Bringing Blurry Images Alive: High-Quality Image Restoration and Video Reconstruction

Liyuan Pan

A thesis submitted for the degree of
Doctor of Philosophy
The Australian National University

September 2021

Except where otherwise indicated, this thesis is my own original work.

Liyuan Pan
19 September 2021

To my family, to Junhua Pan.

Acknowledgments

Throughout the writing of this dissertation, I have received a great deal of support and assistance from my supervisors, colleagues, friends, and family.

My sincere thanks go to the members of my supervisory panel. Thanks to Richard Hartley for his support throughout the whole journey. His talented ideas are always invaluable in formulating problems and methodology. His insightful feedback pushed me to sharpen my thinking and brought my work to a higher level. He always has great patience to help me whenever I came across difficulties. I appreciate his willingness to spend many hours explaining and guiding me throughout my PhD journey. Thanks to Miaomiao Liu, she always had more faith in me than I did. She provided me with excellent support, guidance, and advice. Thanks to Yuchao Dai for his insightful comments and valuable suggestions. Thanks to Hongdong Li for his guidance in both academics and life.

I would like to acknowledge Fatih Porikli for his support and much-needed encouragement when I was a junior PhD student. Thanks also go to Robert Mahony and Davide Scaramuzza to welcome me into their group and provide insightful discussion.

I would like to acknowledge the support provided to me through an ANU PhD Scholarship and an ANU HDR Fee Remission Merit Scholarship, as well as an Australian Centre of Robotic Vision (ARCV) Top-Up Scholarship. I would like to acknowledge the facilities and technical assistance from the ANU node staff and Carol Taylor.

I would especially like to thank my friends, Jiexiu Chen and Hanxiao Jiang, for their encouragement and friendship. I would also like to thanks my colleagues at ANU, especially Cedric Scheerlinck, Xin Yu, Yiran Zhong, Hongguang Zhang, Jing Zhang, Ziwei Wang, Yonhon Ng, Zheyu Zhuang, Zhiwei Xu, and Shah Chowdhury for the collaborations.

Throughout my education journey, I am fortunate to have some inspirational teachers who encouraged me to challenge the gender stereotype and to choose a career path where girls are much underrepresented. I am proud of myself despite the various difficulties I have encountered because of my gender. In addition, I am lucky and grateful for the love and support of my family, Liu Liu. I might never have a chance to pursue my dream, and start my PhD journey, let alone have achieved so far to reach the place where I am.

Abstract

Consumer-level cameras, (e.g., phone-camera and dash-camera, *etc.*) are affordable for customers, and they are handy and easy to use. However, the images and videos are likely to appear motion blur effect, especially under low-lighting conditions. Moreover, it is rather difficult to take high frame-rate videos due to the hardware limitations of conventional RGB-sensors. Therefore, my thesis focuses on restoring high-quality (e.g., sharp and high frame-rate, *etc.*) images and videos from the low-quality (e.g., blur and low frame-rate, *etc.*) ones for better practical applications.

Recovering latent sharp images from a single image or multiple images is a fundamental task in image processing and computer vision, and various methods have been proposed. In this thesis, I first address the problem of how to restore a sharp image from a single blurred image, a blurred RGB-D image, or a blurred stereo video sequence. Then, using the faithful information about the motion provided by blurry effects in the image, I reconstruct high frame-rate and sharp videos based on an event camera, bringing blurry frames alive.

First, to tackle the challenging, minimal case of image deblurring, I focus on single-image deblurring. The image motion blur process is modelled as the convolution of a blur kernel with a latent image generally. Therefore, estimating the blur kernel is essentially important for blind image deblurring. Unlike existing approaches that focus on approaching the problem by enforcing various priors on the blur kernel and the latent image, we obtain a high-quality blur kernel directly by studying the problem in the frequency domain. It shows that the auto-correlation of the absolute *phase-only image*[1] can provide reliable information about the motion (e.g., the motion direction and magnitude, namely *motion pattern*.) that caused the blur, leading to a new and efficient blur kernel estimation approach. The blur kernel is then refined, and the sharp image is estimated by solving an optimisation problem by enforcing a regularisation on the blur kernel and the latent image. Then, the approach is extended to handle non-uniform blur, which involves spatially varying blur kernels.

