Event-based Blur Kernel Estimation For Blind Motion Deblurring

Takuya Nakabayashi
Keio University
Yokohama, Kanagawa, Japan
nakka0204@keio.jp

Kunihiro Hasegawa
OPPO
Yokohama, Kanagawa, Japan
kunihiro.hasegawa@oppo.com

Masakazu Matsugu
OPPO
Yokohama, Kanagawa, Japan
masakazu.matsugu@srtechnologies.co.jp

Hideo Saito
Keio University
Yokohama, Kanagawa, Japan
hs@keio.jp

Abstract

Motion blur can significantly reduce the quality of images, and researchers have developed various algorithms to address this issue. One common approach to deblurring is to use deconvolution to cancel out the blur effect, but this method is limited by the difficulty of accurately estimating blur kernels from blurred images. This is because the motion causing the blur is often complex and nonlinear. In this paper, a new method for estimating blur kernels is proposed. This method uses an event camera, which captures high-temporal-resolution data on pixel luminance changes, along with a conventional camera to capture the input blurred image. By analyzing the event data stream, the proposed method estimates the 2D motion of the blurred image at short intervals during the exposure time, and integrates this information to estimate a variety of complex blur motions. With the estimated blur kernel, the input blurred image can be deblurred using deconvolution. The proposed method does not rely on machine learning and therefore can restore blurry images without depending on the quality and quantity of training data. Experimental results show that the proposed method can estimate blur kernels even for images blurred by complex camera motions, outperforming conventional methods. Overall, this paper presents a promising approach to motion deblurring that could have practical applications in a range of fields.

1. Introduction

Motion blur is caused by the camera or the subject moving during the exposure time, and it can degrade image quality. Deblurring, the process of removing blur from images, is a major topic in the field of computer vision, and many algorithms for deblurring have been proposed.

Traditionally, deblurring has been tackled by the optimization problem with constraints given by priors of the blurred images, as deblurring is an ill-posed problem that cannot be solved directly. Previous work has dealt with the problem by getting the priors on camera motion or the property of the target images. Although these methods work well only when such constraints fit to the target blurred images, they do not always go well for every blurred image in the real world.

In recent years, many deblurring methods using deep learning have been proposed. These methods learn the relationship between a blurry frame and a blur kernel (or sharp frame) from training data. Although they perform remarkably well in benchmarking, their performance is often degraded by domain differences between the target blurred images and the images used for training, which depends on the quantity and quality of training data.

To solve these problems, we propose a non-learning-based method for motion deblurring, which works by estimating a blur kernel from the events captured by an additional event camera. Event cameras are bio-inspired cameras that output the change of pixel luminance as events. Since these processes are asynchronously at each pixel position, the temporal resolution of event cameras is much higher than that of conventional frame cameras. Therefore, event cameras can capture events without suffering from motion blur.

Using additional sensors such as event cameras is a standard approach for deblurring. Deblurring images is a challenge because, as a consequence of pixel values being averaged, there is a loss of information when images get blurred. It is important to complement this information loss. Nayar and Ben-Ezra proposed a method of using an additional camera with a higher frame rate (but a lower resolution) to measure its motion during the exposure time of the target

camera [11] so that the motion blur kernel can accurately be estimated even for complex camera motion. This suggests that additional information with a high temporal resolution is effective for deblurring. The use of events as additional information is expected to be an effective approach because events provide the information with high temporal resolution as mentioned above.

In this paper, we propose a method for estimating a blur kernel using motions estimated from events. Some previous works [6, 7, 12] have been able to estimate rapid motions of cameras and objects accurately using events. We use Contrast Maximization [6] to estimate translation speeds at short intervals during the exposure time, and then we reconstruct a blur kernel. Using event data for blur kernel estimation enables us to accurately estimate complex blur kernels in the real world.

To demonstrate the effectiveness of the proposed method, experiments with synthetic and real data were conducted to confirm the effectiveness of the proposed method for deblurring. Experimental results quantitatively and qualitatively show that the proposed method can restore images more accurately than other methods.

In summary, the contributions of this paper are as follows.

- We propose a method for image deblurring by estimating a blur kernel from event data that is simultaneously captured with the image.
- We experimentally demonstrate that using the event camera with the conventional camera is effective to estimate the blur kernel for motion deblurring.
- We compare the performance of the deblurring algorithm based on the estimated blur kernel by the proposed method with related deblurring algorithms. The results prove that the proposed deblurring approach outperforms the other methods.

