DeLiEve-Net: Deblurring Low-light Images with Light Streaks and Local Events

Chu Zhou[1] Minggui Teng[2] Jin Han[1] Chao Xu[1] Boxin Shi[2,3,4*]
[1]Key Lab of Machine Perception (MOE), Dept. of Machine Intelligence, Peking University
[2]NELVT, Dept. of Computer Science and Technology, Peking University
[3]Institute for Artificial Intelligence, Peking University
[4]Beijing Academy of Artificial Intelligence

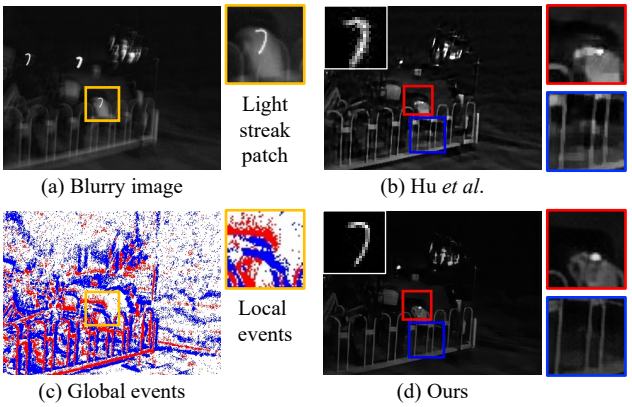
Abstract

Modern blind deblurring methods usually show degenerate performance when handling images captured in low-light conditions because these images often contain saturated regions of light sources, and the image contents and details in dark regions are poorly visible. In contrast, event cameras can faithfully record the positions and polarities of intensity changes with a very high dynamic range and low latency, which suffer less in the dark than conventional cameras. However, existing event-based deblurring methods require guidance from global events with the same spatial resolution as the blurry image (typically 346×260 *pixels), which significantly limits the spatial resolution of images they can process. In this paper, we address this problem in a two-stage way by proposing a neural network named* ***DeLiEve-Net****, which learns to Deblur low-Light images with light streaks and local Events. An RGB-DAVIS hybrid camera system is built to validate that our method can deblur high-resolution RGB images with events in low-light conditions.*

1. Introduction

Taking photos in low-light conditions requires longer exposure time and/or higher sensitivity (ISO) to ensure the sensor receives adequate light. In such a situation, the recorded pictures are very prone to blur due to inevitable camera shakes. Despite modern blind deblurring methods [51, 37] are successful in restoring plausible sharp contents in various scenarios, they usually show degenerate performance when handling images captured in low-light conditions because these images often contain saturated regions of light sources, and the image contents and details in dark regions are poorly visible (Figure 1 (a)). So it is of great interest to deblur low-light images reliably.

Unique features in low-light images, such as noise pattern [55], dark channel [35], and light streaks (caused by light sources during camera shake) [18], could facilitate their deblurring. Existing methods adopt a two-stage pipeline, *i.e.*, blur kernel estimation and deconvolution, by firstly exploring low-light features that contain useful cues about the blur kernel. However, accurately estimating the blur kernel in low-light conditions is non-trivial, since the quality of these features is severely deteriorated due to the limited dynamic range of a conventional camera. Furthermore, the iterative optimization-based deconvolution adopted by these methods [38, 9] tends to fail on images with severe saturation and high noise level, since they rely heavily on handcrafted image priors (Figure 1 (b)).



(a) Blurry image (b) Hu *et al.* (c) Global events (d) Ours

Figure 1. An example for low-light image deblurring (captured by a DAVIS346 event camera). (a) Blurry low-light image containing saturated light streaks. (b) Result of Hu *et al.* [18]. (c) Corresponding events. We use color pair (red, blue) to represent the event polarity $(1, -1)$ throughout this paper. (d) Our result. The estimated blur kernels are shown in the top left of (b) and (d).

The low latency and high dynamic range (HDR) properties of event cameras, such as DVS (captures events) [28] and DAVIS [3] (captures events along with grayscale active pixel sensor (APS) frames), make them particularly useful in deblurring applications [36, 7, 20, 29]. Thanks to the less-contaminated motion cues encoded in events (Figure 1 (c)), these methods are able to generate sharp images directly from blurry inputs, and such end-to-end de-

*Corresponding author: shiboxin@pku.edu.cn

blurring approaches demonstrate stronger robustness to artifacts arising from blur kernel estimation and deconvolution than image-based solutions. However, they require guidance from global events with the same spatial resolution as the blurry image, which significantly limits the application scenarios since the spatial resolution of events (typically 346×260 pixels) is more than 100 times smaller than an image captured by a modern camera (or camera phone).

In this paper, we propose **DeLiEve-Net**, a neural network which learns to Deblur low-Light images with light streaks and local Events. In order to process high-resolution RGB images and take full advantage of events, we design an RGB-DAVIS hybrid camera system inspired by [1]. Unlike existing event-based approaches [7, 20, 29] which are largely based on the event-based double integral (EDI) model proposed in [36], we adopt the two-stage deblurring pipeline (blur kernel estimation and deconvolution) based on spatially-uniform blur assumption and use only local events in the first stage to achieve high-fidelity blur kernel estimation (Figure 1 (d)) by utilizing the additional low latency and HDR observations encoded in events while introducing the ability to process high-resolution RGB images. The two-stage design of DeLiEve-Net is shown in Figure 2: The first stage is a blur kernel estimator analyzing the temporal and structural information of light streaks from local events in a patch to estimate the underlying blur kernel; the second stage is a non-blind image deconvolver extracting multi-scale information from the blurry image to perform noise-resistant deconvolution with the estimated blur kernel. To summarize, this paper makes contributions by demonstrating: (1) The first event-guided solution for the challenging low-light image deblurring task using light streaks. (2) A two-stage neural network for high-fidelity blur kernel estimation and noise-resistant deconvolution. (3) The first event-based deblurring method that can deblur high-resolution RGB images.

Experimental results show that DeLiEve-Net performs better than optimization-based low-light image deblurring approach using light streaks [18] and event-based deblurring methods [36, 29] requiring guidance from global events with the same spatial resolution as the blurry image (*e.g.*, captured by a DAVIS346 event camera).

2. Related works

Generally, image deblurring methods could be divided into two categories: non-blind methods, which assume the blur kernel is known, and blind methods, which deblur without knowing the blur kernel. We focus on blind methods.

