method. We first iteratively find a refocusing timestamp according to the reconstructed sharpness for each image patch. EvRefocusNet fuses two modalities of data (RGB image $\mathbf{I}^{d_j}$ and EFS $\mathcal{E}^{d_j \to d_i}$) to reconstruct a refocused image $\mathbf{I}^{d_i}$. By applying EvRefocusNet on each timestamp, N refocused images are generated, forming an image focal stack. Then, EvMergeNet predicts merging weights guided by the events in the neighbouring time interval of each refocusing timestamp, and finally merges the reconstructed image focal stack with corresponding predicted weights to obtain an all-in-focus result $\mathbf{I}^{\mathrm{AIF}}$.

as a function of time, and T is exposure time. Kuthirummal *et al.* [16] proved that IPSF(r, u) is invariant to scene depth and image location to simplify the problem analysis. As the final output is a single defocused image (third row of Figure 2), they need to estimate the blur kernel and deconvolve images, since frame-based cameras only record scene radiance but not the radiance changes.

Applying the focal sweep technique to an event camera is quite simple. We just need to rotate the focusing ring of its lens.[1] Since the event camera owns high temporal resolution, it outputs continuous event streams that capture the pixel radiance changes as the focal plane sweeps through the scene. We call the event streams an *event focal stack* (EFS), denoted as $\mathcal{E}$:

$$\mathcal{E} = \bigcup_{u(t)} \{e_k\}_{k=1}^{N_e},\ u(t) \in (0, \infty), \tag{4}$$

where $u(t)$ denotes the focused object distance as a function of time that transforms from nearly 0 to infinity. EFS has two important advantages over image-based focal stack (fourth row of Figure 2): 1) It continuously records the intensity changes with respect to focus distance compared with a discrete set of images [47]; 2) Texture information at different depths is distinguished by the event timestamps, compared with focal sweep method, which integrates the depth information [16] and outputs a single image.

Given an image $\mathbf{I}^{d_j}$ focused at an arbitrary distance d_j, as Equation (2) has shown the relationship of two latent frames by corresponding events, we can rewrite it to connect the refocused images as:

$$\begin{aligned} \log \mathbf{I}^{d_i} &= \log \mathbf{I}^{d_j} + \sum_k c_k \cdot \bigcup_{u(t) \in (d_j, d_i)} \{e_k\} \\ &= \log \mathbf{I}^{d_j} + \mathbf{R}^{d_j \to d_i}. \end{aligned} \tag{5}$$

$\mathbf{I}^{d_j}$ and $\mathbf{I}^{d_i}$ denote the refocused images, whose focused object distances are d_j and d_i, and $\mathbf{R}^{d_j \to d_i}$ is the intensity residual computed from event summation. By iteratively applying Equation (5), we can obtain the image focal stack $\{\mathbf{I}^{d_i}\}^{N_d}$, consisting of N_d refocused images. Combining the image focal stack with proper weights, an all-in-focus image can be recovered from the EFS and an arbitrarily focused image as inputs.

3.3. All-in-focus imaging from EFS

The pipeline of our method is shown in Figure 3. We first divide the input image into patches and select a refocusing timestamp for each patch guided by reconstructed sharpness. And then an EvRefocusNet is used to reconstruct refocused images. After that, we use an EvMergeNet to predict merging weights with the EFS and finally obtain an all-in-focus result.

Refocusing time selection. As shown in the first row of Figure 2, traditional focal stack methods [11, 47] capture a set of images with uniform time intervals. To ensure that all objects are focused in at least one of the images in a focal stack, it is important to make sure each object is focused in at least one of the images that requires a specific-designed device [47] or careful selection of refocusing distances.

[1] Focal sweep setup can be found in the supplementary material.