2. Related Work

2.1. Image Deblurring

Deblurring is one of the main topics in the field of computer vision. Blurred images are generated by averaging pixel values along the motion of the subject or camera during the exposure time. Blurred images lose some information about the latent sharp image due to this averaging. Therefore, the deblurring process is an ill-posed problem, and it is generally impossible to determine a single solution. There are two major ways to avoid this problem; one is to determine a single solution by adding constraints or prior information to the given information, and the other is to find a solution using deep learning.

The former determines a plausible solution from among countless possible solutions by adding priors to the provided image information. Priors include, for example, edge sharpness [9] and dark channel [14]. However, the problem is that these priors are not always valid in the real world, thus limiting their applicability.

With the development of deep learning, a number of the latter methods have been proposed. Chakrabarti proposed a network to predict the Fourier coefficients of blur kernels in the deblurring process [1]. Kaufman and Fattal proposed a pipeline for deblurring using an Analysis network for estimating blur kernels and a Synthesis network for non-blind deblurring [10]. More recently, an end-to-end network for deblurring has also been proposed. Cho *et al.* proposed a U-Net type network based on the concept of coarse-to-fine [3]. These methods, however, have the limitation that the recovery accuracy depends on the quantity and quality of training data.

2.2. Event-based Image Deblurring

Deblurring is one of the main applications of event cameras. Since event cameras have high temporal resolution and are capable of getting data without any blur, the use of event cameras for deblurring should be effective. L. Zhang *et al.* [20] proposed Hybrid Deblur Net that generates a dense representation from sparse events using an Encoder-Decoder network and performs deblurring using it. C. Zhou *et al.* [22] proposed DeLiEve-Net, which performs deblurring by focusing on the association between light streaks in motion blurred images and blur kernels. Also, several methods have been proposed to simultaneously perform deblurring and frame interpolation. L. Pan *et al.* [15, 16] proposed an EDI model to estimate the amount of luminance change between frames from events and a mEDI model, which is an extension of the EDI model. X. Zhang *et al.* [21] proposed a method to perform a series of processes using deep learning concerning the EDI model. Z. Jiang *et al.* [8] proposed a convolutional recurrent neural network that incorporates a directional event filtering module to extract boundary prior from events. X. Fu *et al.* [19] proposed a model for motion deblurring using real events. B. Wang *et al.* [18] proposed eSL-Net using sparse learning framework.

3. Method

Figure 1 shows an overview of the proposed method. This method takes blurred RGB frame from a RGB camera and events from an event camera attached with the RGB camera. The blur kernel is estimated from events recorded during the frame exposure. Then the RGB frame is deblurred by deconvolution with the estimated blur kernel. As described above, this method consists of two phases: estimation of blur kernels from event data and deblurring of input RGB frame. Each phase is described in detail below.

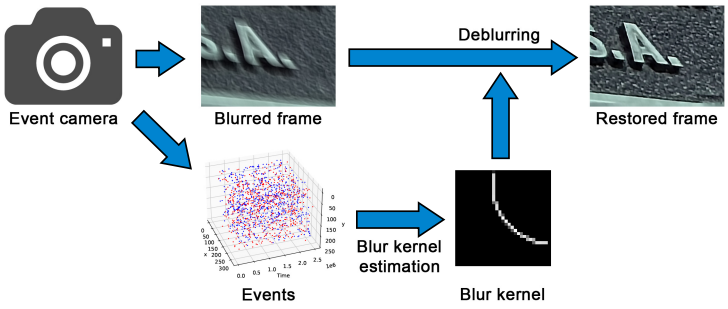

Figure 1. Overview of the proposed method.

3.1. Blur Kernel Estimation

Estimating a blur kernel accurately is crucial for deblurring. Theoretically, it is possible to remove a blur by deconvolution of a blurry image with a blur kernel. However, when there is an estimation error in the blur kernel, the blur remains or artifacts like ringing appear in the recovered image. Therefore, it can be said that the quality of deblurring depends on the accuracy of blur kernel estimation.

Since events are generated in response to changes in luminance in an image, they follow edges, which change luminance spatially. Therefore, it can be said that the events encode the motion of the camera and the subject. Since a blur kernel corresponds to the motion trajectory of them, the blur kernel can be recovered by estimating motion from the events. Since the events have detailed temporal information, blur kernels caused by severe camera motion, which are difficult to be recovered by conventional image-based methods, can be estimated by using the events.

