2023 IEEE/CVF International Conference on Computer Vision (ICCV) | 979-8-3503-0718-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCV51070.2023.01219

E^{2}NeRF: Event Enhanced Neural Radiance Fields from Blurry Images

Yunshan Qi[1] Lin Zhu[2*] Yu Zhang[3] Jia Li[1,4*]

[1]State Key Laboratory of Virtual Reality Technology and Systems, SCSE, Beihang University
[2]Beijing Institute of Technology [3]SenseTime and Tetras.AI [4]Peng Cheng Laboratory

{qi_yunshan, jiali}@buaa.edu.cn, linzhu@bit.edu.cn, zhangyulb@gmail.com

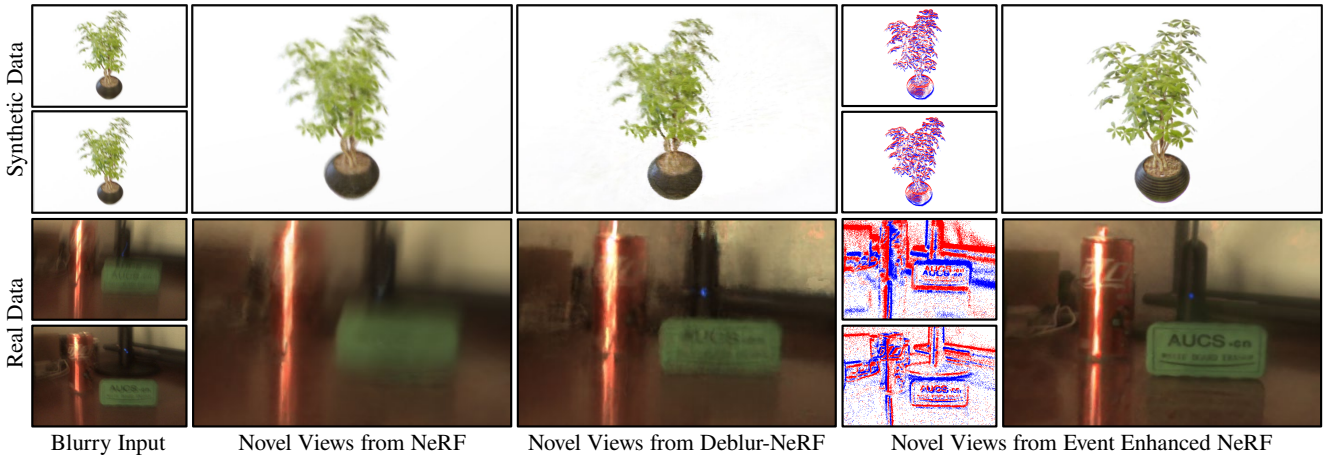

Figure 1: Given a set of blurry images from multiple views of an object or a scene, the novel view rendering results of original NeRF [27] is severely blurred and the performance of Deblur-NeRF [24] is also limited. In contrast, our approach leverages event data to significantly enhance the learning of neural 3D representation even when the input images are highly blurred. Consequently, the rendered views of the object or scene are much sharper.

Abstract

Neural Radiance Fields (NeRF) achieves impressive rendering performance by learning volumetric 3D representation from several images of different views. However, it is difficult to reconstruct a sharp NeRF from blurry input as often occurred in the wild. To solve this problem, we propose a novel Event-Enhanced NeRF (E^{2}NeRF) by utilizing the combination data of a bio-inspired event camera and a standard RGB camera. To effectively introduce event stream into the learning process of neural volumetric representation, we propose a blur rendering loss and an event rendering loss, which guide the network via modelling real blur process and event generation process, respectively. Moreover, a camera pose estimation framework for real-world data is built with the guidance of event stream to generalize the method to practical applications. In contrast to previous image-based or event-based NeRF, our framework effectively utilizes the internal relationship between events and images. As a result, E^{2}NeRF not only achieves image deblurring but also achieves high-quality novel view image generation. Extensive experiments on both synthetic data and real-world data demonstrate that E^{2}NeRF can effectively learn a sharp NeRF from blurry images, especially in complex and low-light scenes. Our code and datasets are publicly available at `https://github.com/iCVTEAM/E2NeRF`.

1. Introduction

With the proposal of Neural Radiance Fields (NeRF) [27], significant progress has been made in neural 3D representation and novel view synthesis tasks in the past few years. NeRF takes 3D location and 2D view direction as input and uses multi-view images of objects or scenes as supervision to learn the neural volumetric representation,

*Correspondence should be addressed to Lin Zhu and Jia Li. Website: `https://cvteam.buaa.edu.cn`

2380-7504/23/$31.00 ©2023 IEEE
DOI 10.1109/ICCV51070.2023.01219

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

which is parameterized as a multilayer perceptron (MLP). To generate high-fidelity novel view images, NeRF uses volume rendering techniques with the output of the network (color and density) to render each pixel.

The premise that NeRF is capable of producing impressive results relies on the underlying assumption that the input image quality is of high standards, devoid of any blurs and has sufficient lighting. However, in many real-world settings, obtaining such high-quality images can be challenging. For instance, capturing a handheld shot can cause motion blurs, especially in low-light conditions, which require increasing the exposure time of the camera to collect enough luminance information of the scene, consequently leading to blurred images. Deblur-NeRF [24] solves this issue and proposes the deformable sparse kernel to model image blurring. Though this method attempts to mitigate the impact of motion blurs, it could fail in scenarios where the camera shakes in roughly the same direction across all views or the blur of the image is very severe.

In contrast to only relying on blurry images, combining additional information to guide neural radiance fields learning process is promising. Event camera is a new bio-inspired vision sensor, which measures the brightness changes of each pixel asynchronously. Compared to traditional frame-based cameras, event cameras can record high temporal resolution and high dynamic range information of the scene, which is important for modelling the blurring process. Therefore, event-based image deblurring has become a very attractive research topic in recent years [16, 20, 31, 35, 41]. Inspired by this, we introduce event stream into the learning process of neural volumetric representation to solve the problem caused by blurry input.

In this paper, we propose E^2NeRF to learn sharp 3D volumetric representation with blurry images and the corresponding event data. We introduce two novel losses into NeRF framework to enhance volumetric representation learning: Firstly, during the training process, we predict a blurry image with the poses and compare it to the input image to obtain our blur rendering loss. Additionally, the generation process of events is simulated along with the change of camera pose to simulate the event data from predicted sharp images. With the actual event data as supervision, we develop a novel event rendering loss to refine the neural 3D representation learning. To process real-world data efficiently, we design a camera pose estimation framework to guide the estimation of pose sequences of the blurry images, making our method robust for severe blurry images. Due to the augmentation of the network with event data, we can learn a sharp NeRF, which not only achieves deblurring of the input image but also achieves high-quality novel view generation when the quality of the input image is degraded. We conduct experiments on both simulated data and real data and achieve satisfactory results.