Blind image deblurring. Blind image deblurring is a highly ill-posed problem due to the complexity of natural image structures and the diversity of blur kernel shapes. Some works treated this problem as a maximum *a posteriori* (MAP) estimation problem and proposed several image priors (*e.g.*, total variation regularization [5], heavy-tailed gradient distributions [12, 27], local smoothness prior [42], normalized sparsity prior [24], L_0-regularized prior [49, 34]) to relieve its ill-posedness. These handcrafted priors have shown their effectiveness in a large variety of scenes, however, they did not make full use of the information lying in specific image patterns. To improve the deblurring performance, several methods tried to exploit the latent priors in various blurring-aware indicators, such as strong edges [21, 8], patch recurrences [31], blurry image outliers [11], channel statistics [35, 50], and light streaks [18]. Recently, deep neural networks have been adopted to handle this problem. These learning-based methods could be divided into two categories: direct methods and indirect methods. Direct methods try to deblur in an end-to-end manner [32, 52, 46, 25, 26, 13, 51, 44, 53]. They usually run much faster than conventional optimization-based approaches and demonstrate visually more impressive results. However, indirect methods, which try to estimate the blur kernel [22, 37] or its attributes (*e.g.*, Fourier coefficients [4], patch-wise motion vectors [45], and dense motion flows [6, 14]) first and then use them to deblur, usually show better generalization ability and suffer less from overfitting.

Low-light image deblurring. Whereas modern deblurring methods are successful in restoring plausible sharp contents from a single blurry image, they usually show degenerate performance when handling images captured in low-light conditions due to saturation and high noise level. Several methods have been designed for such extreme conditions. Zhuo *et al*. [56] fused a pair of blurred and flash images to recover a sharp image. Zhong *et al*. [55] applied a directional low-pass filter to reduce the noise level. Pan *et al*. [35] proposed a dark channel prior which enforces the sparsity of the dark channel to facilitate blur kernel estimation. Hu *et al*. [18] utilized the light streaks as additional cues to estimate the blur kernel.

Event-based deblurring. Event cameras (*e.g.*, DVS [28] and DAVIS [3]) are bio-inspired sensors that can detect per-pixel brightness changes asynchronously. They have many attractive properties that frame-based cameras do not possess: high temporal resolution, very high dynamic range, low power consumption, and high pixel bandwidth, some of which could naturally benefit the deblurring task. Pan *et al*. [36] proposed an event-based double integral (EDI) model that clarifies the relationship among the blurry image, events, and latent frames. Chen *et al*. [7] proposed a residual model suitable for learning image deblurring and high frame rate video generation with events. Jiang *et al*. [20] used a convolutional recurrent neural network that integrates visual and temporal knowledge of both global and local scales to recover image details. Lin *et al*. [29] proposed an end-to-end trainable neural network to generate high-speed videos and used dynamic filtering to handle the

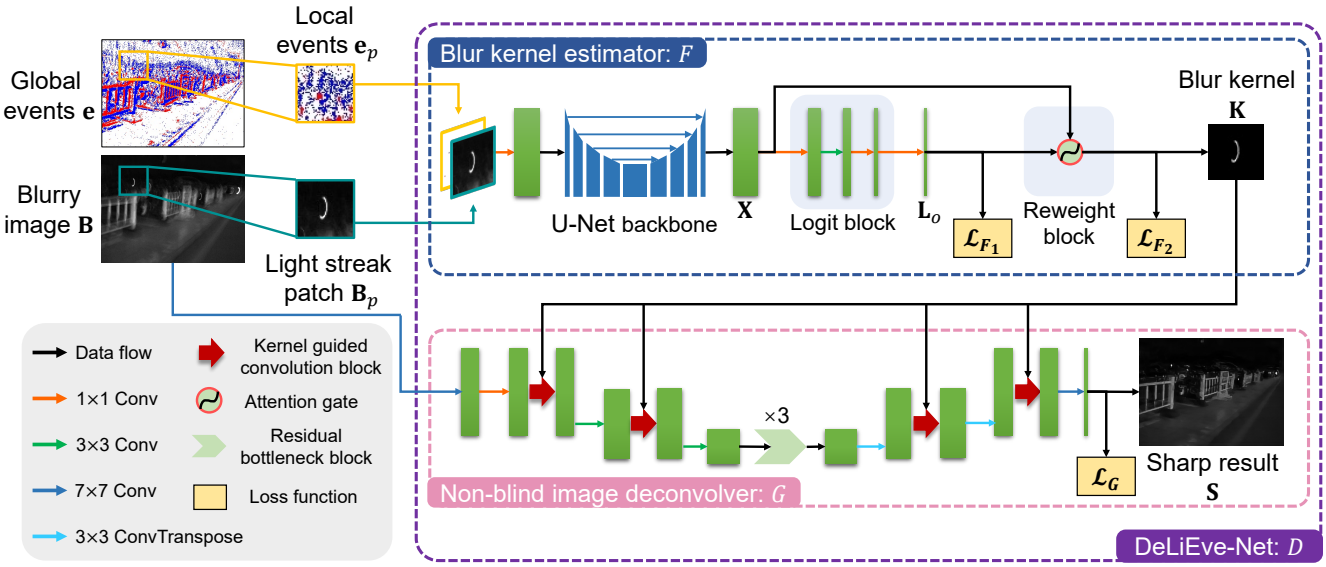


Figure 2. Architecture of the proposed deblurring network DeLiEve-Net (D). It consists of two stages: a blur kernel estimator (F) that estimates the underlying blur kernel $\mathbf{K}$ by analyzing the temporal and structural information of light streaks $\mathbf{B}_p$ from local events $\mathbf{e}_p$ in a patch, and a non-blind image deconvolver (G) that performs noise-resistant deconvolution with the estimated blur kernel $\mathbf{K}$ by extracting multi-scale information from the blurry image $\mathbf{B}$.

events triggered by the spatially-varying threshold.

Our method belongs to blind deblurring approaches designed for low-light images, and it specially utilizes light streaks in blurry images and their corresponding events to estimate the blur kernel. It suffers less from saturation and noise due to the benefit of employing events and our context-aware deconvolution process. Moreover, we expect the resolution of the images we can process is not limited to the spatial resolution of events by using only local events.

3. Proposed method

In this section, we first derive the formulation of the low-light image deblurring problem using light streaks and local events in Section 3.1. Then, we introduce our two-stage framework designs in Section 3.2 and Section 3.3. Implementation details are presented in Section 3.4.