The proposed method assumes that there is no local motion in the image, and reconstructs the blur kernel by estimating the motion trajectory of the entire image from the events. The process of blur kernel estimation is comprised of two parts: velocity estimation and blur kernel construction.

3.1.1 Contrast Maximization

The proposed method uses Contrast Maximization [6] to estimate the speed of constant velocity linear motion of an image from events. Contrast Maximization is a method to estimate parameters related to the motion of an image from events.

We explain how to estimate the velocity of a constant velocity linear motion of an image using Contrast Maximization. Consider warping all events to the reference time t_{ref} according to the velocity $\boldsymbol{v}$. Event e_k is warped according to

$$\boldsymbol{x}'_k = \boldsymbol{x}_k + (t_{ref} - t_k)\boldsymbol{v} \tag{1}$$

where $\boldsymbol{x}_k$, t_k are the position and the timestamp of e_k, and $\boldsymbol{x}'_k$ is warped position. We add the polarity $b_k = +1, -1$ of e_k to the coordinate $\boldsymbol{x}'_k$ after warp, summed value $H(\boldsymbol{x}; \boldsymbol{v})$ at the position $\boldsymbol{x}$ is calculated as

$$H(\boldsymbol{x}; \boldsymbol{v}) = \Sigma_{k=1}^{N_e} b_k \delta(\boldsymbol{x} - \boldsymbol{x}'_k) \tag{2}$$

where $\delta(\cdot)$ denotes the Dirac delta function and N_e is the number of all events. If the whole image is in translational motion with velocity $\boldsymbol{v}_{GT}$, all points in the image move on a straight line with gradient $\boldsymbol{v}_{GT}$ in xyt space. Since the events follow the edges in the image, the events are also aligned on the straight line of gradient $\boldsymbol{v}_{GT}$ in the xyt-space as shown in Figure 2. Therefore, when $\boldsymbol{v} = \boldsymbol{v}_{GT}$, since events generated from the same point are warped to the same coordinates, the value of $H(\boldsymbol{x}; \boldsymbol{v})$ should be highly biased at each coordinate, i.e., the variance should be large. In this method, the velocity is estimated by finding the value of $H(\boldsymbol{x}; \boldsymbol{v})$ that maximizes the variance of $H(\boldsymbol{x}; \boldsymbol{v})$ through optimization.

Contrast Maximization can also be used to estimate parameters of more general motions such as homography. However, as the number of parameters increases for more general motions, and dimensionality of the search space increases, optimization becomes more difficult. In the present method, the optimization is more easily completed by restricting the estimation to translational velocity.

In addition, Contrast Maximization is insensitive to noise in the events because it estimates parameters in a majority voting manner using all events. Therefore, proposed method is robust against noise in the events.

3.1.2 Velocity Estimation

Since the general motion of an image is curved and variable in speed, it is not possible to directly estimate the motion of an image by simply applying Contrast Maximization. However, the general motion can be approximated as a sequence of constant velocity linear motions within a very short period. For this purpose, we first estimate the speed of the im-

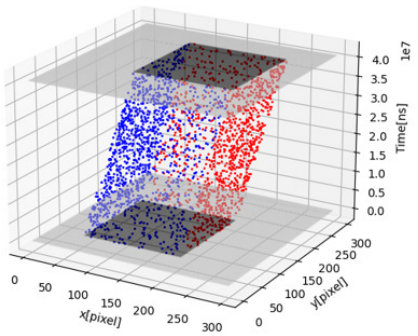

Figure 2. Event alignment. This figure visualizes events occurring when a black square moves with constant velocity in a diagonal direction. The red and blue dots correspond to the respective polarities. It can be seen that the events are aligned in xyt-space.

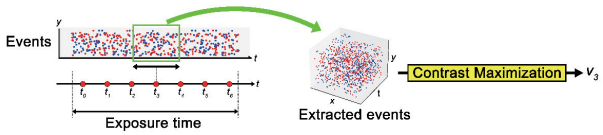

Figure 3. Velocity estimation.

age's motion at some timestamps during the exposure time using Contrast Maximization.

Figure 3 shows an overview of velocity estimation. In this part, we use Contrast Maximization to estimate the velocities of the motion during the exposure time.