Then, I focus on blur caused by camera shake. Camera shake during the exposure time is a major problem in hand-held photography. While several approaches restore a blurred image based on assumptions regarding the scene structure or the camera motion, few existing methods can handle the real 6 DoF camera motion. Therefore, we jointly estimate the 6 DoF camera motion and remove the non-uniform blur by exploiting their underlying geometric relationships, with a single blurry image and its depth map (either direct depth measurements or a learned depth map). I for-

[1]Phase-only image means the image is reconstructed only from the phase information of the blurry image.

mulate the joint problem as an energy minimisation problem, which is solved in an alternative manner. By recovering the 6 DoF camera motion and the latent image, I could also achieve the goal of generating a sharp sequence from a single blurry image.

To date, we have shown that 1) the blur kernel could be directly recovered from the *phase* information of a single blurred image and be used to deblur the image; 2) the geometry of the scene and the camera motion can be recovered from a single blurred image caused by camera shake.

In addition to the single image-based deblurring techniques, I recognise that the availability of the stereo system on smartphone devices have made significant progress recently. It helps to solve the image restoration task with stereo images. Stereo camera systems can provide motion information incorporated to remove complex spatially-varying motion blur in dynamic scenes. Given consecutive blurred stereo video frames, I aim to recover the latent images, estimate the 3D scene flow, and segment the moving objects simultaneously. These three tasks have been previously addressed by researchers separately, but they failed to exploit the internal connections among these tasks, which can potentially lead to a better solution than handling them separately. In a coupled manner, the above three tasks are naturally connected and expressed as the parameter estimation of 3D scene structure and camera motion for the dynamic scenes.

Video reconstruction is another trend in image deblurring, which reverses the blurring process by extracting a video from a single blurred image. Therefore, we introduce the event camera to this research field. Event cameras (Dynamic and Active-pixel Vision Sensor, DAVIS) are gaining attention as they can measure intensity changes in log space (called '*events*') with microsecond accuracy, even under high-speed motion and challenging lighting conditions. A blurred image can be regarded as the integral of a sequence of latent frames, while events indicate changes between latent frames. Therefore, the blur-generation process can be modelled by associating event data to a latent image. I propose a simple and effective approach, the **Event-based Double Integral (EDI)** model, to reconstruct a high frame-rate, sharp video (>1000 fps) from a single blurry frame and its event data. The video generation is based on solving a simple non-convex optimisation problem in a single scalar variable. Then, I improve the EDI model to the **multiple Event-based Double Integral (mEDI)** model by using multiple images and their events to handle the flickering effects and noise in the generated video. Besides, a more efficient solver is provided to minimise the proposed energy model.

Last, the blurred image and events also can contribute to optical flow estimation. High-speed optical flow can serve as the backbone for moving object detection, human pose estimation, and action recognition. Thus, a single image (potentially blurred) and events based optical flow estimation approach is proposed to unlock the potential applications. First, we encode the relation between flow and events effectively by presenting an event-based photometric consistency formulation. Then, we consider the special case of motion blur caused by high dynamics in the visual environments and show that including the blur formation in the model further con-

strains flow estimation. In sharp contrast to existing works that ignore blurred images, our formulation can naturally handle either blurred or sharp images to achieve accurate flow estimation. Finally, I reduce flow estimation and image deblurring to an alternative optimisation problem of an objective function using the primal-dual algorithm.

In summary, this thesis addresses the problem of sharp image restoration (with a single image, RGBD image, stereo video), as well as high frame-rate video reconstruction from both intensity images and events. Extensive experimental results demonstrate our proposed methods outperform the state-of-the-art.

Keywords: Motion Blur, Restoration, Event Camera, High Temporal Resolution Reconstruction, Primal-Dual, Fibonacci Sequence.