To the best of our knowledge, this is the first work to reconstruct a sharp NeRF using both event data and RGB data. The event data can effectively enhance the robustness of NeRF to complex scenes such as motion blur. Our contributions can be summarized as follows:

1) We propose an Event-Enhanced Neural Radiance Fields (E^2NeRF), the first framework for reconstructing a sharp NeRF from blurry images and corresponding event data. Unlike previous image-based or event-based NeRF, our framework effectively exploits the internal relationship between events and images, which significantly enhances the performance and robustness of NeRF.

2) We develop a blur rendering loss and an event rendering loss, which are effective in enhancing neural volumetric representation learning. Furthermore, an event-image-based pose estimation framework that can estimate the sequence of camera poses corresponding to a blurry image is designed for real-world data with severe blur.

3) We build both synthetic and real-world datasets for training and testing our model. Experimental results demonstrate that our method outperforms existing methods. Additionally, we propose a benchmark for future research on NeRF reconstruction from blurry images and event stream.

2. Related work

2.1. Neural radiance fields

In the past few years, NeRF has achieved impressive results and attracted a lot of attention for tasks of neural implicit 3D representation and novel view synthesis. Many improvements have been made to NeRF, such as FastNeRF [7] and Depth-supervised NeRF [6], which aim to improve the learning speed of NeRF. Neural scene flow fields [18] explores 3D scene representation learning of dynamic scenes. PixelNeRF [44] and RegNeRF [29] try to use a small number of input images to achieve high-quality novel view synthesis. Mip-NeRF [3] proposes a frustum-based sampling strategy to implement NeRF-based anti-aliasing, which solves the problem of artifacts and improves the training speed. In addition, some works aim to improve NeRF with low-quality input images. NeRF in the wild [25] uses low-quality images captured by tourists as input and trains NeRF under conditions where input images are occluded and the lighting environment is inconsistent. NeRF in the dark [26] and HDR-NeRF [14] enable the synthesis of high dynamic range novel view images from noisy and low dynamic images. Moreover, Deblur-NeRF [24] proposes the deformable sparse kernel to simulate the blurring process, which realizes sharp novel view synthesis from blurry images. However, it may fail when the camera coincidentally shakes in roughly the same direction across all views or the input images have a strong blur.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

2.2. Image deblurring

A blurred image can be expressed as a sharp image multiplied by a blur kernel plus noise. However, due to the non-uniqueness of the blur kernel, the deblurring problem becomes ill-posed. In order to solve this problem, traditional algorithms use hand-crafted or sparse priors to predict the blur kernel [5, 17, 43]. With the development of deep learning, some works have attempted to learn end-to-end mapping directly from blurry to sharp images using neural networks [38, 40, 48]. Zamir *et al.* [45] introduce a novel per-pixel adaptive design to reweight the local features and uses encoder-decoder architectures, which achieves state-of-the-art performance for single-image deblurring.

However, in real-world scenarios, the occurrence of motion blur is intricate and varies in nature, which can affect the generalization of learning blurring processes using hand-crafted prior and deep networks. Therefore, it is challenging for deblurring algorithms to perfectly recover a sharp image based only on blurry image data. Traditional cameras can only capture brightness information at a fixed frame rate, which leads to the absence of information on pixel changes during the motion blur.

2.3. Event camera

Dynamic vision sensor (DVS) [19], also known as event camera, can generate an event when the brightness change of each pixel reaches a threshold. This framework gathers asynchronous brightness change information and effectively overcomes the problem of information loss between frames in traditional cameras. Dynamic active vision sensor (DAVIS) [4] realizes the simultaneous acquisition of RGB images and events which attracts widespread attention in the computer vision community. At present, event cameras have achieved remarkable results in optical flow estimation [2, 8, 11, 30, 36], depth estimation [1, 12, 37, 50], feature detection and tracking [42, 46, 47] and simultaneous localization and mapping [9, 21, 22]. In addition, to address the lack of event-based datasets, some event simulators [10, 13, 32] are designed to simulate events through videos. With the high temporal resolution of the event camera, event data has significant advantages in image deblurring. Pan *et al.* [31] propose an event-based double integral model, which realizes the event-rgb-based image deblurring. Jiang *et al.* [16] propose a convolutional recurrent neural network that integrates visual and temporal knowledge from both global and local scales, which generalizes better for handling real-world motion blur. Shang *et al.* [35] develop D2Net for video deblurring with events and propose a flexible event fusion module (EFM) to bridge the gap between event and video deblurring.

Recently, Ev-NeRF [15] and EventNeRF [33] have proposed neural radiance fields derived only from the event stream. However, Ev-NeRF [15] can only learn a grayscale NeRF and the results of EventNeRF [33] has noticeable artifacts and chromatic aberration without RGB data supervising. Besides, both of these works have limited generalization on pose estimation for neural representation learning.

To the best of our knowledge, we are the first to use both blurry RGB images and corresponding event data to train a sharp NeRF. Our approach emphasizes event representation in blurred images, encompassing pose estimation and a unique blur-solving method, which has better results and stronger generalization for real-world complex scenes.

3. Background

3.1. Neural radiance fields

The core of NeRF is to learn 3D volume representation via MLP. Its input is 3D position $\mathbf{o}$ and 2D ray direction $\mathbf{d}$, and the output is color $\mathbf{c}$ and density σ. As shown in Eq. (1), F_θ is the MLP network and θ is parameters of the network:

$$(\mathbf{c}, \sigma) = F_\theta(\gamma_o(\mathbf{o}), \gamma_d(\mathbf{d})). \tag{1}$$

The $\gamma_o(\cdot)$ and $\gamma_d(\cdot)$ functions are defined in Eq. (2), which map the input 5D coordinates into a higher dimension space. The encoder enables the neural network to better learn the color and geometry information in the scene. And we set $M = 10$ for position $\mathbf{o}$, $M = 4$ for direction $\mathbf{d}$:

$$\gamma_M(x) = \{\sin(2^m \pi x), \cos(2^m \pi x)\}_{m=0}^{M}. \tag{2}$$

To get images of different views from the implicit 3D scene representation, NeRF uses the classical volume rendering method as shown in Eq. (3). For a given ray $\mathbf{r}(l) = \mathbf{o} + l\mathbf{d}$ emitting from camera center$\mathbf{o}$ and dicrection $\mathbf{d}$, its expected color projected on the pixel is $C(\mathbf{r})$. NeRF divides $[l_n, l_f]$ into N discrete bins. l_n, l_f are the near and far bounds of the ray. $\mathbf{c}_i$, σ_i are the output of F_θ, indicating the color and density of each bin through which the ray passes. $\delta_i = l_{i+1} - l_i$ is the distance between adjacent samples. T_i is the transparency of the particles between l_n and bin i.