3.1. Problem formulation

We aim to restore a sharp image from a spatially-uniform blurred image captured in low-light conditions. For the spatially-uniform blur, we could use convolution to build the image formation model:

$$\mathbf{B} = \texttt{clip}(\mathbf{I} \otimes \mathbf{K} + \mathbf{N}), \tag{1}$$

where $\mathbf{B}$, $\mathbf{I}$, and $\mathbf{K}$ denote the captured blurry image, the latent irradiance, and the spatially-uniform blur kernel respectively, $\mathbf{N}$ represents the noise term, $\otimes$ stands for the convolution operator, $\texttt{clip}$ is a clipping function defined as $\texttt{clip}(v) = v$ if v falls into the dynamic range of the camera sensor, and $\texttt{clip}(v) = 0$ or 1[1] otherwise. Since light streaks are common in low-light images and they roughly resemble the shapes of underlying blur kernels (an example is shown in Figure 1), we could explicitly utilize them as additional cues for blur kernel estimation by cropping a patch containing a light streak from the blurry image, so Equation (1) could be patch-wisely formulated as

$$\mathbf{B}_p = \texttt{clip}(\mathbf{I}_p \otimes \mathbf{K} + \mathbf{N}_p), \tag{2}$$

where the subscript p identifies variables related to the cropped light streak patch. However, due to the limited dynamic range of a conventional camera, light streaks are usually saturated so that it is difficult to estimate the blur kernel accurately. Prominently, we recognize that an event camera can faithfully record the positions and polarities of intensity changes, which suffers less from saturation than conventional cameras. Therefore, we introduce an event camera as a secondary detector to capture events during the exposure time to handle this issue.

Inside an event camera, each event $e(\mathbf{u}, t, \sigma)$ is triggered whenever the latent irradiance $L(\mathbf{u}, t)$ of pixel $\mathbf{u} = (x, y)^\top$ at time t exceeds a preset threshold c. Here σ is the polarity given by: $\sigma = 1$ if $\Delta \log L(\mathbf{u}, t) \geq c$, and $\sigma = -1$ if $\Delta \log L(\mathbf{u}, t) \leq -c$. Turning the image formation model into an event-based view [36] and denoting the latent irradiance of a light source patch (will become a light streak patch after exposure) at time t as $\mathbf{L}_p(t)$, the relationship between $\mathbf{B}_p$ and $\mathbf{L}_p(t)$ can be described as

$$\mathbf{B}_p = \texttt{clip}\left(\frac{\mathbf{L}_p(0)}{T}\int_0^T \exp\left(c\int_0^t \mathbf{E}_p(s)\mathrm{d}s\right)\mathrm{d}t\right), \tag{3}$$

where $\mathbf{L}_p(0) = \mathbf{I}_p$, T is the exposure time, and $\mathbf{E}_p(s)$ is a

[1] All irradiance values are normalized to $[0, 1]$ in this paper.

function which equals to σ if a local event $e_p(\mathbf{u}, s, \sigma)$ (triggered in the light streak patch) exists, or 0 otherwise. Then, to deblur using a two-stage pipeline, we need to 1) estimate the blur kernel $\mathbf{K}$ from the light streak patch $\mathbf{B}_p$ and its corresponding local events $\mathbf{e}_p$, and 2) conduct non-blind deconvolution on the blurry image $\mathbf{B}$ with the estimated blur kernel $\mathbf{K}$. So, from the analyses above, we could see that $\mathbf{K}$ can be determined by $\mathbf{B}_p$ and $\mathbf{e}_p$:

$$\mathbf{K} = f(\mathbf{B}_p, \mathbf{e}_p), \tag{4}$$

where f is an implicit function derived from Equation (2) and Equation (3). As $\mathbf{K}$ becomes available, we can then estimate the sharp image $\mathbf{S} = \texttt{clip}(\mathbf{I})$:

$$\mathbf{S} = g(\mathbf{B}, \mathbf{K}), \tag{5}$$

where g stands for non-blind deconvolution.

Based on Equation (4) and Equation (5), we design two network modules: blur kernel estimator F and non-blind image deconvolver G, to fit f and g respectively. The overall deblurring pipeline can be described as

$$\mathbf{S} = G(\mathbf{B}, \mathbf{K}) = G(\mathbf{B}, F(\mathbf{B}_p, \mathbf{e}_p)) \triangleq D(\mathbf{B}, \mathbf{B}_p, \mathbf{e}_p). \tag{6}$$

We call D DeLiEve-Net, whose complete pipeline is shown in Figure 2.

3.2. Blur kernel estimator

Although local events contain low latency and HDR observations about the light streak, estimating the blur kernel accurately is still difficult because events are sparse, noisy, and non-uniformly distributed signals. However, by jointly extracting features from the light streak patch and its corresponding local events, we could use the unique shape of the light streak to localize and extract the events related to the latent light source for obtaining fine-grained but less-noisy motion cues. Besides, in this way, the lost information about the unclipped radiance of the light source could be compensated by events that are not affected by the limited dynamic range of a conventional camera, which makes the blur kernel estimation more accurately.

We therefore design a blur kernel estimator F to estimate the spatially-uniform blur kernel $\mathbf{K}$ from a cropped light streak patch $\mathbf{B}_p$ and its corresponding local events $\mathbf{e}_p$. As shown in the first stage of Figure 2, this network module could be described as $\mathbf{K} = F(\mathbf{B}_p, \mathbf{e}_p)$. We first use convolutions with a large receptive field (7×7) and a U-Net [40] backbone to perform feature extraction and fusion from $\mathbf{B}_p$ and $\mathbf{e}_p$ jointly because the U-Net backbone have excellent localization and context generalization ability. Strided convolutions are used to substitute max-pooling layers in each scale of the U-Net backbone respectively for finer feature fusion. Then we try to reconstruct the blur kernel $\mathbf{K}$ from the output features $\mathbf{X}$ of the U-Net backbone. Note that we cannot reconstruct $\mathbf{K}$ directly by performing convolutions on $\mathbf{X}$ because blur kernels are always very sparse (all of the padding pixels are zero); and once a padding pixel is predicted to be a valid pixel (non-zero pixel, which belongs to the valid part of the blur kernel), the structure of the blur kernel is destroyed so that the robustness of deconvolution will be affected. To overcome this issue, we propose a "logit and reweight" strategy: We first predict logits $\mathbf{L}_o$ from $\mathbf{X}$ using a bottleneck block [16] and a 1×1 convolution ("Logit block" in Figure 2) to distinguish valid pixels from padding pixels, then predict the valid pixel values in the blur kernel $\mathbf{K}$ by reweighting the logits $\mathbf{L}_o$ using an attention gate [33] ("Reweight block" in Figure 2). This procedure could be written as $\mathbf{L}_o = \texttt{LogitBlock}(\mathbf{X})$ and $\mathbf{K} = \texttt{ReweightBlock}(\mathbf{L}_o, \mathbf{X}) = \mathbf{L}_o \odot \mathbf{M}$, where $\mathbf{M} = M(\mathbf{L}_o, \mathbf{X})$ denotes the attention map [33] computed by $\mathbf{L}_o$ (as input signal) and $\mathbf{X}$ (as gating signal) inside the attention gate, and $\odot$ stands for the point-wise multiplication operator. The "logit and reweight" strategy can enforce the sparsity of the blur kernel by predicting its logits, so more accurate blur kernel estimation can be achieved.

Note that in this stage our blur kernel estimator only operates in a local patch (identified by the subscript p), so the size of the event patch does not affect the resolution of the image to be deblurred in the next stage.