Contents

List of Figures

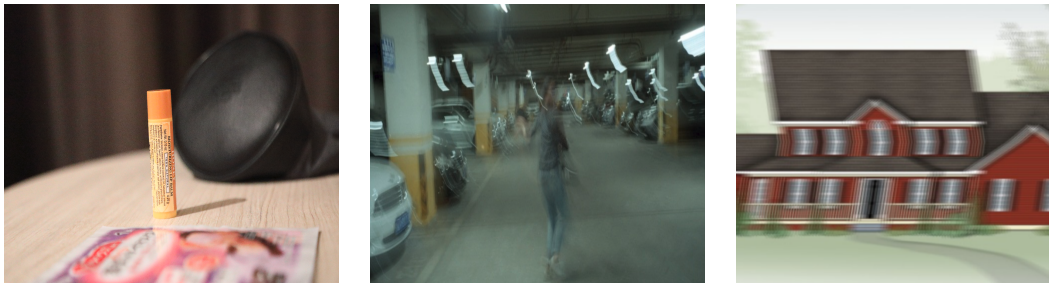
List of Tables

Introduction

In this chapter, we first introduce the problem formation on image restoration as well as video reconstruction. Then, we review related solutions to the challenging problems in the above fields. Last, we outline the organization of this thesis and the relationship between each chapter.

The format of this thesis is 'thesis by compilation' where each chapter is composed of a published paper during my PhD period.

1.1 Introduction



(a) Defocus blur (b) Motion blur (c) Rolling shutter blur

Figure 1.1: *Examples of different blurred images. (c) From Pichaikuppan et al. [2014]. (Best viewed on screen).*

Image blur is a widely encountered issue in real-world applications, and it can be caused by various reasons, such as optical aberration, medium perturbation, rolling shutter effect, and relative motion between camera and scene (seeing Fig 1.1 for more details). The unexpected blur will decrease the image quality by losing essential information, which will hamper further analysis and applications, such as optical flow estimation Gong et al. [2017b]; Pan et al. [2017b], depth estimation Hu et al. [2014]; Pan et al. [2021], matching Liu et al. [2020, 2021], and objects segmentation Pan et al. [2016a]; Pan et al. [2020]. In this thesis, I focus on images with motion blur.

Motion blur is caused by the way a camera takes pictures and its aperture time. The longer the aperture is open, which is the exposure time, or the faster the motion,

the more possible the blurrier moving objects might appear. In other words, motion blur is the result of the relative motion between the camera and the scene, together with the exposure time. The relative motion is further divided into camera shake, object motion, depth variations, or a combination of them. To reduce the degree of blur, one can capture images using shorter exposure intervals. This, however, increases the amount of noise in the image, especially under low-lighting conditions. Another approach is to recover the unknown image from its blurred version, namely image deblurring, restoration, or deconvolution.

The blurring process can be modeled as a convolution of a latent image with a blur kernel (also known as point-spread-function, PSF), and is mathematically given by

$$\mathbf{B} = \mathbf{k} \otimes \mathbf{L} \,, \tag{1.1}$$

where $\mathbf{B} \in \mathbb{R}^{h \times w}$ denotes the blurred image, and $\mathbf{L} \in \mathbb{R}^{h \times w}$ is the latent sharp image. Here, $\otimes$ is the convolution operator, h and w are the image height and width, and $\mathbf{k}$ is the 2D blur kernel. With a known or unknown blur kernel, traditional image deblurring methods are broadly categorised into two cases: non-blind deblurring methods and blind deblurring methods.

1.1.1 Non-blind deblurring

Non-blind deblurring methods attempt to restore the latent image with a given blur kernel faithfully.

With a known blur kernel $\mathbf{k}$, computations are performed in the frequency domain for ease, as the convolution theorem states that Fourier transform of a convolution is the element-wise multiplication. In the Fourier domain, Eq. ((1.1)) corresponds to

$$\mathcal{F}(\mathbf{B}) = \mathcal{F}(\mathbf{k}) \odot \mathcal{F}(\mathbf{L}) \,, \tag{1.2}$$

where $\odot$ represents the component-wise multiplication, $\mathcal{F}()$ denotes the Fourier transform. Then an estimate of $\mathcal{F}(\mathbf{L})$ can be directly obtained by an inverse filter,

$$\mathcal{F}(\mathbf{L}) = \mathcal{F}(\mathbf{B}) / \mathcal{F}(\mathbf{k}) \,. \tag{1.3}$$

This process is called direct inverse filtering, and it works if $\mathcal{F}(\mathbf{k})$ is invertible. However, the sparse matrix $\mathcal{F}(\mathbf{k})$ usually has a large condition number, which is nearly an ill-posed, singular matrix. It indicates the direct inverse filtering solution is sensitive to perturbation and rounding errors.