$$\begin{aligned} C(\mathbf{r}) &= \sum_{i=1}^{N} T_i (1 - \exp(-\sigma_i \delta_i)) \mathbf{c}_i, \\ \text{where} \quad T(i) &= \exp(-\sum_{j=1}^{i-1} \sigma_j \delta_j). \end{aligned} \tag{3}$$

$$\mathcal{L} = \sum_{\mathbf{r} \in \mathcal{R}} [\|C_c(\mathbf{r}) - C(\mathbf{r})\|_2^2 + \|C_f(\mathbf{r}) - C(\mathbf{r})\|_2^2]. \tag{4}$$

In order to achieve reasonable sampling for the final model, NeRF uses the hierarchical volume sampling strategy, which optimizes the coarse model and the fine model at the same time ($C_c(\mathbf{r})$ and $C_f(\mathbf{r})$ in Eq. (4)), and applies the

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

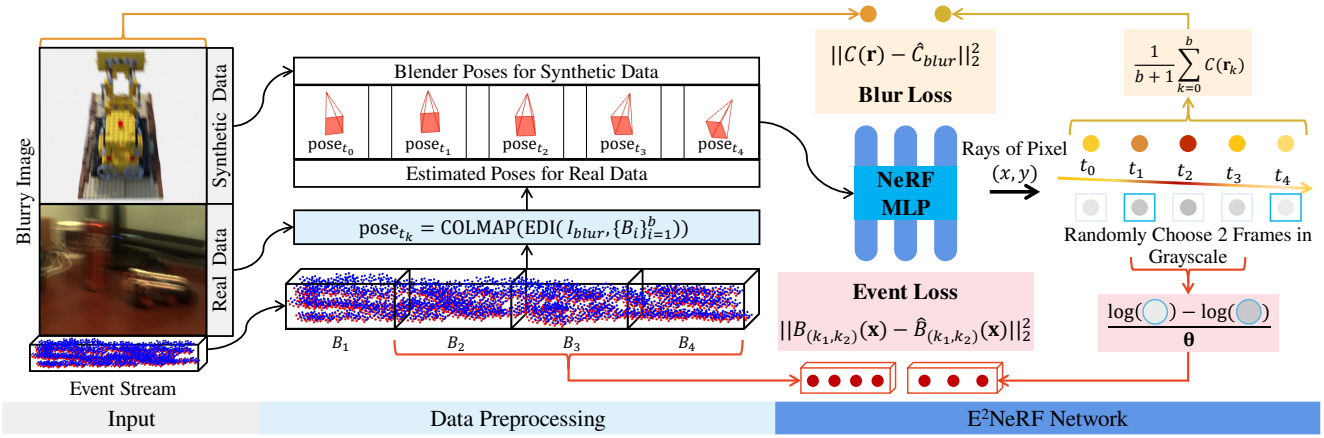

Figure 2: The architecture of E^{2}NeRF. The input is a blurry image and its corresponding event stream of one of the views. For real data, we use the position estimation model to obtain the pose sequence. Then the poses are sent to the E^{2}NeRF network. After rendering we get the color of each pose on pixel (x, y) and calculate the predicted blurry color $\hat{C}_{blur}$ and event bin $\hat{B}_{(k_1,k_2)}(\mathbf{x})$. Then with input color $C(\mathbf{r})$ and event bin $B_{(k_1,k_2)}$ we get proposed blur loss and event loss as supervision.

density obtained by the coarse model to determine the sampling weight of the fine model. The final loss of NeRF is a mean squared loss between the predicted color and ground truth color for both the coarse model and fine model. $\mathcal{R}$ is the set of rays in each batch.

3.2. Event generation model

Unlike frame-based cameras that record the brightness of each pixel at a fixed frame rate, the event camera asynchronously generates an event $\mathbf{e}(x, y, \tau, p)$ when the changes of the brightness of pixel (x, y) reach threshold Θ in the log domain at time τ. As shown in Eq. (5), p indicates the direction of brightness change, $\mathcal{I}_{(x,y,\tau)}$ is the brightness value of pixel (x, y) at time τ.

$$p_{x,y,\tau} = \begin{cases} -1, & \log(\mathcal{I}_{x,y,\tau}) - \log(\mathcal{I}_{x,y,\tau-\Delta\tau}) < -\Theta, \\ +1, & \log(\mathcal{I}_{x,y,\tau}) - \log(\mathcal{I}_{x,y,\tau-\Delta\tau}) > \Theta. \end{cases} \tag{5}$$

$$B_k = \{\mathbf{e}_i(x_i, y_i, \tau_i, p_i)\}_{t_{k-1}<\tau_i\leq t_k}. \tag{6}$$

Since the event camera does not have the concept of frame, to facilitate processing and representation, we usually divide the events into b event bins by time. In this work, given a blurred image with exposure time from t_{start} to t_{end} and the corresponding event data $\{\mathbf{e}_i\}_{t_{start}<\tau_i\leq t_{end}}$, we can generate $\{B_k\}_{k=1}^{b}$ as defined in Eq. (6), where $t_k = t_{start} + \frac{k}{b}t_{exp}$ is the time division point between bins and $t_{exp} = t_{start} - t_{end}$ is exposure time.

4. Method

Fig. 2 illustrates the overall architecture of our E^{2}NeRF. We introduce two novel losses into NeRF framework to enhance the volumetric representation and design an event-image-based pose estimation framework to efficiently process the real-world data. E^{2}NeRF takes blurry image I_{blur} and the corresponding event bins $\{B_k\}_{k=1}^{b}$ as input of each view. Blur rendering loss simulates the process of blurring image generation and provides more information about texture details of scenes to the network. Event rendering loss introduces event data into the NeRF training process, enabling the network to better learn the real 3D volume representation. Similar to NeRF, we use image poses in Blender to train the network for synthetic data. For real-world data, a pose estimation framework based on event-image pairs is designed to obtain the pose sequences for the network.

4.1. Blur rendering loss

In order to adapt NeRF to taking blurry images as input, we propose blur rendering loss. With $b+1$ poses $\{\mathbf{P}_k\}_{k=0}^{b}$ of each view, we can get $b+1$ rays $\{\mathbf{r}_k\}_{k=0}^{b}$ emitted from each pixel. And through the NeRF network, we can get $b+1$ color values $\{\hat{C}_k = C(\mathbf{r}_k)\}_{k=0}^{b}$ of the pixel as the process of blurry pixel generation. We regard the average of the results as the predicted blurry color:

$$\hat{C}_{blur} = \frac{1}{b+1}\sum_{k=0}^{b} C(\mathbf{r}_k). \tag{7}$$

$$\mathcal{L}_{blur} = \sum_{\mathbf{r}\in\mathcal{R}} [\|\hat{C}^c_{blur} - C(\mathbf{r})\|_2^2 + \|\hat{C}^f_{blur} - C(\mathbf{r})\|_2^2]. \tag{8}$$

In this way, the loss function of NeRF with blurry images as input is converted into Eq. (8). We adopt the design of the joint optimization of NeRF's coarse model and fine model, which is still beneficial in our framework.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

4.2. Event rendering loss

Blur loss only uses discrete $b+1$ frames corresponding to $b+1$ poses in a blurry image to simulate blurry process. However, the generation of image blur is a continuous process. With the high temporal resolution of event data, we propose event loss, which utilizes event information to supervise the continuous blurring process between any two predicted frames.