First, we set N timestamps to estimate a velocity during the exposure time. The time interval D between adjacent timestamps is represented as

$$D = \frac{t_{end} - t_{begin}}{N} \tag{3}$$

where t_{begin} and t_{end} are the timestamp when the exposure time begins and ends, respectively. Also, i-th timestamp t_i $(0 \leq i < N)$ is represented as follows.

$$t_i = t_{begin} + (i + \frac{1}{2})D \tag{4}$$

Next, we estimate the velocity of an image at timestamp t_i. Using Contrast Maximization, the velocity v_i at timestamp t_i is estimated from events in the neighboring period $[t_i - \frac{T}{2}, t_i + \frac{T}{2}]$ where T is the length of the interval. If T is short enough, the motion of the image in the time interval can be regarded as a constant velocity linear motion.

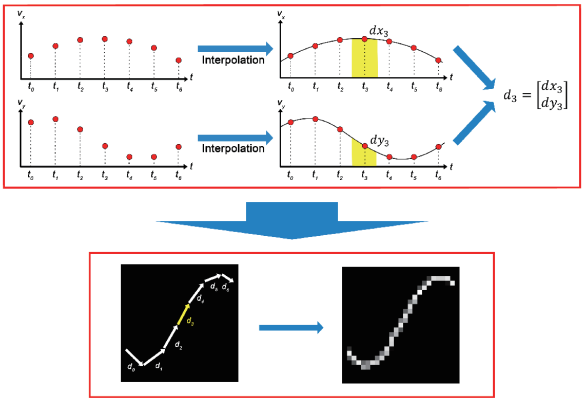

Figure 4. Blur kernel construction.

3.1.3 Blur Kernel Construction

Figure 4 shows an overview of blur kernel construction. In this part, we estimate the blur kernel corresponding to a blurry frame using the velocity estimated in Section 3.1.2.

First, we fit the velocity in the time interval around timestamp t_i as a quadratic function with respect to time t. As the velocity v_i estimated in Section 3.1.2 is a sampled value at timestamp t_i, we cannot integrate the velocity with respect to time and cannot compute the displacement. Fitting is performed in the x and y directions respectively, using three velocities v_{i-1}, v_i, and v_{i+1}. The displacement d_i in the time interval $[t_{begin} + iD,\ t_{begin} + (i+1)D]$ around time t_i is represented as

$$d_i = \int_{t_{begin}+iD}^{t_{begin}+(i+1)D} v_i(t)dt \tag{5}$$

where $v_i(t)$ is the fitting result.

Next, we construct a blur kernel by joining the estimated displacements. In this paper, since a 2D uniform blur is assumed, the motion of the entire image corresponds to the blur kernel.

3.2. Deblurring of Each Frame

After estimating blur kernels, we perform deblurring of each blurry frame input. The proposed method uses the non-blind deblurring method proposed by Cho *et al.* [2]. In the method, deblurring is performed considering the fact that there are inlier and outlier pixels in a blurred image. Inlier pixels follow the general blurred image generation process, while outliers do not follow the general generation process due to saturation. Uniform deblurring of inlier and outlier pixels may cause ringing. The method applies EM algorithm while mitigating the ringing by considering a certain percentage of outlier pixels.

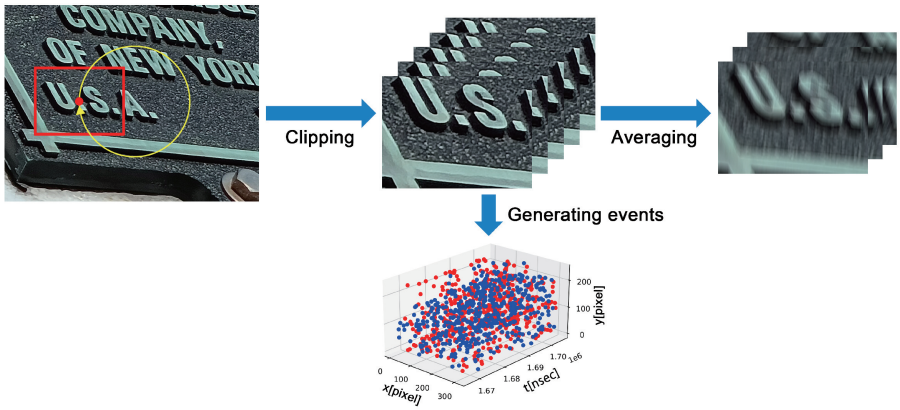

Figure 5. Overview of experimental data generation.