Algorithm 1 Refocusing time selection with EFS

Data: threshold μ, golden ratio $\varphi = 1.618$
Input: EFS $\mathcal{E}$ and an RGB image $\mathbf{I}^d$
Result: Refocusing timestamp t_r
$L \leftarrow 0, R \leftarrow N_e$
while $R - L > \mu$ **do**
  $t_1, t_2 \leftarrow R - (R-L)/\varphi, L + (R-L)/\varphi$
  Reconstruct $\mathbf{I}^{d_1}, \mathbf{I}^{d_2}$ with Equation (5)
  if $\mathbb{D}(\mathbf{I}^{d_1}) > \mathbb{D}(\mathbf{I}^{d_2})$ **then** $R \leftarrow t_2$
  else $L \leftarrow t_1$
  end if
  $t_r \leftarrow (L+R)/2$
end while

According to Section 3.2, event signals represent the intensity changes, that naturally encode the temporal gradient changes [8]. Assuming local events are triggered by the same edge with uniform motion, the event triggering rate is proportional to the spatial gradient. Based on such an observation, Lin *et al.* [22] designed an auto-focus algorithm for event cameras to find the maximum event triggering rate timestamp as the refocusing timestamp. However, the majority of events in the EFS are triggered by the focal sweep, instead of object motion. Thus, the event triggering rate is not suitable as a metric for refocusing timestamp selection.

To obtain an accurate refocusing timestamp, we do not search it in the event domain. Inspired by the image-based auto-focus method [40], we use reconstructed image sharpness as a focus metric. We fuse the EFS with a given RGB image to reconstruct refocused images by Equation (5), and then utilize the variance of reconstructed image intensity value $\mathbb{D}(\mathbf{I})$ to evaluate the image sharpness. We assume that the time t_r with the maximum variance value is the refocusing timestamp we want to find. We adopt the golden-section search method in [14] to EFS for searching the time t_r with maximal image sharpness, as summarized in Algorithm 1.

The depths of objects in a scene are different, leading to different refocusing time. Therefore, we split the image into $N \times N$ spatially non-overlapping patches $\{\mathbf{I}_p^{d_j}\}^{N\times N}$, with corresponding EFS patches $\{\mathcal{E}_p\}^{N\times N}$. We apply the aforementioned Algorithm 1 to each of patches to find their refocusing times, resulting in a set of $N \times N$ refocusing timestamp, *i.e.*,

$$\{t_r\}^{N\times N} = \bigcup_p \texttt{TS}(\mathcal{E}_p, \mathbf{I}_p^{d_j}), \tag{6}$$

where TS denotes refocusing time selection with EFS using the golden-section search method [14].

EvRefocusNet. As described in Equation (2), the threshold of an event camera is not a constant [12, 21]. Besides, event cameras suffer from current leakage [12, 21], which leads to noisy events varying with illumination conditions. Given a refocusing timestamp, directly reconstructing a refocused image by Equation (5) with a constant threshold causes severe artifacts. To handle the spatial-temporal variant thresholds, we design a U-Net [30] architecture network named EvRefocusNet, to model the intensity residual $\mathbf{R}$ in Equation (5) and to generate refocused images in a data-driven manner. Given a set of refocusing timestamps computed from Algorithm 1, we obtain an image focal stack from an RGB image and the EFS denoted as:

$$\{\mathbf{I}^{d_i}\}^{N\times N} = \texttt{f}_r(\mathbf{I}^{d_j}, \{\mathcal{E}^{d_j\to d_i}\}^{N\times N}, \{t_r\}^{N\times N}), \tag{7}$$

where $\mathbf{I}^{d_j}$ is a given RGB image focusing at an arbitrary distance d_j, $\mathcal{E}^{d_j\to d_i}$ denotes corresponding events triggering between $\mathbf{I}^{d_j}$ and $\mathbf{I}^{d_i}$, and $\texttt{f}_r$ is an implicit function modeled by EvRefocusNet. As input images represent scene conditions to some extent (the defocused regions are blurry), the network can predict intensity residual with spatial-temporal variant thresholds guided by input images.

The multi-scale architecture has been proven to be effective for multi-modal data fusion [39]. We fuse the image and EFS features in the multi-scale by a U-Net backbone. We also formulate it using residual learning with global connection. By adding such a residual to the input RGB image, the refocused images can be restored. Thanks to the continuous information encoded by event streams, our network can refocus images to arbitrary time.