Given a pixel $\mathbf{x} = (x, y)$, we first randomly select the estimated values of two moments from $\{\hat{C}_k\}_{k=0}^{b}$ as C_{k_1} and C_{k_2} $(k_1 < k_2)$ at this pixel and convert them into grayscale values to get L_{k_1}, L_{k_2}. We take the difference of the two values in the log domain and divide it by the threshold Θ. Then an estimate of the number of events between two frames for the given pixel $\mathbf{x}$ is obtained:

$$\hat{B}_{(k_1,k_2)}(\mathbf{x}) = \begin{cases} \lceil \dfrac{\log(L_{k_2}) - \log(L_{k_1})}{\Theta_{neg}} \rceil, L_{k_2} < L_{k_1}, \\ \lfloor \dfrac{\log(L_{k_2}) - \log(L_{k_1})}{\Theta_{pos}} \rfloor, L_{k_2} \geq L_{k_1}. \end{cases} \tag{9}$$

We use the mean squared error between the estimated number of events $\hat{B}_{(k_1,k_2)}(\mathbf{x})$ and the actual number of events $B_{(k_1,k_2)}(\mathbf{x})$ from $\{B_k\}_{k=1}^{b}$ as our event loss. Note that in the event bin of pixel $\mathbf{x}$ $B_k(\mathbf{x})$, we set the number of negative events as its additive inverse so that the positive event and the negative event can cancel each other out when we add the event bins. $\mathcal{X}$ is the set of pixels in each batch.:

$$\begin{aligned} \mathcal{L}_{event} &= \sum_{\mathbf{x}\in\mathcal{X}} \|\hat{B}_{(k_1,k_2)}(\mathbf{x}) - B_{(k_1,k_2)}(\mathbf{x}))\|_2^2, \\ \text{where} \quad B_{(k_1,k_2)}(\mathbf{x}) &= \sum_{k=k_1+1}^{k_2} B_k(\mathbf{x}). \end{aligned} \tag{10}$$

$$\mathcal{L} = \mathcal{L}_{blur} + w\mathcal{L}_{event}. \tag{11}$$

Our final loss function defines as in Eq. (11), where w is the weight parameter. With the event loss, our E^{2}NeRF can learn the 3D volumetric representation more precisely. We will analyze specifically in Sec. 5.4.

4.3. Position estimation

In general, NeRF utilizes the ground truth camera poses in Blender with synthetic data. For real data, COLMAP [34] is used to estimate the camera poses. However, when the input image becomes blurred, the pose estimation of COLMAP will fail, which is also a problem that has not been solved by the Debur-NeRF. Therefore, for real captured data, we refer to the EDI [31] to perform the initial deblurring of the blurry images and then input the results into COLMAP to get the poses during the blurring process.

The EDI model uses event data to convert a single blurry image into multiple time-sequenced sharp images. We simplify its formulation to a discrete version. Given a blurred image I_{blur} and the corresponding event bins $\{B_k\}_{k=1}^{b}$. We assume that the sharp image at t_{start} is I_0, according to Eq. (5), the sharp image I_k at the moment t_k of dividing each event bin can be expressed as:

$$I_k = I_0 e^{\Theta \sum_{i=1}^{k} B_i}, (k > 0). \tag{12}$$

According to the general model of image formation, we can assume that a blur map is time-weighted from multiple images. Since our exposure time is equally divided into b parts in this paper, the blurry image can be directly regarded as the average of these images:

$$\begin{aligned} I_{blur} &= \frac{1}{b+1} \sum_{k=0}^{b} I_k \\ &= \frac{I_0}{b+1} (1 + e^{\Theta \sum_{i=1}^{1} B_i} + \cdots + e^{\Theta \sum_{i=1}^{b} B_i}). \end{aligned} \tag{13}$$

Then I_0 can be expressed as Eq. (14). According to Eq. (12), we can get the rest of the sharp images $\{I_k\}_{k=1}^{b}$ during the blurring process as Eq. (15). Next we feed $\{I_k\}_{k=0}^{b}$ into COLMAP to get $b+1$ poses $\{\mathbf{P}_k\}_{k=0}^{b}$:

$$I_0 = \frac{(b+1) I_{blur}}{1 + e^{\Theta \sum_{i=1}^{1} B_i} + \cdots + e^{\Theta \sum_{i=1}^{b} B_i}}. \tag{14}$$

$$I_k = \frac{(b+1) I_{blur} e^{\Theta \sum_{i=1}^{k} B_i}}{1 + e^{\Theta \sum_{i=1}^{1} B_i} + \cdots + e^{\Theta \sum_{i=1}^{b} B_i}}. \tag{15}$$

$$\{\mathbf{P}_k\}_{k=0}^{b} = \text{COLMAP}(\{I_k\}_{k=0}^{b}). \tag{16}$$

The event-image-based pose estimation framework enhances the robustness against real-world data with severe blur and generalizes our method to practical applications.

4.4. Implement details

Our code is based on NeRF [27]. We train each scene with 200k iterations on a single NVIDIA RTX3090 GPU. For all data, we set $w = \frac{1}{625}$ and $b = 4$. We take the batch size as 1024 rays. The rest of the parameters are the same as NeRF default values. We set the positive threshold $\Theta_{pos} = 0.2$ and negative threshold $\Theta_{neg} = 0.3$. For the synthetic data, we use the poses from Blender. For each view, we select 5 poses between 4 equal time intervals during the blurring process and enter them into the network in chronological order. For real data, we only have the blur images and the corresponding events, so we use the position estimation model to get the 5 poses.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

Blur View	NeRF	Deblur-NeRF	E^{2}NeRF^{25}	D2Net	D2Net-NeRF	EDI	EDI-NeRF	MPR	MPR-NeRF	E^{2}NeRF
PSNR↑	22.91	21.71	29.07	27.81	27.46	27.88	27.94	28.12	27.93	**29.77**
SSIM↑	.9072	.8795	.9535	.9517	.9450	.9451	.9497	.9548	.9525	**.9600**
LPIPS↓	.1441	.2364	.0887	.0867	.1029	.0860	.0746	.0865	.0882	**.0725**

Table 1: Quantitative analysis on blur view. The results in the table are the averages of the six synthetic scene from NeRF. We use **bold** to mark the best data. E^{2}NeRF^{25} represents training E^{2}NeRF with only 25 blurry images as in Deblur-NeRF.