3.3. Frame Fusion

Due to errors in the estimated displacement using Contrast Maximization, ringing may occur in the restored image. In the proposed method, restored images are fused together to suppress the ringing. However, translational motion can cause misalignment between the restored images, making it necessary to compensate for this misalignment before fusing the images. We can calculate the amount of misalignment by summing up the displacements estimated in 3.1.2.

However, as the misalignment between the images can have errors, the Scale-Invariant Feature Transform (SIFT) features are utilized to correct them before fusing the images. Let's assume that image I is being fused onto image J. Firstly, the feature points using SIFT are extracted from both images. Next, the estimated velocity from the events is employed to calculate the rough misalignment from I to J, and the feature points extracted from I are then shifted based on the calculated misalignment. The corresponding points are searched for within a limited range around the shifted feature points, resulting in more accurate correspondence points than searching the entire image. Subsequently, the homography from I to J is estimated from the coordinates of the corresponding points, and I is transformed projectively. The final result image is obtained by projectively transforming all the restored images in this manner and calculating the average pixel value at each pixel position.

4. Experiment

4.1. Experimental Settings

4.1.1 Experimental Data

The experiments use both synthetic data generated by a simulator and real data captured by an event camera.

Figure 5 illustrates the process for generating synthetic data. Initially, we crop images of a sufficiently large source image while moving the window of size QVGA in three patterns: circle, wave, and spiral. Figure 5 shows how the center (red dot) of a QVGA-sized window (red rectangle) is moved in a circular manner (yellow arrow) while the image is cropped. Additionally, Figure 6 shows examples of the source images. Next, blurry images are produced by averaging the pixel values of several adjacent frames. Finally, we generate events by using the cropped images as input to ESIM [17], a simulator of an event camera. Although this synthetic data can be used for quantitative evaluation as the ground truth is available, it does not include noise and thus is not representative of realistic situations.

Figure 6. Examples of source images

The real data was obtained using DAVIS346 MONO. The optical center position of the camera was fixed to ensure uniform blur kernel in the obtained images. While the ground truth cannot be obtained, the real data can still be used to assess how well the proposed method performs in general situations. However, quantitative evaluation cannot be carried out using real data.

4.1.2 Comparison Method

In this paper, we compare our proposed method with four existing methods [4, 5, 13, 16]. [4, 5, 13] are image-based,

Table 1. Quantitative evaluation results in synthetic data

Pattern		Input	[5]	[4]	[13]	[16]	Ours (no Frame Fusion)	Ours
circle	PSNR	15.848	18.486	16.179	16.105	14.551	**20.849**	20.008
	SSIM	0.627	0.804	0.665	0.696	0.658	**0.897**	0.871
wave	PSNR	15.424	14.231	15.313	15.425	15.611	**23.015**	22.351
	SSIM	0.612	0.560	0.627	0.637	0.755	**0.943**	0.924
spiral	PSNR	15.898	14.907	15.699	16.515	14.978	21.065	**21.301**
	SSIM	0.614	0.615	0.618	0.699	0.696	0.890	**0.893**