Early non-blind deblurring methods using the simple Wiener deconvolution Wiener [1950] to estimate the pseudo-inverse filter in the frequency domain, and is expressed as

$$W(\mathbf{k}) = \frac{\mathcal{F}^*(\mathbf{k})}{|\mathcal{F}(\mathbf{k})|^2 + 1/SNR} \,, \tag{1.4}$$

where the superscript $*$ denotes complex conjugation, and SNR denotes signal-to-noise ratio. Then, the corresponding latent image is estimated by taking inverse Fourier transform $\mathcal{F}^{-1}(\mathcal{F}(\mathbf{B})W(\mathbf{k}))$. Albeit efficient and straightforward, in real

(a) Blurred image (b) Direct inverse filtering

(c) Wiener deconvolution (d) With a regularization term

Figure 1.2: *Example result of (non-uniform blurred) image deblurring. The blurred image and its ground-truth blur kernel are from Levin et al. [2009]. Even if the kernel is known precisely, ringing artefacts appear in (b) and (c). The ringing effect is due to amplified singularities by the presence of an unknown noise. (a) Blurred image. (b) The image is deblurred by using a direct inverse filter and therefore causes ringing artefacts. (c) Ringing artefacts caused by the Wiener deconvolution. (d) A restored image with a penalty term by Pan et al. [2019b]. (Best viewed on screen).*

blurred images, an unknown noise n is added in the above model and makes the direct inverse filtering solution and Wiener deconvolution failed (seeing Fig 1.2 for more details).

In general, the non-blind deblurring problem may be formulated as finding

$$\text{argmin}_{\mathbf{L}} \|\mathbf{k} \otimes \mathbf{L} - \mathbf{B}\|_2^2 \ . \tag{1.5}$$

However, in most cases, blurring acts as a form of a low-pass filter – losing high-frequency information. The deblurring process is to restore the lost frequency components of the image. Thinking of convolution with a known $\mathbf{k}$ as a linear operator, the existing near-zero eigenvalues whose eigenvectors correspond to high-frequency components of the signal (image). If high-frequency components are over-emphasized in the deblurring process, the resulting latent image $\mathbf{L}$ will be noisy, or edges will show ringing. Consequently, this problem is not well-conditioned. A standard solution to this is to add regularisation terms that discourage excessive high-

frequency components. They are used to introduce prior knowledge and make the approximation of ill-posed (pseudo-)inverses feasible. Therefore, led to the following minimization problem.

$$\min_{\mathbf{L}} \|\mathbf{k} \otimes \mathbf{L} - \mathbf{B}\|_2^2 + \alpha \phi_{\text{reg}}(\mathbf{L}) \ ,$$

where α is a weighted parameter and $\phi_{\text{reg}}(\cdot)$ is a regularization term (e.g., l_0 regularization Xu et al. [2013] and low rank regularization Candès and Recht [2009], *etc.*). Various regularization terms are used to discourage excessive noise and over-emphasized edges and guide latent image estimation. Such as the total variation regularization Rudin et al. [1992] and Tikhonov regularization (of the magnitude of the image gradient $\|\nabla \mathbf{L}\|^2$).

In most situations, however, the blurring kernel is unknown. Therefore, the deblurring task requires the estimation of the underlying blurring kernel, namely blind deblurring.

1.1.2 Blind deblurring

Blind deblurring approaches aim at restoring the latent sharp image and the underlying kernel from blurred images simultaneously. In this thesis, we focus on blind motion deblurring.

Blind deblurring is an ill-posed problem Levin et al. [2011], as there are infinitely many pairs of blur kernels and images that could generate the same blurry image. For example, one undesirable solution that perfectly satisfies Eq. (1.1) is the no-deblur explanation: $\mathbf{L} = \mathbf{B}$ and $\mathbf{k}$ is a delta (identity) function. Various kinds of priors/constraints have been proposed either on the blur kernel or the latent image to regularise the solution space.

As we mentioned in Eq. (1.1), a motion blur is characterised by its blur kernel (PSF), whose parameters are closely related to the motion. Hence, assumptions for camera motions and the number of moving objects have been made by researchers and significantly contribute to the estimation of the exact kernel needed for deconvolution. The blur kernel can be further categorised into two cases: the spatially-invariant blur kernel and the spatially-variant blur kernel.