Novel View	NeRF	Deblur-NeRF	E^{2}NeRF^{25}	D2Net-NeRF	EDI-NeRF	MPR-NeRF	E^{2}NeRF
PSNR↑	22.27	19.93	29.14	26.65	27.71	27.91	**29.56**
SSIM↑	.9018	.8584	.9573	.9427	.9522	.9571	**.9627**
LPIPS↓	.1483	.2573	.0895	.1087	.0896	.0861	**.0726**

Table 2: Quantitative analysis on novel view. The results in the table are the averages of the six synthetic scene from NeRF. We use **bold** to mark the best data. E^{2}NeRF^{25} represents training E^{2}NeRF with only 25 blurry images as in Deblur-NeRF.

Number of obtained poses	letter	lego	camera	plant	toys
COLMAP	24	14	25	25	27
Our Framework	30	30	30	30	30

Table 3: Number of obtained poses from 30 blurry images of each real-world scene.

5. Experiment

5.1. Dataset

Synthetic data. We extend the six synthetic scenes: chair, ficus, hotdog, lego, materials and mic in NeRF, and use the Camera Shakify plugin in Blender to simulate camera shake. For each viewpoint, we render 17 sharp images taken by the camera during the shaking process and record their corresponding poses. Then input these 17 images into the event simulation tool V2E [13] to simulate the event data generated by the camera shake process. In addition, in order to get the simulated blurred image, we first use inverse isp processing to transfer 17 images into the raw domain and superimpose them. Then we use isp processing to obtain the final blurred image. Each scene has 100 views of blurry images and the corresponding event data.

Real-world data. We use the DAVIS346 color event camera [39] to capture the real data. The camera is capable of capturing spatial-temporal aligned event data and RGB frames. The resolution of the camera is 346×260 and exposure time is set to 100ms for the RGB frames. We hold the camera by hand and capture five challenging scenes (letter, lego, camera, plant and toys), which contain rich color and texture details in a low-light environment (5-100 lux). Each scene has 30 images with varying degrees of blur on different views and the corresponding event data.

5.2. Comparison methods

For comparison, we first chose Deblur-NeRF [24], which is the first method to learn a sharp NeRF from blurry images. Additionally, we utilize two state-of-the-art deblurring methods: MPR [45], a single-image deblurring method, and D2Net [35], an event-based deblurring method, to deblur the input blurry images. We also compared our method to EDI [31] in order to verify the effectiveness of our framework. Following this, we train NeRF with these deblurred images and named them MPR-NeRF, D2Net-NeRF, and EDI-NeRF.

For the synthetic data, the camera coincidentally shakes in roughly the same direction across all views. As mentioned in Sec. 2.1, Deblur-NeRF will fail when we train with the full 360° of 100 views images. Therefore, we only input 25 blurry images of the scene of 180° views when we train Deblur-NeRF. For a fair comparison, we use the same input with our E^{2}NeRF and named as E^{2}NeRF^{25} in Tab. 1 and Tab. 2. We discuss the quantitative comparison in Sec. 5.3.

For real-world data, every scene comprises a total of 30 poses that need to be estimated from a set of 30 blurry images. As shown in Tab. 3, COLMAP [34] fails to estimate some poses corresponding to the severely blurred images. In contrast, our framework successfully estimates all the poses, which proves the robustness of our approach. For a fair comparison, in our experiments on real-world data, all NeRF-based methods utilize the poses obtained by our pose estimation framework as input poses for the network

5.3. Quantitative analysis

Synthetic data. As shown in Tab. 1 and Tab. 2, we divide the experimental results into two groups: blur view and novel view (blur view is a perspective of input blurry images, while novel view does not have any input image for

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

Blur View & Novel View	NeRF	D2Net-NeRF	MPR-NeRF	Deblur-NeRF	EDI-NeRF	E^{2}NeRF
BRISQUE↓	44.66	42.61	41.17	38.41	31.98	**30.26**
RankIQA↓	5.464	4.693	4.563	4.165	3.936	**3.609**

Table 4: Quantitative analysis on real-world data. The results are the averages of five scenes on blur view and novel view.

	-	$\mathcal{L}_{event}$	$\mathcal{L}_{blur}$	$\mathcal{L}_{event}$&$\mathcal{L}_{blur}$(Ours)
PSNR↑	22.59	27.22	28.68	29.67
SSIM↑	.9045	.9437	.9543	.9614
LPIPS↓	.1462	.1222	.0830	.0725

Table 5: Ablation study on blur loss and event loss. The results are averages of the results on blur and novel view.

Ablation Study	Blur View E^{2}NeRF*	Blur View E^{2}NeRF	Blur View Δ	Novel View E^{2}NeRF*	Novel View E^{2}NeRF	Novel View Δ
PSNR↑	29.38	29.77	1.3% ↑	27.98	29.56	5.6% ↑
SSIM↑	.9567	.9600	0.3% ↑	.9519	.9627	1.1% ↑
LPIPS↓	.0812	.0725	10.7% ↓	.0848	.0726	14.4% ↓

Table 6: Analysis of event loss on blur and novel view. E^{2}NeRF* denotes E^{2}NeRF without event loss supervision.

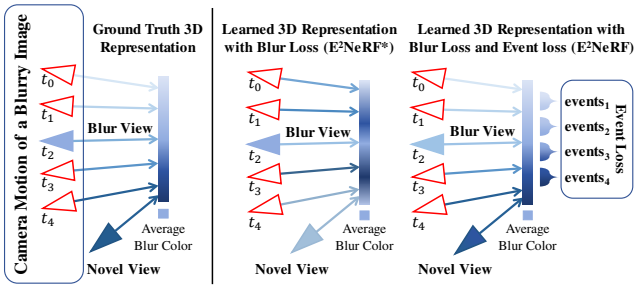

Figure 3: Effect of event rendering loss. E^{2}NeRF* denotes E^{2}NeRF without event loss supervision. As shown in figure, E^{2}NeRF* and E^{2}NeRF both get the right blur color with their learned 3D representation. But without event loss which can supervise the light intensity change, E^{2}NeRF* tend to learn a wrong 3D representation. And novel view is more sensitive to the wrong 3D representation because there is no direct constraint like blur view during the training. In other words, the blur view is overfitting in E^{2}NeRF* which is more prone to artifacts especially in novel view.

reference). We use PSNR, SSIM and LPIPS [49] to evaluate the results. And, we compare the results of three deblurring methods on blur view only. In the blur view experiments in Tab. 1, our method achieves the best average results and has significant improvement over all other methods. With only 25 views of the scene of 180° as input, E^{2}NeRF^{25} has only slight performance degradation on both blur view and novel view. Deblur-NeRF's performance is unsatisfactory due to its limitations on view consistent blur. We find that EDI-NeRF is better than EDI, although D2Net-NeRF and MPR-NeRF will cause performance degradation on blur view compared with D2Net and MPR. This is mainly because the results of EDI has more noise and NeRF training process weakens the impact of noise. But the results of EDI-NeRF will have color deviation which reduce its performance. On the novel view experiments in Tab. 2, E^{2}NeRF still achieves the best results on all three metrics.