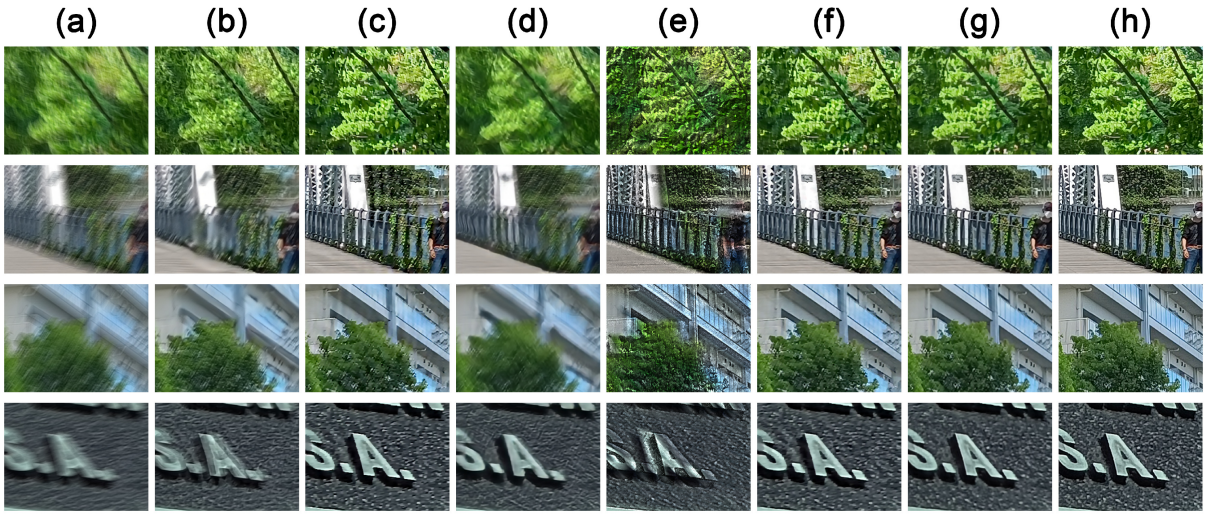

Figure 7. Qualitative evaluation results in synthetic data
(a)Input, (b) [5], (c) [4], (d) [13] (e) [16], (f)Ours(no Frame Fusion), (g)Ours, (h)Blur free

while [16] is event-based. [5] estimates blur kernels from images while accounting for outlier pixels. [4] is a U-Net type network that is one of the single-frame deblurring methods. It takes an original image and some downsampled images as input. [13] is a multi-frame deblurring method that uses information regarding optical flow between adjacent frames to restore images. [16] simultaneously performs frame interpolation and deblurring by modeling the generation process of events and blurry images and performing optimization calculations.

4.2. Result and Discussion

4.2.1 Synthetic Data

Table 1 shows the average values of PSNR and SSIM in synthetic data, and Figure 7 shows the example of the restoration images. The average values of PSNR and SSIM of the restored images are higher than those of other methods, indicating that the proposed method can perform deblurring the images with high accuracy.

While the images restored by [5] are close to being blur-free, the quantitative results do not show much improvement from the blurred images. There could be two possible reasons for this. Firstly, the estimated blur kernel may be misaligned. The image center of the blur kernel should correspond to the center time pixel of the exposure time of the motion-blurred image. As the blurred image lacks temporal information and contains only spatial information about the blurring, misalignment could occur. However, this problem does not arise in the proposed method, which estimates the blur kernel using detailed temporal information from event data. Secondly, images with biased frequency components may show low accuracy of restoration. Since [5] uses image gradient information as prior knowledge, optimization of the blur kernel may fail for such images. The results of restoration for such images, as shown in Figure 8, exhibit severe ringing.

The restored images by [4] and [13] are still blurred, and PSNR and SSIM are not much improved. These methods are based on deep learning, and their performance depends on the training data. The nature of images in the synthetic

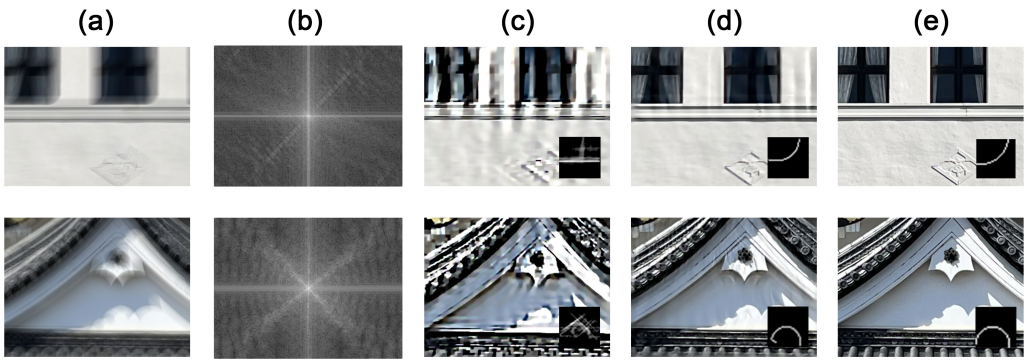

Figure 8. The results of the restoration of images with biased frequency components
(a)Input, (b)Frequency domain of (a), (c) [5], (d)Ours(no Frame Fusion), (e)Blur free

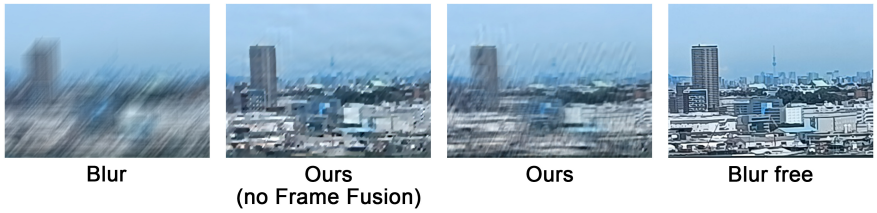

Figure 9. An example of the failure of frame fusion in our proposed method.