EvMergeNet. Given an image focal stack, the quality of the merged all-in-focus image highly depends on the accuracy of the estimated merging weights. Merging images in a focal stack can be conducted by using image spatial gradient as guidance [11]. As shown in Equation (1), since event streams naturally encode salient information along edges, reliable clues for merging weight prediction can be found in events and used to further improve merge quality. Since edge information is clearest when events are in focus, we select the events in time intervals of Δt neighbouring each selected refocusing timestamp, and transform them to an event stack $\mathbf{E}$ to guide the merging weight prediction :

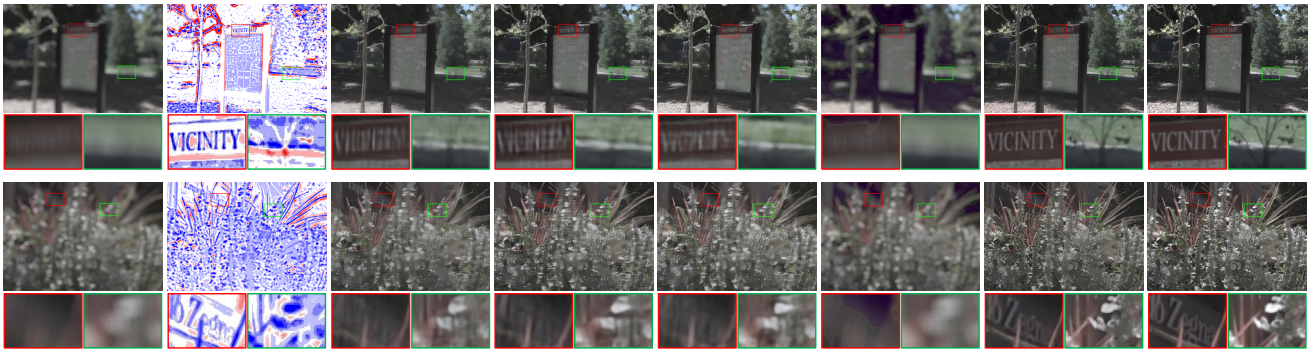
$$\mathbf{E} = \{\texttt{ST}(\{e_k\}_{t_1-\Delta t}^{t_1+\Delta t}), \ldots, \texttt{ST}(\{e_k\}_{t_{N\times N}-\Delta t}^{t_{N\times N}+\Delta t})\}, \tag{8}$$

where ST denotes event stack [44]. In our experiment, we set N as 64. We design another U-Net [30] backbone named EvMergeNet to restore the all-in-focus image $\mathbf{I}^{\text{AIF}}$ as:

$$\mathbf{W} = \texttt{f}_m(\{\mathbf{I}^{d_i}\}^{N\times N}, \mathbf{E}). \tag{9}$$

$$\mathbf{I}^{\text{AIF}} = \sum_{i=1}^{N\times N} \mathbf{W}_i \otimes \mathbf{I}^{d_i}, \tag{10}$$

where $\mathbf{W}$ is the weight matrix with dimension $N^2 \times H \times W$, $\otimes$ represents Hadamard product, and $\texttt{f}_m$ is an implicit



(a) Defocused Image (b) Events (c) DRBNet (d) IFAN (e) KPAC (f) APL (g) Ours (h) Ground Truth

Figure 4. Visual quality comparison with image-based defocus deblurring methods on synthetic data. (a) Defocused image. (b) Events. (c)∼(g) All-in-focus results of DRBNet [31], IFAN [18], KPAC [38], APL [46], and ours. (h) Ground truth. More results are in the supplementary material.

function modeled by EvMergeNet, which predicts merge weights from both the image focal stack and EFS.

To avoid over-fitting on synthetic data, we only predict the merge weights, instead of directly generating an all-in-focus image. Following the merging process [47], we apply the same weight map across all three RGB channels and merge them independently using refocused images.

3.4. Implementation details

Dataset. Since there is no large-scale image focal stack dataset with event information, we render a synthetic image focal stack with Blender [4] and simulate corresponding event streams with the latest event simulator DVS-Voltmeter [21]. Our dataset is composed of 200 random scenes. From each scene, we render an image focal stack with a shallow DoF camera setup (aperture f/1.2, focal length 100mm), which sweeps its focus distances through the scene, and an all-in-focus image as ground truth. In each scene, we further scale and scatter geometric objects to increase the diversity. To better match the data distribution to real-world images, we wrap the surfaces of the objects with images sampled from the MS-COCO dataset [23] as their textures. After rendering the image focal stack with 480 frames, we input them into DVS-Voltmeter [21] to generate event streams. To improve the generalization of the model to unknown types of event cameras, we set the 6 different camera parameters in DVS-Voltmeter randomly[2].