3.3. Non-blind image deconvolver

As the blur kernel $\mathbf{K}$ becomes available, we next need to perform non-blind deconvolution on the blurry input image $\mathbf{B}$ to restore the sharp image $\mathbf{S}$. However, for a low-light scene containing light sources, a conventional camera usually captures images with obvious saturation and high noise level, which may lead to severe ringing artifacts if deconvolution is directly applied. Fortunately, these images still contain useful semantic and contextual features in non-saturated regions, which should be explored to relieve the pressure of deconvolving a high-noise image.

We therefore design our non-blind image deconvolver G to extract multi-scale features for increasing the robustness. It performs deconvolution on the input blurry image $\mathbf{B}$ with the estimated blur kernel $\mathbf{K}$ for predicting the sharp image $\mathbf{S}$. As shown in the second stage of Figure 2, this network module could be described as $\mathbf{S} = G(\mathbf{B}, \mathbf{K})$. We adopt the autoencoder [17] architecture to extract image features at different levels, which is proved to be effective in recovering sharp image contents [32, 46, 51]. Furthermore, since the deconvolution is dictated by the blur kernel $\mathbf{K}$, inspired by [22], we introduce the kernel guided convolution blocks in both the encoder and decoder. Kernel guided convolution blocks map the blur kernel $\mathbf{K}$ into a list of multipliers and biases, which modulate and shift the output of the convolutions at each layer. It allows the blur kernel $\mathbf{K}$ to act on our deconvolver uniformly across the entire spatial domain,

which stabilizes the deconvolution process. Moreover, to expand the receptive field for more detailed contextual information, we embed multiple residual bottleneck blocks in the coarsest layer.

Thanks to the multi-scale contextual information extracted from the input blurry image, our deconvolver could understand the scene better and handle more complicated blurry images with saturation and high noise level, so that the deconvolution becomes more robust and the ringing artifacts are alleviated to a large extent.

3.4. Implementation details

Loss function. The loss function of the blur kernel estimator $\mathcal{L}_F$ is defined as

$$\begin{aligned}\mathcal{L}_F &= \mathcal{L}_{F_1}(\mathbf{L}_o, \texttt{bin}(\mathbf{K}_{gt})) + \mathcal{L}_{F_2}(\mathbf{K}, \mathbf{K}_{gt}) \cdot \alpha \\ &= \texttt{BCEWL}(\mathbf{L}_o, \texttt{bin}(\mathbf{K}_{gt})) + \texttt{L}_1(\mathbf{K}, \mathbf{K}_{gt}) \cdot \alpha,\end{aligned} \tag{7}$$

where $\mathcal{L}_{F_1}$ defines the loss between the logits $\mathbf{L}_o$ and the binarized ground truth blur kernel $\texttt{bin}(\mathbf{K}_{gt})$, $\mathcal{L}_{F_2}$ defines the loss between the estimated blur kernel $\mathbf{K}$ and the ground truth blur kernel $\mathbf{K}_{gt}$, $\texttt{BCEWL}$ and $\texttt{L}_1$ denote the binary cross entropy with logits loss and L_1 loss respectively, and α is set to 100. The loss function of the non-blind image deconvolver $\mathcal{L}_G$ is defined as

$$\begin{aligned}\mathcal{L}_G &= \mathcal{L}_G(\mathbf{S}, \mathbf{S}_{gt}) \\ &= \texttt{Perc}(\mathbf{S}, \mathbf{S}_{gt}) \cdot \beta_1 + \texttt{L}_2(\mathbf{S}, \mathbf{S}_{gt}) \cdot \beta_2,\end{aligned} \tag{8}$$

where $\mathbf{S}$ and $\mathbf{S}_{gt}$ represent the estimated and ground truth sharp image, $\texttt{Perc}$ and $\texttt{L}_2$ denote the perceptual loss and L_2 loss, β_1 and β_2 are set to 0.1 and 10 respectively. The perceptual loss is defined as

$$\texttt{Perc}(\mathbf{S}, \mathbf{S}_{gt}) = \texttt{L}_2(\phi_h(\mathbf{S}), \phi_h(\mathbf{S}_{gt})), \tag{9}$$

where ϕ_h denotes the feature map from h-th layer of VGG-19 network [43] pretrained on ImageNet [41], and here we use activations from $VGG_{3,3}$ convolutional layer.

Training dataset generation. Our two-stage framework designs also make generating the training dataset quite simple and flexible, whose pipeline is shown in Figure 3. Note that it is unnecessary to extract light streaks in the training phase since the training of the two stages is independent [2]. We only need to generate light streak patches directly for training the first stage and use images unrelated to those light streak patches to train the second stage.

For training the blur kernel estimator, we first generate the irradiance patch $\mathbf{I}_p$ with a fixed size of 48×48 pixels. It contains a single Gaussian light source with randomly sampled radius in the range of $[1, 5]$ pixels and intensity in the range of $[\frac{180}{255}, \frac{1000}{255}]$, laid on a random background cropped

[2]As for inference, we can either use the method proposed in [18] to automatically extract light streaks or extract them manually.