1.1.2.1 Spatially-invariant blur kernel

Several approaches assume that the captured image has a spatially-invariant kernel (also known as the uniform blur kernel), which means the blur kernel is the same for all pixels in an image. Under this condition, the blur is usually caused by a camera shake (in-plane shifting, no rotation) with a static scene, and the scene must have a constant depth Dai and Wu [2008]; Hirsch et al. [2011]; Pan et al. [2016b] (seeing Fig 1.3 for more details).

Camera shake can be modelled as a blur kernel, describing the camera motion during exposure. In the basic formulation Eq. (1.1), the problem is under-constrained: there are more unknowns (the latent image and the blur kernel) than

(a) (b)

Figure 1.3: *Examples of spatially-invariant blurred image and its blur kernel. (b) From Köhler et al. [2012]. (Best viewed on screen).*

measurements (the observed image). Hence, to resolve the ill-posed underlining optimisation problem, practical solutions must make strong prior assumptions about the blur kernel, about the image to be recovered, or both. One straightforward approach is to apply the maximum a posteriori (MAP) solution. Besides, the variational Bayes approximation considers uncertainties in the unknowns, allowing us to find the blur kernel implied by a distribution of probable images Fergus et al. [2006]. Numerous methods estimate the blur kernel and the latent image in an alternating optimisation scheme. An alternative approach is to estimate the actual motion of the camera. With the motion direction, magnitude, or other motion assumptions, the exact kernel can be derived. Once knowing the blur kernel, the problem becomes a non-blind deconvolution problem Gupta et al. [2010].

However, for a real-world image, when the scene contains several objects moving independently, camera rotations, or forward out-of-plane translations, the assumption of a single uniform motion blur kernel is not always held.

1.1.2.2 Spatially-variant blur kernel

Non-uniform deblurring has attracted much attention in recent years. This section of the chapter discusses existing deblurring methods with different motion assumptions (seeing Fig 1.4 for more details).

For a scenario contains camera rotations or forward out-of-plane translations, several single image-based approaches can be extended to handle this non-uniform deblurring directly based on the geometric model of camera motion Whyte et al. [2012]. Another approach is to estimate motion blur kernels using the local patches Sun et al. [2015]. As a result, the only unknown parameters for the blur kernel are its direction and width within each local patch. However, the above approaches do not apply to non-static scenes or scenes with varying depth.

For a static camera that recording scenes with multiple moving objects, the blur cannot be described by a single uniform kernel or a single geometric motion model. Researchers tend to describe the motion blur at the patch-level. In other words,

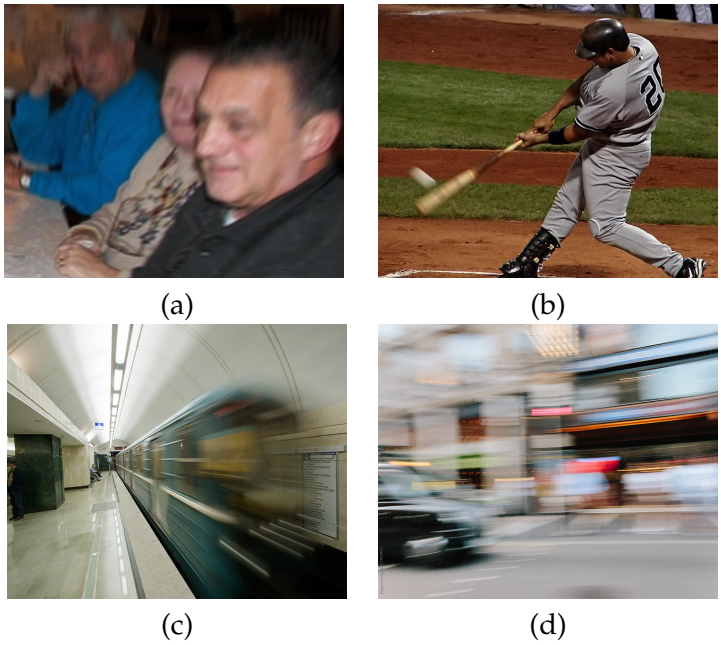


Figure 1.4: *(a) An example of a spatially-variant blurred image captured by a rotating camera. (b) An example of a spatially-variant blurred image with multiple moving objects. (c) An example of a spatially-variant blurred image with a varying depth. (d) An example of a spatially-variant blurred image with both a moving camera and a moving object. From Sun et al. [2015] and Shi et al. [2014]. (Best viewed on screen).*