Training details. Both EvRefocusNet and EvMergeNet are trained with the same loss function as:

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{\text{perc}}(\mathbf{I}^{\text{o}}, \mathbf{I}^{\text{gt}}) + \beta \cdot \mathcal{L}_2(\mathbf{I}^{\text{o}}, \mathbf{I}^{\text{gt}}), \tag{11}$$

where $\alpha = 0.5$, $\beta = 100$, $\mathcal{L}_2$ denotes the MSE loss, and $\mathcal{L}_{\text{perc}}$ denotes a perceptual loss calculated from a VGG-19 network [37] pre-trained on ImageNet [33]. The output image $\mathbf{I}^{\text{o}}$ is the predicted refocused image or all-in-focus image. The corresponding ground truth is denoted as $\mathbf{I}^{\text{gt}}$.

[2]More details about the data generation pipeline can be found in the supplementary material.

Table 1. Quantitative comparisons on the LiFF dataset [6]. ↑ (↓) indicates the higher (lower), the better throughout this paper. The best performances are highlighted in **bold**.

	PSNR↑	SSIM↑	MS-SSIM↑	LPIPS↓
KPAC [38]	26.00	0.7643	0.8402	0.3778
IFAN [18]	26.97	0.7891	0.8644	0.3435
APL [46]	24.33	0.6753	0.7158	0.5471
DRBNet [31]	27.75	0.7882	0.8583	0.3243
Ours	**33.25**	**0.9323**	**0.9611**	**0.1510**

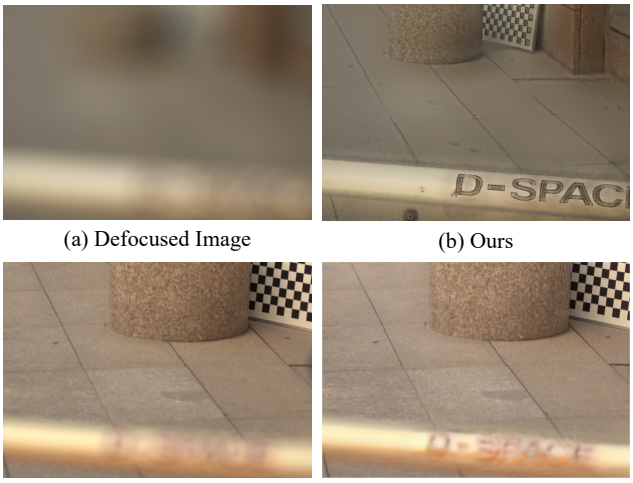
We implement our method with PyTorch on an NVIDIA GeForce GTX 1080 Ti GPU. We train both EvRefocusNet and EvMergeNet for 100 epochs, starting with the learning rate 5×10^{-4}, and after the first 50 epochs, we decrease the learning rate to $1/10$ for every 20 epochs. The ADAM optimizer [15] is used in the training phase. For EvRefocusNet training, we randomly select two frames from our synthetic dataset, one as input and the other one as ground truth, and we randomly crop the images for data augmentation. For the input of EvRefocusNet, we uniformly split the events into 64 time intervals and merge them. For the input of EvMergeNet, the i-th frame is the summation of events triggered neighboring the i-th refocus time.

4. Experimental Results

In this section, we qualitatively and quantitatively compare our method with state-of-the-art image-based defocus deblurring methods on a public synthetic dataset (Section 4.1) and our real-captured data (Section 4.2). In Section 4.3, ablation studies are conducted to validate the effectiveness of each module of the proposed method.

(a) Defocused Image (b) Events (c) DRBNet (d) IFAN (e) KPAC (f) APL (g) Ours

Figure 5. Visual quality comparison with image-based defocus deblurring methods on real data. (a) Defocused image. (b) Events. (c)∼(g) All-in-focus results of DRBNet [31], IFAN [18], KPAC [38], APL [46], and ours. More results are in the supplementary material.