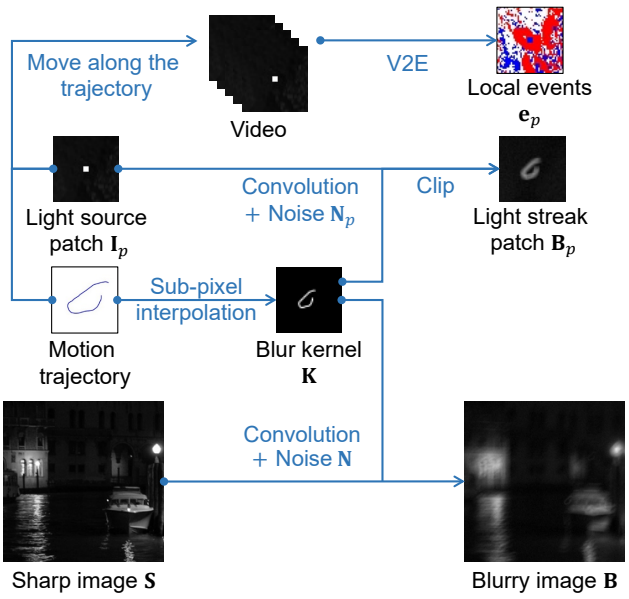


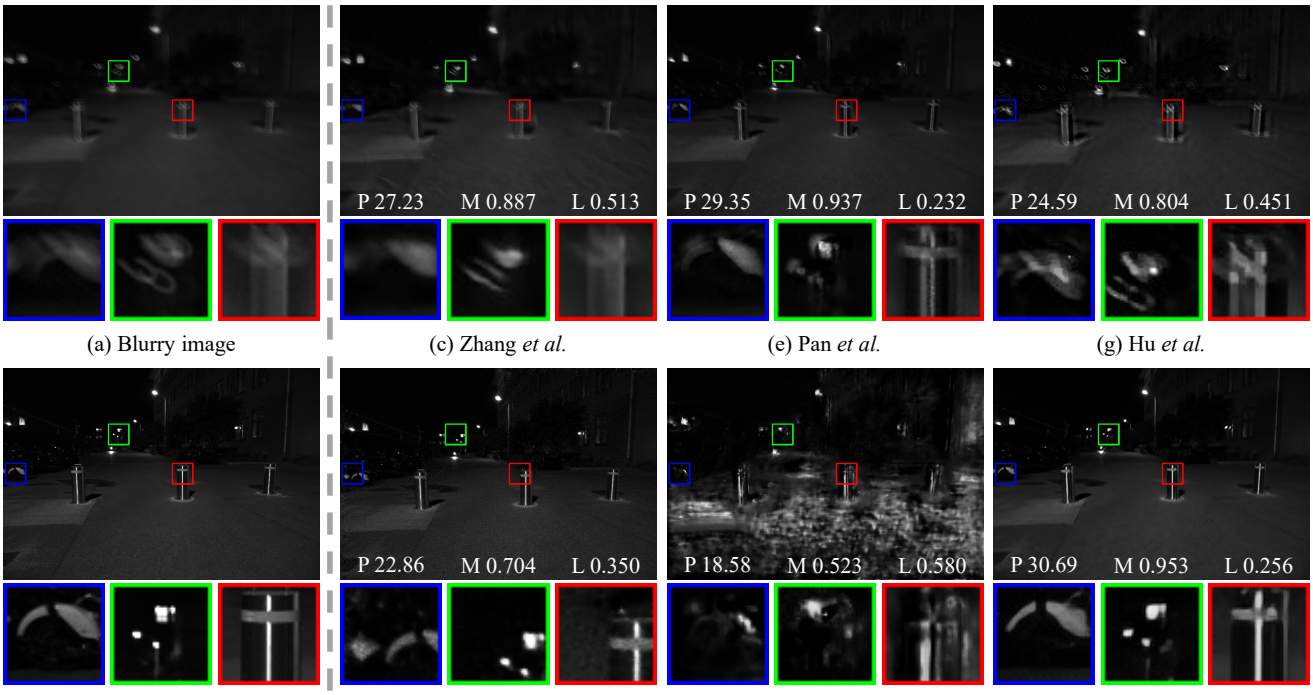
Figure 3. Training dataset generation pipeline.

from images in the GoPro dataset [32]. We then generate a random camera motion trajectory using the algorithm proposed in [2]. Sub-pixel interpolation is applied to obtain the blur kernel $\mathbf{K}$ (also 48×48 pixels), which is further convolved with $\mathbf{I}_p$ to generate the blurry light streak patch as $\mathbf{B}_p = \texttt{clip}(\mathbf{I}_p \otimes \mathbf{K} + \mathbf{N}_p)$. Here, $\mathbf{N}_p$ is additive Gaussian noise with zero mean and 1% variance. Finally, we move $\mathbf{I}_p$ along the trajectory to generate a high frame rate video at 1200 FPS and apply V2E [10] (without frame interpolation) to generate corresponding local events $\mathbf{e}_p$ (stacked into a spatio-temporal voxel grid before sending to the network).

For training the non-blind image deconvolver, we first randomly select an image being resized and cropped to 256×256 pixels from the GoPro dataset [32] and ExDARK dataset [30] as the latent irradiance $\mathbf{I}$[3]. Then we generate a blur kernel $\mathbf{K}$ and convolve it with $\mathbf{I}$ to obtain the blurry image as $\mathbf{B} = \texttt{clip}(\mathbf{I} \otimes \mathbf{K} + \mathbf{N})$ using the same method as above. Note that here the sharp image $\mathbf{S} = \texttt{clip}(\mathbf{I}) = \mathbf{I}$ because these images are captured by a conventional camera with limited dynamic range.

Training strategy. We implement DeLiEve-Net using PyTorch on a PC with an Intel Core i7-8700K CPU and an NVIDIA 2080Ti GPU, and train two stages independently for 2000 and 300 epochs respectively. ADAM optimizer [23] is used for both two stages. The learning rates for two stages are set to 1×10^{-3} and 5×10^{-4} respectively. After the first 200 epochs we linearly decay the learning rate of the second stage to 2.5×10^{-4} over the next 100 epochs. Instance normalization [47] is added during training.

[3]Strictly speaking, we should use scene radiance values here, but these images are non-linear with unknown radiometric responses. We just use the contents of them as the "ground truth" of our image irradiance.



(a) Blurry image (c) Zhang *et al.* (e) Pan *et al.* (g) Hu *et al.*

(b) Ground truth (d) Ren *et al.* (f) Lin *et al.* (h) Ours

Figure 4. Qualitative comparisons on synthetic data. (a) Blurry image. (b) Ground truth sharp image. (c)∼(h) Deblurring results of Zhang *et al*. [51], Ren *et al*. [37], Pan *et al*. [36], Lin *et al*. [29], Hu *et al*. [18], and ours. Quantitative results evaluated using PSNR (P), MS-SSIM (M), and LPIPS (L) are labeled in each image.

4. Experiments

4.1. Evaluation on synthetic data

We compare the results of DeLiEve-Net with two state-of-the-art learning-based blind deblurring methods (Zhang *et al*. [51] and Ren *et al*. [37]), two state-of-the-art event-based methods (Pan *et al*. [36] and Lin *et al*. [29]), and a low-light image deblurring method which also specially utilizes light streaks (Hu *et al*. [18][4]). Since existing benchmark dataset containing low-light images with light streaks (*e.g*., [39]) does not provide corresponding events[5], to perform the quantitative evaluation, we build a synthetic test dataset consisting of 50 different images, whose generation pipeline could be summarized as: First, we capture 10 low-light images in the RAW format including diverse scenes by a Sony α7R III camera with an FE 24-70 mm F2.8 GM lens; for each RAW image we make four additional copies so that in total we have 50 linear images whose dynamic ranges are higher than the 8-bit low dynamic range images to serve as image irradiance $\mathbf{I}$; then we use the same way as the training dataset generation pipeline in Section 3.4 to obtain 50 different camera motion trajectories and corresponding blur kernels for generating blurry images and local events.