Real-world data. Since the real data does not have ground truth sharp images, we conduct quantitative analysis experiments on five real scenes with no-reference image quality assessment metrics BRISQUE [28] and RankIQA [23]. As shown in Tab. 4, E^{2}NeRF also achieves best results. With blur rendering loss and event rendering loss, E^{2}NeRF is effectively strengthened by the event data. The explicit simulation of the blurring process not only achieves better deblurring performance than both the image-based deep leaning method and image-event-based method but also enables the reconstruction of a sharp NeRF from blurry input.

5.4. Ablation study

Blur loss and event loss. In Tab. 5, the results demonstrate that the proposed blur loss and event loss significantly improve the performance. In Tab. 6, we further analyze the effect of event loss on blur view and novel view. E^{2}NeRF* denotes E^{2}NeRF without event loss. On blur view, the introduction of event loss has slight improvement on the results. But on novel view, there is a more significant improvement with event loss. We believe that in the absence of event loss supervision, E^{2}NeRF* is likely to overfit to the input blurry poses and images. As shown in Fig. 3, the rendering result of the pose at t_3 may be mistaken for the pose at t_1. Therefore, the inaccurate neural 3D representation is learned and the wrong novel view color is rendered. With event data as supervision, we have the brightness change information on each pixel during the camera goes through all poses. Then the network can accurately associate each pose with the rendering result and learn an accurate neural 3D representation, thereby maintaining the performance stability on the task of generating novel views images. Fig. 4 shows the qualitative comparison between E^{2}NeRF* and E^{2}NeRF.

Ablation study on b and w. Fig. 5(a) shows that as b in-

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

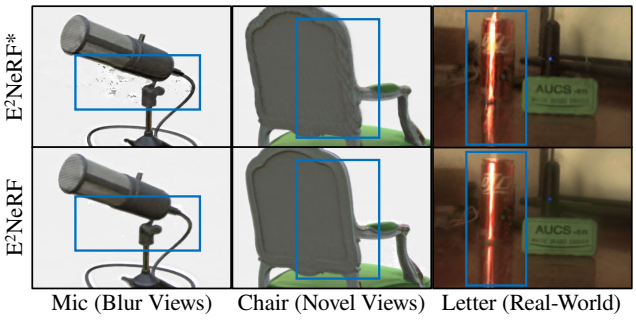

Figure 4: Qualitative analysis between E^{2}NeRF and E^{2}NeRF*. E^{2}NeRF* denotes E^{2}NeRF without event loss supervision. The results of E^{2}NeRF* tend to generate cloudy material, ripples and artifact compared to E^{2}NeRF.

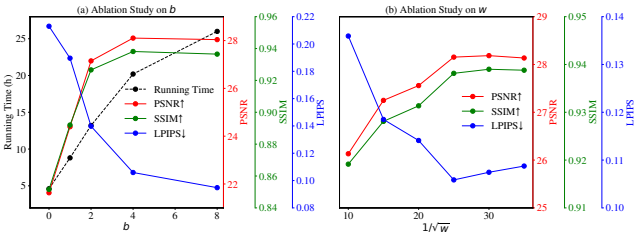

Figure 5: Ablation study on b and w on lego synthetic scene.

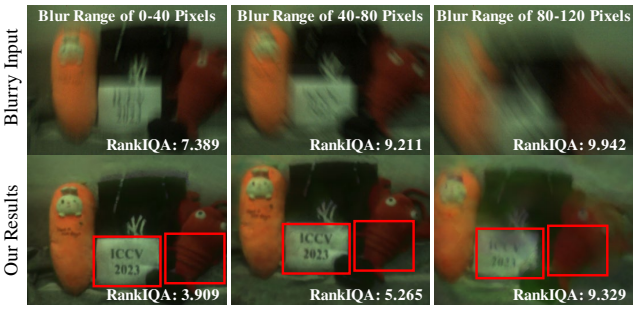

Figure 6: Results of high-speed camera motion deblurring.

creases from $b = 0$ (original NeRF), the results are gradually getting better but at the same time the training time is also increasing. And there is no significant improvement when $b >= 4$, so we choose $b = 4$ as a trade-off between training time and performance. Fig. 5(b) shows that when w gradually decreases from 0.01, the performance first improves and then decreases. Note that when w is infinitely close to 0, E^{2}NeRF will degenerate to E^{2}NeRF* and when $w = \frac{1}{625}$ we get the best results.

5.5. Qualitative analysis

Synthetic data. As shown in Fig. 8, Deblur-NeRF, D2Net-NeRF and MPR-NeRF have limited effect on deblurring. EDI-NeRF has some black artifacts on the ficus scene and

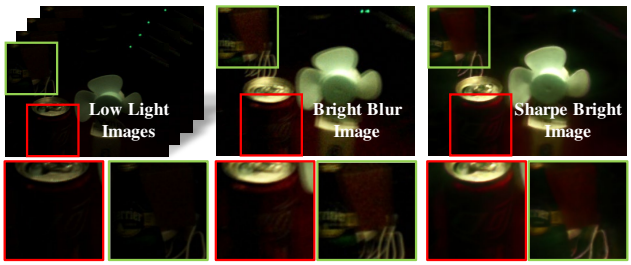

Figure 7: Results of low-light scene enhancement.

can not reconstruct the details of mustard on the hotdog scene. E^{2}NeRF has the best visual performance, which is consistent with the results of quantitative analysis.

Real-world data. In Fig 9, Deblur-NeRF and MPR-NeRF do not have any deblurring effect when the blur is very severe. With event data enhanced, D2Net-NeRF has a slightly deblurring effect. Although EDI-NeRF can achieve deblurring, it has severe chromatic aberration and noise on the results. Additionally, EDI-NeRF misses the texture details of the black belt of the camera and the grain of the lobster's back. While realizing image deblurring E^{2}NeRF maintains the texture details and color information of the scene.

A completely qualitative comparison of synthetic data and real-world data is shown in the supplement material.

5.6. Application

High-speed camera motion deblurring. Due to the challenge of accurately determining the speed of camera movement, we quantify the degree of blur caused by camera motion with pixel drift to assess the capabilities of our method. In Fig. 6, our method can effectively acquire the poses and recover scene details under a blur range of 0-80 pixels in a 346×260 resolution frame. Furthermore, our method maintains basic performance even under a blur of 120 pixels, highlighting the robustness of our model.

Low-light scene enhancement. Since our method performs well in low light, we extend it to the task of scene brightness enhancement with multiple low-light images. As shown in Fig. 7, we first use a DAVIS346 camera to capture a low-light image sequence in a low-light scene, and then synthesize it into one blurred bright image, which is then fed into the E^{2}NeRF network together with events captured by the camera. After training and rendering, a sharp and bright enhanced image can be obtained.