(a) Defocused Image (b) Ours

(c) Dual-pixel Image (d) DPDNet

Figure 6. Visual quality comparison with a dual-pixel method. (a) Defocused image captured by machine vision camera. (b) All-in-focus image recovered by our method. (c) Dual-pixel image captured by Canon 5D Mark IV. (d) All-in-focus image recovered by DPDNet [1] using (c).

4.1. Quantitative comparison using synthetic data

As the majority of existing all-in-focus image datasets do not contain image focal stacks, we generate image focal stacks from a light field dataset, the Stanford Multiview Light Field (LiFF) Dataset [6], which was captured with hand-held Lytro Illum cameras. We synthesize the image focal stacks from their light field images and corresponding paired events are generated by DVS-Voltmeter [21]. The first image of each focal stack is selected as the input defocused image. Among all the synthetic triplet clips, consisting of defocused images, all-in-focus images, and corresponding events, we select 50 sets which are consistent with the LFDOF dataset [32] as our testing dataset for a fair comparison with other methods.

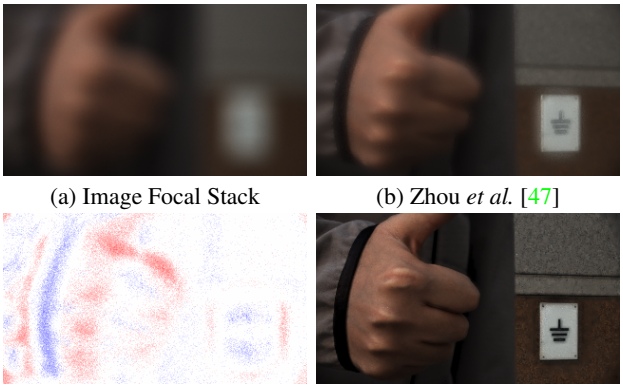
We compare our method with four recent image-based defocus deblurring methods: DRBNet [31], IFAN [18], KPAC [38], and APL [46]. We utilize Peak Signal-to-Noise Ratio (PSNR), Structural Similarity (SSIM), Multi-Scale Structural Similarity (MS-SSIM), and Learned Perceptual Image Patch Similarity (LPIPS) to measure the quality of restored images. The quantitative comparisons are shown in Table 1 and qualitative comparisons are shown in Figure 4. Our method outperforms other state-of-the-art methods with more than 15% improvement on all metrics, restores more high-frequency details encoded inside the event streams, and recovers an all-in-focus image with higher quality and fewer artifacts. Note that this comparison is a little unfair since image-based defocus methods only have one image as input. The purpose is to demonstrate a significant performance boost can be achieved when continuous information from event streams is involved.

4.2. Qualitative comparison using real data

To verify the effectiveness of our method in real-world scenarios, we capture real data by building a hybrid camera system, which consists of a machine vision camera (HIKVISION MV-CA050-12UC) and an event camera (PROPHESEE GEN4.0) with a beam splitter[3]. We synchronously capture an EFS and an RGB image focused at an arbitrary distance in both indoor and outdoor scenarios. Visual quality comparisons of all-in-focus results are shown in Figure 5. Our method can recover all-in-focus images with the correct texture in defocused regions. In comparison, other image-based methods cannot recover the sharp details well and even introduce undesired ringing artifacts.

With dual-pixel methods. Dual-pixel images are validated as effective inputs to recover an all-in-focus image [1, 2]. These methods also take additional input like our method. To compare the performance between EFS and dual-pixel imaging, we capture a real scenario with our hybrid camera system and a Canon 5D Mark IV DLSR camera. We compare with dual-pixel-based defocus deblurring method DPDNet [1], and the results are shown in Fig-

[3]Detailed setup can be found in the supplementary material.