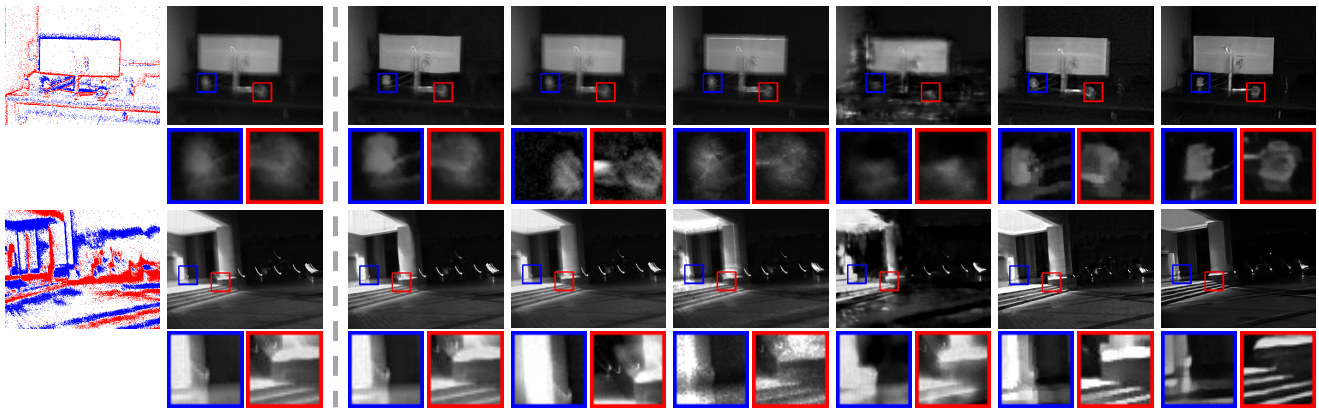
[4]For a fair comparison, we adopt their spatially-uniform deblurring method in our experiments.

[5]We cannot generate events from their data either because the motion trajectory or video is unavailable.

Table 1. Quantitative evaluation results on synthetic data among Zhang *et al*. [51], Ren *et al*. [37], Pan *et al*. [36], Lin *et al*. [29], Hu *et al*. [18], and ours. ↑ (↓) means the higher (lower) the better results throughout this paper. $*$ means that the model is retrained on our training dataset.

	PSNR↑	MS-SSIM↑	LPIPS↓
Zhang *et al*. [51]	26.201	0.9073	0.3529
Zhang *et al*. [51] $*$	25.747	0.8849	0.4014
Ren *et al*. [37]	21.270	0.7379	0.3501
Pan *et al*. [36]	26.879	0.9413	0.2074
Lin *et al*. [29]	18.881	0.6349	0.4651
Hu *et al*. [18]	24.829	0.8674	0.3075
Ours	**28.585**	**0.9621**	**0.1925**

Since the event-based methods [36, 29] we compare need global events in the whole image plane as input, we provide such information to them while only local events to our method. To evaluate the results quantitatively, we adopt three commonly adopted image quality metrics including PSNR, MS-SSIM, and LPIPS [54] (learned perceptual image patch similarity, higher (lower) means more different (similar) to ground truth, which is different from PSNR and MS-SSIM). Results are shown in Table 1. We have retrained all learning-based models with released training codes on our training dataset, and these results are marked with $*$. From the results we can see that our model outperforms the compared methods in all of the metrics, while Ren *et al*. [37] and Lin *et al*. [29] do not perform well, because Ren



(a) Events (b) Blurry image (c) Zhang *et al.* (d) Ren *et al.* (e) Pan *et al.* (f) Lin *et al.* (g) Hu *et al.* (h) Ours

Figure 5. Qualitative comparisons on real data captured by a DAVIS346 event camera. (a) Events. (b) Blurry image. (c)∼(h) Deblurring results of Zhang *et al*. [51], Ren *et al*. [37], Pan *et al*. [36], Lin *et al*. [29], Hu *et al*. [18], and ours.

Table 2. Quantitative evaluation results using kernel similarity (KS) among the methods which also estimate blur kernels (Hu *et al*. [18] and Ren *et al*. [37]) and ours.

	Hu *et al*. [18]	Ren *et al*. [37]	Ours
KS↑	0.3723	0.4947	**0.5989**

et al. [37] suffers from misalignment and ringing artifacts severely, and Lin *et al*. [29] needs information of adjacent frames, which is unavailable in our settings[6]. Visual quality comparisons[7] are shown in Figure 4, our model can deblur robustly with fewer artifacts. Furthermore, to show that DeLiEve-Net can achieve high-accuracy blur kernel estimation, we compare it to the methods which also estimate blur kernels (Hu *et al*. [18] and Ren *et al*. [37]) using kernel similarity (KS) proposed in [19] (higher means more similar to ground truth kernels), as shown in Table 2.

4.2. Evaluation on real data

To show that DeLiEve-Net has a good generalization ability on real captured low-light images and events and real camera shake, we capture several images along with corresponding events using a DAVIS346 event camera. As shown in Figure 5, our method generalizes well in both indoor and outdoor scenarios with excellent performance.

4.3. Results using different event-image resolutions

To demonstrates that DeLiEve-Net can deblur high-resolution RGB images with events, we build an RGB-DAVIS hybrid camera system consisting of an RGB camera (PointGrey Chameleon3, resolution of 2448×2048 pixels, and we resize the images to 1224×1024 pixels in our experiments) and an event camera (DAVIS346, resolution of 346×260 pixels) with the same F/1.4 lens to capture high-resolution RGB images and low-resolution events, as shown in Figure 6. To ensure the motion trajectories of the two sensors are approximately the same, we use a beam splitter in front of them to make their fields of view aligned [15, 48].

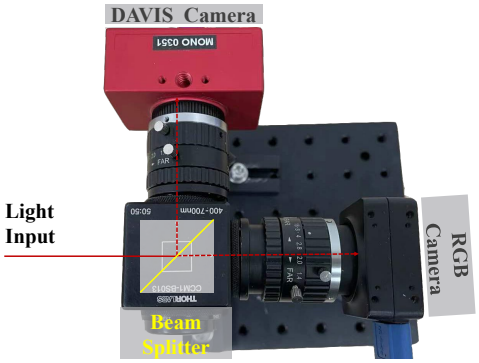
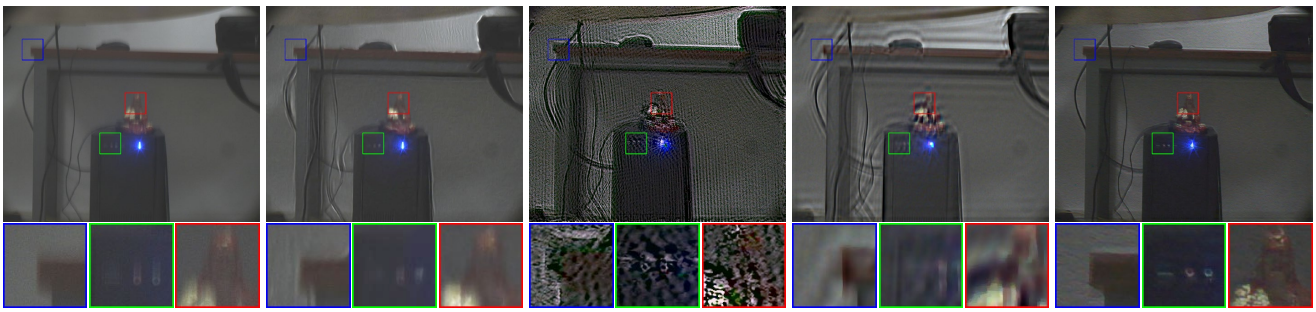


Figure 6. Our RGB-DAVIS hybrid camera system.