6. Conclusion

In this paper, we propose a novel Event-Enhanced NeRF (E^{2}NeRF), which is the first framework for learning a sharp neural 3D representation from blurry images and event data. We demonstrate the effectiveness of the proposed model on

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

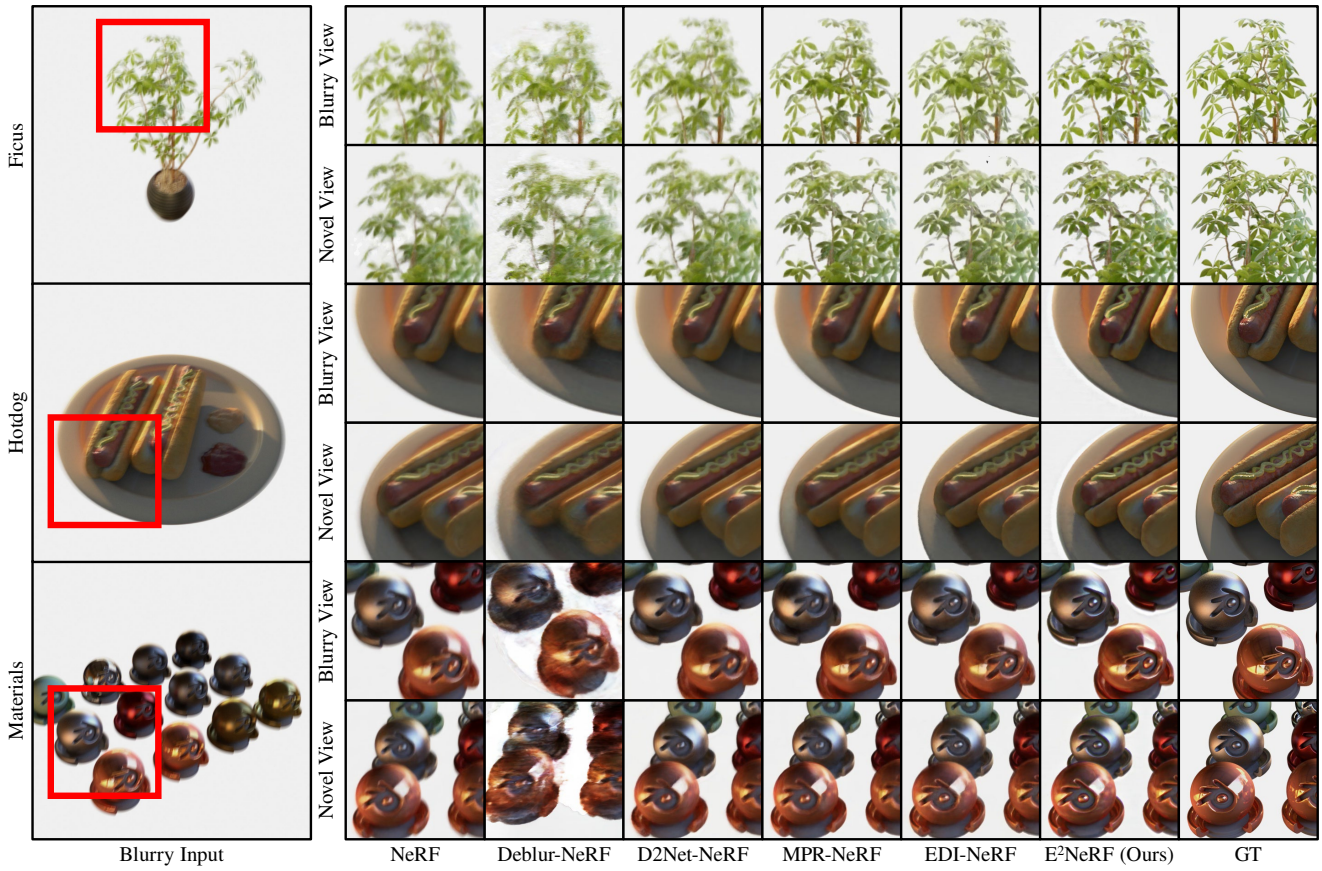

Figure 8: Qualitative comparison on synthetic data.

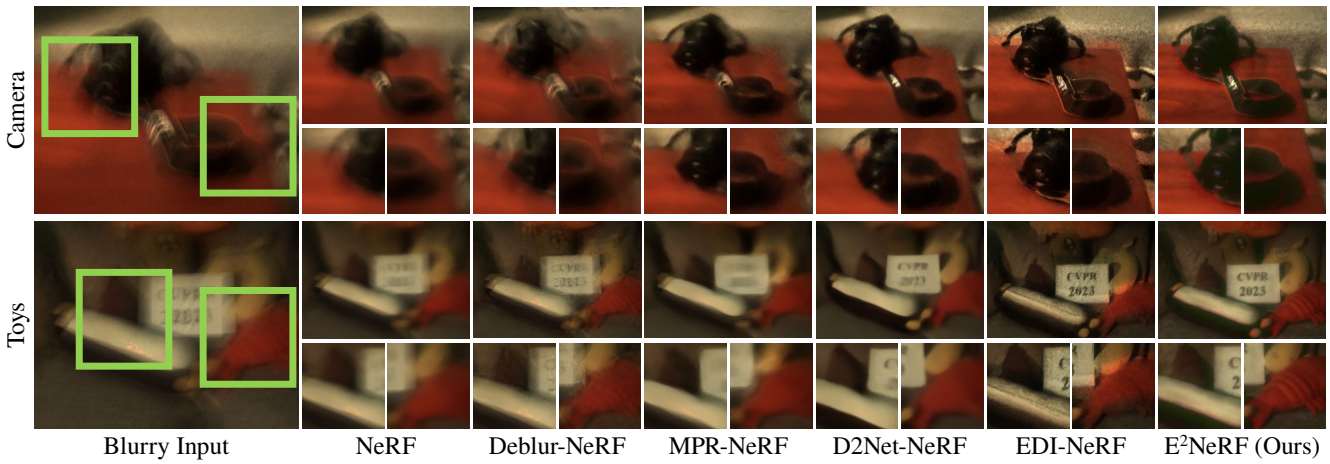

Figure 9: Qualitative comparison on real-world data.

both synthetic dataset and real-world dataset. The results indicate that our framework has significant improvement over Deblur-NeRF and image deblurring approaches. Overall, we believe that our work will shed light on the research of high-quality 3D representation learning with event-rgb data in complex and low-light scenes.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of China under Grant 62132002 and the Beijing Institute of Technology Research Fund Program for Young Scholars.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

References

[1] Soikat Hasan Ahmed, Hae Woong Jang, SM Nadim Uddin, and Yong Ju Jung. Deep event stereo leveraged by event-to-image translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 882–890, 2021.

[2] Himanshu Akolkar, Sio-Hoi Ieng, and Ryad Benosman. Real-time high speed motion prediction using fast aperture-robust event-driven visual flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):361–372, 2020.

[3] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021.

[4] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014.