(a) Image Focal Stack (b) Zhou *et al.* [47]

(c) EFS (d) Ours

Figure 7. Visual quality comparison with an image-based focal stack method. (a) Image focal stack. (b) All-in-focus image restored by Zhou *et al.* [47]. (c) The visualization of EFS. (d) All-in-focus image restored by ours. Please visit our project page for animated results.

ure 6. We can see our method outperforms DPDNet [1]. Thanks to the high temporal resolution information in event streams, our method recovers clearer texture information. Since the DLSR camera cannot be directly mounted on our beam splitter and the lens are also different, we cannot obtain the EFS and dual-pixel image with perfect spatial alignment. Thus the field of view and DOF in this example are somewhat inconsistent, but the levels of details recovered by these two methods are clearly different.

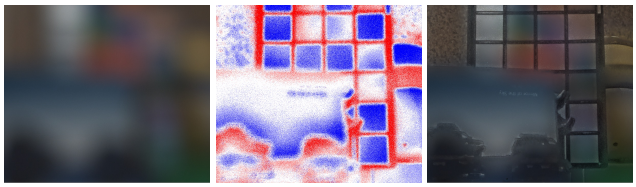
With image-based focal stack methods. Traditional image-based focal stack methods [3] need to capture multiple images with different focus distances, which is sensitive to camera shake. Although Zhou *et al.* [47] proposed a space-time refocusing method to stabilize the input images by selecting corresponding pixels in the focal stack, they still require that the velocity of focal sweep is constant, which limits the applicability of their method. Since we rotate the lens to capture the image/event focal stack, leading to unavoidable camera shake, we show that our method is robust to such slight motion and produces a sharper all-in-focus image, shown in Figure 7, while the result of Zhou *et al.* [47] shows ringing artifacts.

4.3. Ablation studies

To verify the effectiveness of the each part of our method, we conduct several ablation studies and show results in Table 2. We show the effectiveness of EvRefoucsNet by replacing it with ET-Net [45], an event-based image reconstruction method (denoted as "ET+MNet"). We further verify the contribution of EvMergeNet compared with all-in-focus imaging from gradient domain fusion [47] (denoted as "RNet+GDF"). Finally, we demonstrate the necessity of refocusing time selection by substituting it with uniform time selection (denoted as "Uniform"). As the results shown, our complete model achieves the best performance.

Table 2. Quantitative results of ablation study.

	PSNR↑	SSIM↑	MS-SSIM↑	LPIPS↓
ET+MNet	12.62	0.3474	0.2169	0.7179
RNet+GDF	32.66	0.9272	0.9556	0.1698
Uniform	32.84	0.9224	0.9564	0.1605
Ours	**33.25**	**0.9323**	**0.9611**	**0.1510**



(a) Defocused Image (b) Event (c) Ours

Figure 8. Failure case: Recovering an all-in-focus image from a significantly defocused color checkerboard.

5. Conclusion

In this paper, we propose a novel event focal stack to record intensity changes with respect to focus distance. With EFS, we introduce a reliable refocusing timestamp selection algorithm, and further design EvRefocusNet and EvMergeNet to recover an all-in-focus image. Thanks to successfully exploring the continuous focusing related information from EFS, our method exhibits superior performance over state-of-the-art methods.

Limitations. With our current hybrid camera system, our event cameras only record intensity changes in grayscale, which cannot reflect the difference across the RGB channels. This will cause failure cases as shown in Figure 8. It demonstrates that although our method can restore sharp edges of color checkerboards, the color space is not correctly recovered, as the input defocused image only preserves degenerated color information especially when it is significantly blurred. This brings difficulty for our network to compensate for an accurate color map. In our current implementation, we capture an EFS by rotating the lens manually, which is not fast enough to record scenes with object motions. Thus, our method is not applicable to dynamic scenes either. Extending our method with a fast focusing device (*e.g.*, liquid lens [24]) may be an option for dealing with scenes with motions, which is left as our future work.

Acknowledgement

This work was supported by National Key R&D Program of China (2021ZD0109803) and National Natural Science Foundation of China under Grant No. 62088102, 62136001.