We need to extract events in the exposure time of a certain RGB image from the event camera using this hybrid camera system. However, it is non-trivial to achieve precisely temporal synchronization unless we can configure a synchronized clock to trigger two cameras simultaneously at the chip level, which is beyond the scope of this paper. Instead, we propose an alternative strategy to demonstrate the possibility of using DeLiEve-Net for deblurring high-resolution RGB images with local events. We assume the APS frames and events are well synchronized in the event camera. The approximated temporal synchronization is achieved by periodically capturing a scene and select the "best" aligned frame pair between RGB images and APS frames. We first set the exposure time of the RGB images to the same value as the APS frames and set the RGB camera to burst mode. Next, we capture a sequence of RGB images and APS frames, and select a frame pair with the closest appearance by scaling them to the same size and seeking a pair with maximum image similarity evaluated using MS-SSIM.

[6]Only a single frame with its corresponding events is available.

[7]Since the retrained results are not better than the original results in quantitative evaluation, we use the original results for visual quality comparisons throughout this paper.



(a) Blurry image (b) Zhang *et al.* (c) Ren *et al.* (d) Hu *et al.* (e) Ours

Figure 7. Qualitative comparisons on high-resolution real RGB data captured by our RGB-DAVIS hybrid camera system. (a) Blurry image. (b)∼(e) Deblurring results of Zhang *et al*. [51], Ren *et al*. [37], Hu *et al*. [18], and ours.

Table 3. Quantitative evaluation results of ablation study.

	PSNR↑	MS-SSIM↑	LPIPS↓
End-to-end	27.250	0.9590	0.2081
W/o events	27.112	0.9438	0.1984
W/o "l. & r."	21.203	0.7825	0.3510
W/o perc. loss	27.598	0.9582	0.2705
Our complete model	**28.585**	**0.9621**	**0.1925**

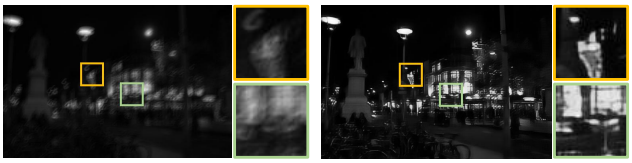
Visual quality comparisons are shown in Figure 7[8]. This proof-of-concept experiment shows a great potential of applying events for deblurring images satisfying modern camera specifications and daily life photography.

4.4. Ablation study

To verify the validity of each model design choice, we conduct ablation studies and show comparisons in Table 3. We first remove the blur kernel estimator to deblur in an end-to-end manner (End-to-end). Our two-stage model outperforms the end-to-end model, this is because estimating the blur kernel first introduces constraints in the deconvolution process, which turns the blind deblurring problem into a less ill-posed non-blind one [22, 37]. Then, we remove the local events from the input of the blur kernel estimator (W/o events) to verify the necessity of the additional low latency and HDR observations encoded in local events. Furthermore, we remove the "logit and reweight" strategy in the blur kernel estimator (W/o "l. & r."), and the performance decreases badly because the estimation of blur kernels becomes unstable. Finally, we remove the perceptual loss to show its effectiveness (W/o perc. loss). These results demonstrate our complete model achieves the optimal performance with these specific designed strategies.

5. Conclusion and discussion

We propose DeLiEve-Net, consisting of a blur kernel estimator and a non-blind image deconvolver, to deblur low-light images with light streaks and local events. It analyzes the temporal and structural information of light streaks from local events in a patch to estimate the underlying blur kernel, and extracts multi-scale information from the blurry image to perform noise-resistant deconvolution with the estimated blur kernel. Experiments show that DeLiEve-Net not only deblurs low-light images robustly with fewer artifacts but also handles high-resolution color images.



(a) Blurry image (b) Ours

Figure 8. An example of handling spatially-variant blur.

DeLiEve-Net for the first time shows a great potential of applying events for deblurring high-resolution and color images, which cannot be achieved by existing event-based methods using an APS image [36, 7, 20, 29]. However, with only local events provided, it can only handle spatially-uniform blur at present. While global events are available, handling spatially-variant blur becomes possible by adopting a similar approach as [18]: We first split the blurry image and its corresponding global events into a grid of patches to estimate the patch-wise blur kernels and use them to perform deconvolution respectively, then reweight the deconvolution results to get the final deblurred image. An example is shown in Figure 8. Although the performance of handling spatially-variant blur may not as good as spatially-uniform blur, we believe it can be improved by extending the 2D blur kernel into a 3D one, which is left for our future work. In addition, our RGB-DAVIS hybrid camera system cannot achieve temporal synchronization precisely, and this could be solved by locating the two sensors in the same chip with different resolutions (pixel sizes) in the future.

Acknowledgement

This work is supported by National Key R&D Program of China (2020AAA0105200), and National Natural Science Foundation of China under Grant No. 61872012, 61876007, 62088102.

[8]Note that we do not compare with event-based methods [36, 29] because they require guidance from global events with the same resolution as the blurry image, which is not available.

References

[1] Moshe Ben-Ezra and Shree K Nayar. Motion deblurring using hybrid imaging. In *Proc. of Computer Vision and Pattern Recognition*, 2003.

[2] Giacomo Boracchi and Alessandro Foi. Modeling the performance of image restoration from motion blur. *IEEE Transactions on Image Processing*, 21(8):3502–3517, 2012.

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

[4] Ayan Chakrabarti. A neural approach to blind motion deblurring. In *Proc. of European Conference on Computer Vision*, pages 221–235, 2016.

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

[6] Huaijin Chen, Jinwei Gu, Orazio Gallo, Ming-Yu Liu, Ashok Veeraraghavan, and Jan Kautz. Reblur2Deblur: Deblurring videos via self-supervised learning. In *Proc. of International Conference on Computational Photography*, pages 1–9, 2018.

[7] Haoyu Chen, Minggui Teng, Boxin Shi, Yizhou Wang, and Tiejun Huang. Learning to deblur and generate high frame rate video with an event camera. *arXiv preprint arXiv:2003.00847*, 2020.