[5] Tony F Chan and Chiu-Kwong Wong. Total variation blind deconvolution. *IEEE transactions on Image Processing*, 7(3):370–375, 1998.

[6] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022.

[7] Stephan J Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. Fastnerf: High-fidelity neural rendering at 200fps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14346–14355, 2021.

[8] Mathias Gehrig, Mario Millhäusler, Daniel Gehrig, and Davide Scaramuzza. E-raft: Dense optical flow from event cameras. In *2021 International Conference on 3D Vision (3DV)*, pages 197–206. IEEE, 2021.

[9] Cheng Gu, Erik Learned-Miller, Daniel Sheldon, Guillermo Gallego, and Pia Bideau. The spatio-temporal poisson point process: A simple model for the alignment of event camera data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13495–13504, 2021.

[10] Daxin Gu, Jia Li, Yu Zhang, and Yonghong Tian. How to learn a domain-adaptive event simulator. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1275–1283, 2021.

[11] Jesse Hagenaars, Federico Paredes-Vallés, and Guido De Croon. Self-supervised learning of event-based optical flow with spiking neural networks. *Advances in Neural Information Processing Systems*, 34:7167–7179, 2021.

[12] Javier Hidalgo-Carrió, Daniel Gehrig, and Davide Scaramuzza. Learning monocular dense depth from events. In *2020 International Conference on 3D Vision (3DV)*, pages 534–542. IEEE, 2020.

[13] Yuhuang Hu, Shih-Chii Liu, and Tobi Delbruck. v2e: From video frames to realistic dvs events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1312–1321, 2021.

[14] Xin Huang, Qi Zhang, Ying Feng, Hongdong Li, Xuan Wang, and Qing Wang. Hdr-nerf: High dynamic range neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18398–18408, 2022.

[15] Inwoo Hwang, Junho Kim, and Young Min Kim. Ev-nerf: Event based neural radiance field. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 837–847, 2023.

[16] Zhe Jiang, Yu Zhang, Dongqing Zou, Jimmy Ren, Jiancheng Lv, and Yebin Liu. Learning event-based motion deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3320–3329, 2020.

[17] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR 2011*, pages 233–240. IEEE, 2011.

[18] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021.

[19] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008.

[20] Songnan Lin, Jiawei Zhang, Jinshan Pan, Zhe Jiang, Dongqing Zou, Yongtian Wang, Jing Chen, and Jimmy Ren. Learning event-driven video deblurring and interpolation. In *European Conference on Computer Vision*, pages 695–710. Springer, 2020.

[21] Daqi Liu, Alvaro Parra, and Tat-Jun Chin. Globally optimal contrast maximisation for event-based motion estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6349–6358, 2020.

[22] Daqi Liu, Alvaro Parra, and Tat-Jun Chin. Spatiotemporal registration for event-based visual odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4937–4946, 2021.

[23] Xialei Liu, Joost Van De Weijer, and Andrew D Bagdanov. Rankiqa: Learning from rankings for no-reference image quality assessment. In *Proceedings of the IEEE international conference on computer vision*, pages 1040–1049, 2017.

[24] Li Ma, Xiaoyu Li, Jing Liao, Qi Zhang, Xuan Wang, Jue Wang, and Pedro V Sander. Deblur-nerf: Neural radiance fields from blurry images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12861–12870, 2022.

[25] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7210–7219, 2021.

[26] Ben Mildenhall, Peter Hedman, Ricardo Martin-Brualla, Pratul P Srinivasan, and Jonathan T Barron. Nerf in the dark:

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.

High dynamic range view synthesis from noisy raw images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16190–16199, 2022.

[27] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[28] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012.

[29] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490, 2022.

[30] Liyuan Pan, Miaomiao Liu, and Richard Hartley. Single image optical flow estimation with an event camera. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1669–1678. IEEE, 2020.

[31] Liyuan Pan, Cedric Scheerlinck, Xin Yu, Richard Hartley, Miaomiao Liu, and Yuchao Dai. 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*, pages 6820–6829, 2019.

[32] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. Esim: an open event camera simulator. In *Conference on robot learning*, pages 969–982. PMLR, 2018.

[33] Viktor Rudnev, Mohamed Elgharib, Christian Theobalt, and Vladislav Golyanik. Eventnerf: Neural radiance fields from a single colour event camera. *arXiv preprint arXiv:2206.11896*, 2022.

[34] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

[35] Wei Shang, Dongwei Ren, Dongqing Zou, Jimmy S Ren, Ping Luo, and Wangmeng Zuo. Bringing events into video deblurring with non-consecutively blurry frames. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4531–4540, 2021.

[36] Shintaro Shiba, Yoshimitsu Aoki, and Guillermo Gallego. Secrets of event-based optical flow. In *European Conference on Computer Vision*, pages 628–645. Springer, 2022.

[37] Tsuyoshi Takatani, Yuzuha Ito, Ayaka Ebisu, Yinqiang Zheng, and Takahito Aoto. Event-based bispectral photometry using temporally modulated illumination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15638–15647, 2021.

[38] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8174–8182, 2018.

[39] Gemma Taverni, Diederik Paul Moeys, Chenghan Li, Celso Cavaco, Vasyl Motsnyi, David San Segundo Bello, and Tobi Delbruck. Front and back illuminated dynamic and active pixel vision sensors comparison. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 65(5):677–681, 2018.

[40] Patrick Wieschollek, Michael Hirsch, Bernhard Scholkopf, and Hendrik Lensch. Learning blind motion deblurring. In *Proceedings of the IEEE international conference on computer vision*, pages 231–240, 2017.

[41] Fang Xu, Lei Yu, Bishan Wang, Wen Yang, Gui-Song Xia, Xu Jia, Zhendong Qiao, and Jianzhuang Liu. Motion deblurring with real events. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2583–2592, 2021.

[42] Lan Xu, Weipeng Xu, Vladislav Golyanik, Marc Habermann, Lu Fang, and Christian Theobalt. Eventcap: Monocular 3d capture of high-speed human motions using an event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4968–4978, 2020.

[43] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1107–1114, 2013.

[44] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021.

[45] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021.

[46] Jiqing Zhang, Bo Dong, Haiwei Zhang, Jianchuan Ding, Felix Heide, Baocai Yin, and Xin Yang. Spiking transformers for event-based single object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8801–8810, 2022.

[47] Jiqing Zhang, Xin Yang, Yingkai Fu, Xiaopeng Wei, Baocai Yin, and Bo Dong. Object tracking by jointly exploiting frame and event domain. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13043–13052, 2021.

[48] Kaihao Zhang, Wenhan Luo, Yiran Zhong, Lin Ma, Bjorn Stenger, Wei Liu, and Hongdong Li. Deblurring by realistic blurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2737–2746, 2020.

[49] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[50] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 989–997, 2019.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 07,2024 at 07:20:32 UTC from IEEE Xplore. Restrictions apply.