References

[1] Abdullah Abuolaim and Michael S Brown. Defocus deblurring using dual-pixel data. In *Proc. of European Conference on Computer Vision*, 2020. 3, 7, 8

[2] Abdullah Abuolaim, Mauricio Delbracio, Damien Kelly, Michael S Brown, and Peyman Milanfar. Learning to reduce defocus blur by realistically modeling dual-pixel data. In *Proc. of International Conference on Computer Vision*, 2021. 3, 7

[3] Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen. Interactive digital photomontage. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH)*, pages 294–302, 2004. 8

[4] Blender Foundation. The Blender project - free and open 3D creation software. Accessed: 2022-11-04. 6

[5] Shoushun Chen and Menghan Guo. Live demonstration: CeleX-V: A 1m pixel multi-mode event-based sensor. In *Proc. of Computer Vision and Pattern Recognition Workshops*, 2019. 2

[6] Donald G. Dansereau, Bernd Girod, and Gordon Wetzstein. LiFF: Light field features in scale and depth. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 6, 7

[7] Edward R Dowski and W Thomas Cathey. Extended depth of field through wave-front coding. *Applied Optics*, 34(11):1859–1866, 1995. 3

[8] Peiqi Duan, Zihao W Wang, Boxin Shi, Oliver Cossairt, Tiejun Huang, and Aggelos K Katsaggelos. Guided event filtering: Synergy between intensity images and neuromorphic events for high performance imaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8261–8275, 2021. 5

[9] Laurent D'Andrès, Jordi Salvador, Axel Kochale, and Sabine Süsstrunk. Non-parametric blur map regression for depth of field extension. *IEEE Transactions on Image Processing*, 25(4):1660–1673, 2016. 1, 2

[10] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proc. of International Conference on Computer Vision*, 2017. 1

[11] Berthold Klaus Paul Horn. Focusing. Technical report, MIT, 1968. 2, 3, 4, 5

[12] Yuhuang Hu, Shih-Chii Liu, and Tobi Delbruck. v2e: From video frames to realistic DVS events. In *Proc. of Computer Vision and Pattern Recognition Workshops*, 2021. 3, 5

[13] Ali Karaali and Claudio Rosito Jung. Edge-based defocus blur estimation with adaptive scale selection. *IEEE Transactions on Image Processing*, 27(3):1126–1137, 2017. 1, 2

[14] Jack Kiefer. Sequential minimax search for a maximum. In *Proc. of the American Mathematical Society*, 1953. 5

[15] Diederik P Kingma and Jimmy Ba. ADAM: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[16] Sujit Kuthirummal, Hajime Nagahara, Changyin Zhou, and Shree K Nayar. Flexible depth of field photography. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):58–71, 2010. 3, 4

[17] Junyong Lee, Sungkil Lee, Sunghyun Cho, and Seungyong Lee. Deep defocus map estimation using domain adaptation. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 2

[18] Junyong Lee, Hyeongseok Son, Jaesung Rim, Sunghyun Cho, and Seungyong Lee. Iterative filter adaptive network for single image defocus deblurring. In *Proc. of Computer Vision and Pattern Recognition*, 2021. 2, 6, 7

[19] Anat Levin, Rob Fergus, Frédo Durand, and William T Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH)*, 26(3):70, 2007. 3

[20] Anat Levin, Samuel W Hasinoff, Paul Green, Frédo Durand, and William T Freeman. 4D frequency analysis of computational cameras for depth of field extension. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH)*, 28(3):1–14, 2009. 3

[21] Songnan Lin, Ye Ma, Zhenhua Guo, and Bihan Wen. DVS-Voltmeter: Stochastic process-based event simulator for dynamic vision sensors. In *Proc. of European Conference on Computer Vision*, 2022. 5, 6, 7

[22] Shijie Lin, Yinqiang Zhang, Lei Yu, Bin Zhou, Xiaowei Luo, and Jia Pan. Autofocus for event cameras. In *Proc. of Computer Vision and Pattern Recognition*, 2022. 5

[23] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Proc. of European Conference on Computer Vision*, 2014. 6

[24] Carlos A López and Amir H Hirsa. Fast focusing using a pinned-contact oscillating liquid lens. *Nature Photonics*, 2(10):610–613, 2008. 8

[25] Marcella Matrecano, Melania Paturzo, and Pietro Ferraro. Extended focus imaging in digital holographic microscopy: a review. *Optical Engineering*, 53(11):112317, 2014. 1

[26] Gottfried Munda, Christian Reinbacher, and Thomas Pock. Real-time intensity-image reconstruction for event cameras using manifold regularisation. *International Journal of Computer Vision*, 126(12):1381–1393, 2018. 3