[8] Sunghyun Cho and Seungyong Lee. Fast motion deblurring. In *Proc. of ACM SIGGRAPH Asia*, pages 1–8, 2009.

[9] Sunghyun Cho, Jue Wang, and Seungyong Lee. Handling outliers in non-blind image deconvolution. In *Proc. of International Conference on Computer Vision*, pages 495–502, 2011.

[10] Tobi Delbruck, Yuhuang Hu, and Zhe He. V2E: From video frames to realistic DVS event camera streams. *arXiv preprint arXiv:2006.07722*, 2020.

[11] Jiangxin Dong, Jinshan Pan, Zhixun Su, and Ming-Hsuan Yang. Blind image deblurring with outlier handling. In *Proc. of International Conference on Computer Vision*, pages 2478–2486, 2017.

[12] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T Roweis, and William T Freeman. Removing camera shake from a single photograph. In *Proc. of ACM SIGGRAPH*, pages 787–794, 2006.

[13] Hongyun Gao, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In *Proc. of Computer Vision and Pattern Recognition*, pages 3848–3856, 2019.

[14] Dong Gong, Jie Yang, Lingqiao Liu, Yanning Zhang, Ian Reid, Chunhua Shen, Anton Van Den Hengel, and Qinfeng Shi. From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur. In *Proc. of Computer Vision and Pattern Recognition*, pages 2319–2328, 2017.

[15] Jin Han, Chu Zhou, Peiqi Duan, Yehui Tang, Chang Xu, Chao Xu, Tiejun Huang, and Boxin Shi. Neuromorphic camera guided high dynamic range imaging. In *Proc. of Computer Vision and Pattern Recognition*, pages 1730–1739, 2020.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. of Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[17] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[18] Zhe Hu, Sunghyun Cho, Jue Wang, and Ming-Hsuan Yang. Deblurring low-light images with light streaks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2329–2341, 2018.

[19] Zhe Hu and Ming-Hsuan Yang. Good regions to deblur. In *Proc. of European Conference on Computer Vision*, pages 59–72, 2012.

[20] Zhe Jiang, Yu Zhang, Dongqing Zou, Jimmy Ren, Jiancheng Lv, and Yebin Liu. Learning event-based motion deblurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 3320–3329, 2020.

[21] Neel Joshi, Richard Szeliski, and David J Kriegman. PSF estimation using sharp edge prediction. In *Proc. of Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[22] Adam Kaufman and Raanan Fattal. Deblurring using analysis-synthesis networks pair. In *Proc. of Computer Vision and Pattern Recognition*, pages 5811–5820, 2020.

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

[24] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *Proc. of Computer Vision and Pattern Recognition*, pages 233–240, 2011.

[25] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. DeblurGAN: Blind motion deblurring using conditional adversarial networks. In *Proc. of Computer Vision and Pattern Recognition*, pages 8183–8192, 2018.

[26] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. DeblurGAN-v2: Deblurring (orders-of-magnitude) faster and better. In *Proc. of International Conference on Computer Vision*, pages 8878–8887, 2019.

[27] Anat Levin, Yair Weiss, Fredo Durand, and William T Freeman. Understanding and evaluating blind deconvolution algorithms. In *Proc. of Computer Vision and Pattern Recognition*, pages 1964–1971, 2009.

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

[29] 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 *Proc. of European Conference on Computer Vision*, 2020.

[30] Yuen Peng Loh and Chee Seng Chan. Getting to know low-light images with the exclusively dark dataset. *Computer Vision and Image Understanding*, 178:30–42, 2019.

[31] Tomer Michaeli and Michal Irani. Blind deblurring using internal patch recurrence. In *Proc. of European Conference on Computer Vision*, pages 783–798, 2014.

[32] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 3883–3891, 2017.

[33] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention U-Net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*, 2018.

[34] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. L_0-regularized intensity and gradient prior for deblurring text images and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(2):342–355, 2016.

[35] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *Proc. of Computer Vision and Pattern Recognition*, pages 1628–1636, 2016.

[36] 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 *Proc. of Computer Vision and Pattern Recognition*, pages 6820–6829, 2019.

[37] Dongwei Ren, Kai Zhang, Qilong Wang, Qinghua Hu, and Wangmeng Zuo. Neural blind deconvolution using deep priors. In *Proc. of Computer Vision and Pattern Recognition*, pages 3341–3350, 2020.

[38] William Hadley Richardson. Bayesian-based iterative method of image restoration. *Journal of the Optical Society of America*, 62(1):55–59, 1972.

[39] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *Proc. of European Conference on Computer Vision*, 2020.

[40] 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*, pages 234–241, 2015.

[41] 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.

[42] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH)*, 27(3):1–10, 2008.

[43] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[44] Maitreya Suin, Kuldeep Purohit, and AN Rajagopalan. Spatially-attentive patch-hierarchical network for adaptive motion deblurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 3606–3615, 2020.

[45] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *Proc. of Computer Vision and Pattern Recognition*, pages 769–777, 2015.

[46] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 8174–8182, 2018.

[47] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[48] Zihao Wang, Peiqi Duan, Oliver Cossairt, Aggelos Katsaggelos, Tiejun Huang, and Boxin Shi. Joint filtering of intensity images and neuromorphic events for high-resolution noise-robust imaging. In *Proc. of Computer Vision and Pattern Recognition*, pages 1609–1619, 2020.

[49] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural L_0 sparse representation for natural image deblurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 1107–1114, 2013.

[50] Yanyang Yan, Wenqi Ren, Yuanfang Guo, Rui Wang, and Xiaochun Cao. Image deblurring via extreme channels prior. In *Proc. of Computer Vision and Pattern Recognition*, pages 4003–4011, 2017.

[51] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. Deep stacked hierarchical multi-patch network for image deblurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 5978–5986, 2019.

[52] Jiawei Zhang, Jinshan Pan, Jimmy Ren, Yibing Song, Linchao Bao, Rynson WH Lau, and Ming-Hsuan Yang. Dynamic scene deblurring using spatially variant recurrent neural networks. In *Proc. of Computer Vision and Pattern Recognition*, pages 2521–2529, 2018.

[53] Kaihao Zhang, Wenhan Luo, Yiran Zhong, Lin Ma, Bjorn Stenger, Wei Liu, and Hongdong Li. Deblurring by realistic blurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 2737–2746, 2020.

[54] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. of Computer Vision and Pattern Recognition*, 2018.

[55] Lin Zhong, Sunghyun Cho, Dimitris Metaxas, Sylvain Paris, and Jue Wang. Handling noise in single image deblurring using directional filters. In *Proc. of Computer Vision and Pattern Recognition*, pages 612–619, 2013.

[56] Shaojie Zhuo, Dong Guo, and Terence Sim. Robust flash deblurring. In *Proc. of Computer Vision and Pattern Recognition*, pages 2440–2447, 2010.