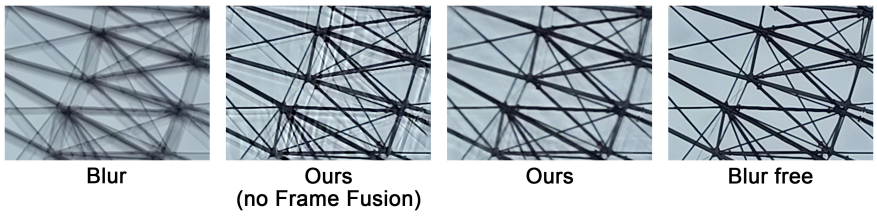

Figure 10. An example of the success of frame fusion in our proposed method.

data is very different from that of images in the training data, which may have caused the failure of the restoration. Domain adaptation is effective to avoid such a problem. Since the proposed method does not use deep learning, stable results can be obtained for any input data without domain adaptation.

In the circle and wave patterns, Ours (no Frame Fusion) exhibits higher PSNR and SSIM values than Ours. This could be due to the fact that frame fusion does not work well for frames with excessive ringing. As shown in Figure 9, when the fused image has severe ringing, feature point matching by SIFT does not work well, resulting in a poor quality image that may be worse than the original image. However, frame fusion is effective for images with minimal ringing, as shown in Figure 10, and can produce images close to the Ground Truth. Hence, it can be concluded that the effectiveness of frame fusion depends on the data being used.

4.2.2 Real Data

Figure 11 shows the qualitative evaluation results on real data. The restoration results of the proposed method demonstrate that the deblurring process is stable for all images. In particular, the third-row image contains a region of saturated pixel values, which causes all image-based methods [4, 5, 13] to fail. Nevertheless, the proposed method succeeds in restoring the image even in that region because it uses event data with a high dynamic range. Additionally, the restored images by [16] have some noisy areas. As [16] reconstructs images by accumulating information on luminance changes of event data into each frame, noise in the event data tends to accumulate in the reconstructed images.

Qualitatively, the restored image before frame fusion is better than the restored image after frame fusion. This is

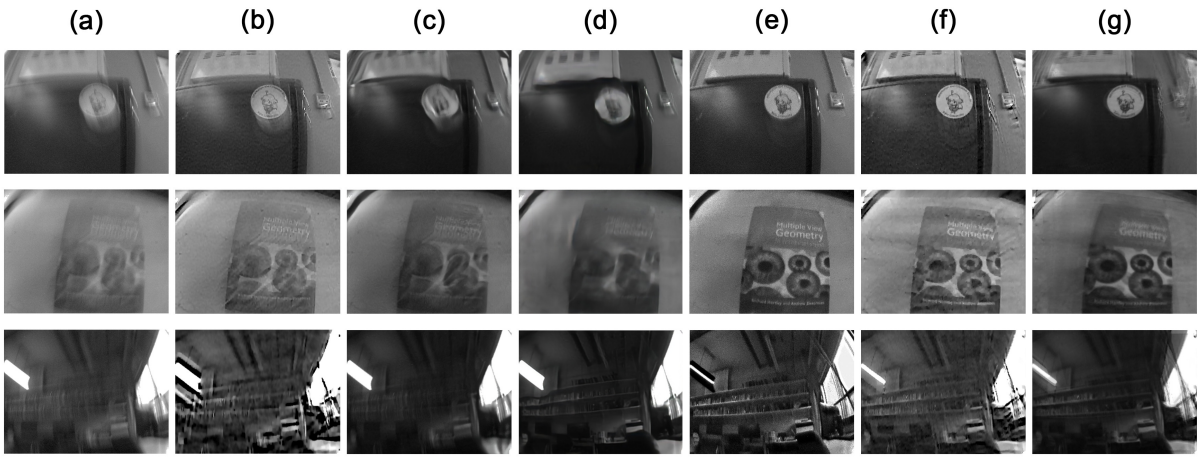

Figure 11. Qualitative evaluation results in real data
(a)Input, (b) [5], (c) [4], (d) [13], (e) [16], (f)Ours(no Frame Fusion), (g)Ours

attributed to the adverse impact of frame fusion on image quality, particularly for images with severe ringing, as previously discussed.