[27] Jinsun Park, Yu-Wing Tai, Donghyeon Cho, and In So Kweon. A unified approach of multi-scale deep and hand-crafted features for defocus estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2017. 2

[28] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6):1964–1980, 2019. 2, 3

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Adv. in Neural Information Processing Systems*, 2015. 1

[30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Proc. of International Conference on Medical Image Computing and Computer Assisted Intervention*, 2015. 5

[31] Lingyan Ruan, Bin Chen, Jizhou Li, and Miuling Lam. Learning to deblur using light field generated and real defocus images. In *Proc. of Computer Vision and Pattern Recognition*, 2022. 1, 2, 6, 7

[32] Lingyan Ruan, Bin Chen, Jizhou Li, and Miu-Ling Lam. AIFNet: All-in-focus image restoration network using a light field-based dataset. *IEEE Transactions on Computational Imaging*, 7:675–688, 2021. 2, 7

[33] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 6

[34] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. Continuous-time intensity estimation using event cameras. In *Proc. of Asian Conference on Computer Vision*, 2018. 3

[35] Teresa Serrano-Gotarredona and Bernabé Linares-Barranco. A 128×128 1.5% contrast sensitivity 0.9% FPN 3 μs latency 4 mw asynchronous frame-free dynamic vision sensor using transimpedance preamplifiers. *IEEE Journal of Solid-State Circuits*, 48(3):827–838, 2013. 2

[36] Jianping Shi, Li Xu, and Jiaya Jia. Just noticeable defocus blur detection and estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2015. 1, 2

[37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. of International Conference on Learning Representations*, 2015. 6

[38] Hyeongseok Son, Junyong Lee, Sunghyun Cho, and Seungyong Lee. Single image defocus deblurring using kernel-sharing parallel atrous convolutions. In *Proc. of International Conference on Computer Vision*, pages 2642–2650, 2021. 2, 6, 7

[39] 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 *Proc. of European Conference on Computer Vision*, 2022. 5

[40] Supasorn Suwajanakorn, Carlos Hernandez, and Steven M Seitz. Depth from focus with your mobile phone. In *Proc. of Computer Vision and Pattern Recognition*, 2015. 2, 5

[41] Minggui Teng, Chu Zhou, Hanyue Lou, and Boxin Shi. NEST: Neural event stack for event-based image enhancement. In *Proc. of European Conference on Computer Vision*, 2022. 2

[42] Stepan Tulyakov, Alfredo Bochicchio, Daniel Gehrig, Stamatios Georgoulis, Yuanyou Li, and Davide Scaramuzza. Time Lens++: Event-based frame interpolation with parametric non-linear flow and multi-scale fusion. In *Proc. of Computer Vision and Pattern Recognition*, 2022. 2

[43] Stepan Tulyakov, Daniel Gehrig, Stamatios Georgoulis, Julius Erbach, Mathias Gehrig, Yuanyou Li, and Davide Scaramuzza. Time Lens: Event-based video frame interpolation. In *Proc. of Computer Vision and Pattern Recognition*, 2021. 2

[44] Lin Wang, Yo-Sung Ho, Kuk-Jin Yoon, et al. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 5

[45] Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Event-based video reconstruction using transformer. In *Proc. of International Conference on Computer Vision*, 2021. 3, 8

[46] Wenda Zhao, Fei Wei, You He, and Huchuan Lu. United defocus blur detection and deblurring via adversarial promoting learning. In *Proc. of European Conference on Computer Vision*, 2022. 2, 6, 7

[47] Changyin Zhou, Daniel Miau, and Shree K. Nayar. Focal sweep camera for space-time refocusing. Technical report, Columbia University, 2012. 2, 3, 4, 6, 8

[48] Changyin Zhou and Shree Nayar. What are good apertures for defocus deblurring? In *Proc. of International Conference on Computational Photography*, 2009. 1

[49] Lin Zhu, Xiao Wang, Yi Chang, Jianing Li, Tiejun Huang, and Yonghong Tian. Event-based video reconstruction via potential-assisted spiking neural network. In *Proc. of Computer Vision and Pattern Recognition*, 2022. 3