4.3. Limitation

While the proposed method can currently only remove 2D uniform blur from images, we anticipate that it will be possible to remove 3D rotational blur in the future. This is because Contrast Maximization can estimate not only the translational velocity of an image but also parameters related to rotation and projective transformation. The same idea presented in this paper could be applied to remove 3D rotational blur by modifying the implementation method.

Furthermore, our method is currently unable to account for local motion in images. However, we believe that this issue can be resolved by segmenting the image into parts with similar motions and applying our proposed method to each image region separately.

5. Conclusion

In this paper, we propose a novel method for estimating blur kernels from events and deblurring images. Utilizing event data with a high temporal resolution, we demonstrate that our approach can accurately estimate the blur kernel. Since our method does not rely on deep learning, the deblurring process is stable and not dependent on the quantity or quality of training data. Experimental results demonstrate the efficacy of our proposed method on both synthetic and real event data.

At present, our method is only able to support 2D uniform blur and does not account for local motion. However, we believe that these limitations can be resolved through further improvements in the implementation method.

Acknowledgement

This work was partially supported by JSPS KAKENHI Grant Number JP23H03422.

References

[1] Ayan Chakrabarti. A neural approach to blind motion deblurring. volume 9907, pages 221–235, 10 2016. 2

[2] Sunghyun Cho, Jue Wang, and Seungyong Lee. Handling outliers in non-blind image deconvolution. In *2011 International Conference on Computer Vision*, pages 495–502, 2011. 4

[3] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking Coarse-to-Fine approach in single image deblurring. Aug. 2021. 2

[4] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *2021 IEEE International Conference on Computer Vision*, pages 4641–4650, 2021. 5, 6, 7, 8

[5] Jiangxin Dong, Jinshan Pan, Zhixun Su, and Ming-Hsuan Yang. Blind image deblurring with outlier handling. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2497–2505, 2017. 5, 6, 7, 8

[6] Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. 2, 3

[7] 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 *International Conference on Computer Vision (ICCV)*, 2021. 2

[8] Zhe Jiang, Yu Zhang, Dongqing Zou, Jimmy Ren, Jiancheng Lv, and Yebin Liu. Learning event-based motion deblurring.

In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3317–3326, 2020. 2

[9] Neel Joshi, Richard Szeliski, and David J. Kriegman. Psf estimation using sharp edge prediction. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 2

[10] Adam Kaufman and Raanan Fattal. Deblurring using Analysis-Synthesis networks pair. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5810–5819, June 2020. 2

[11] Shree. K. Nayar and Moshe Ben-Ezra. Motion-based motion deblurring. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):689–698, 2004. 2

[12] Urbano Miguel Nunes and Yiannis Demiris. Entropy minimisation framework for event-based vision model estimation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 161–176, Cham, 2020. Springer International Publishing. 2

[13] Jinshan Pan, Haoran Bai, and Jinhui Tang. Cascaded deep video deblurring using temporal sharpness prior. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 5, 6, 7, 8

[14] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1628–1636, 2016. 2

[15] Liyuan Pan, Richard Hartley, Cedric Scheerlinck, Miaomiao Liu, Xin Yu, and Yuchao Dai. High frame rate video reconstruction based on an event camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[16] 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 (CVPR)*, June 2019. 2, 5, 6, 7, 8

[17] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. ESIM: an open event camera simulator. *Conf. on Robotics Learning (CoRL)*, Oct. 2018. 5

[18] Bishan Wang, Jingwei He, Lei Yu, Gui-Song Xia, and Wen Yang. Event enhanced high-quality image recovery. In *European Conference on Computer Vision*. Springer, 2020. 2

[19] 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. 2

[20] Limeng Zhang, Hongguang Zhang, Jihua Chen, and Lei Wang. Hybrid deblur net: Deep non-uniform deblurring with event camera. *IEEE Access*, 8:148075–148083, 2020. 2

[21] Xiang Zhang and Lei Yu. Unifying motion deblurring and frame interpolation with events. In *CVPR*, 2022. 2

[22] Chu Zhou, Minggui Teng, Jin Han, Chao Xu, and Boxin Shi. Delieve-net: Deblurring low-light images with light streaks and local events. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 1155–1164, October 2021. 2