several researchers go one step further by assuming that the image consists of a small (fixed) number of blurring layers/patches with the same blurring kernel within each layer/patch. An example of a real-world image, with multiple moving objects and a static background, is shown in Fig 1.4 (b). The baseball and baseball bat are blurred with different blur kernel independently, and the image can be separated into two or more layers/patches rely on the statistics-based Levin [2007], depth-based Hu et al. [2014], or edge-based priors Pan et al. [2014].

For a scenario where camera motion, multiple moving objects, or depth variations exist, the blur kernel is, in principle, defined by each pixel in the blur region (as shown in Fig 1.4 (c) and (d)). Single image-based methods, such as Cho and Lee [2009]; Gupta et al. [2010]; Michaeli and Irani [2014]; Xu et al. [2013], cannot be directly applied since they are restricted to a single or a fixed number of blur kernels, making them inferior in tackling general motion blur problems. In this case, several researchers assume that the depth or optical flow is known, and therefore benefits the single-image-based non-blind motion deblurring problem by estimating the spatially-variant kernel Pan et al. [2019a]; Gong et al. [2017b]. In addition, several approaches attempt to leverage multiple images (e.g., a monocular video and multi-view videos) to restoring the sharp frames.

For multi-image-based deblurring approaches, estimating scene flow via the video

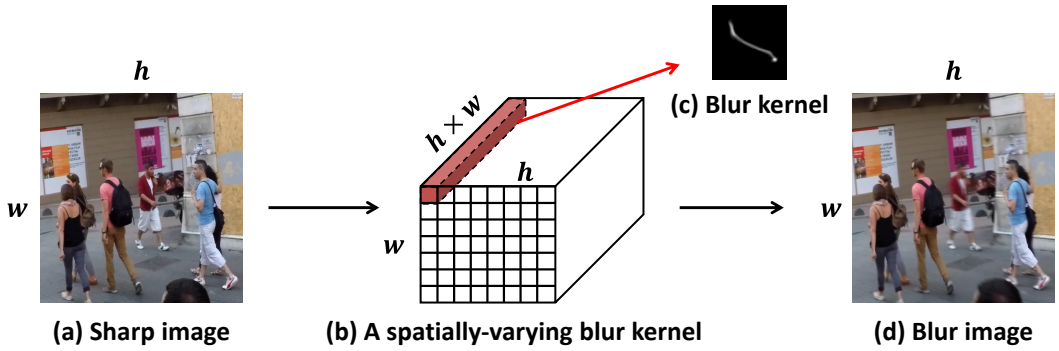


Figure 1.5: *A spatially-varying blurred image (from Nah et al. [2017]) and its blur kernel* $\mathbf{K}$. *Given a input sharp image (a) and a spatially-varying blur kernel (b), a non-uniform blurred image (d) can be generated by Eq.* (1.6). *A vector along the third dimension of (b) corresponding to a blur kernel (c) at pixel* $\mathbf{x}$, *where most values in the vector are zero. For better visualization, we resize the vector (*$\mathbf{k}_\mathbf{x}$*) to a matrix. Each pixel in (a) associate with a blur kernel. (Best viewed in colour on screen).*

helps to restore latent images. Take optical flow as an example, the phenomenon around flow and blur can be viewed as a chicken-egg problem. More effective motion blur removal requires more accurate motion estimation. Yet, the accuracy of motion estimation highly depends on the quality of the images. Recognising that these two problems are intertwined, I suggest developing a method to tackle both issues at once. Several methods Kim and Lee [2015]; Gong et al. [2017b] leverage a pixel-wise blur kernel based on the optical flow,

$$\mathbf{B}(\mathbf{x}) = \text{vec}(\mathbf{k}_\mathbf{x})^\text{T}\text{vec}(\mathbf{L}) \,, \tag{1.6}$$

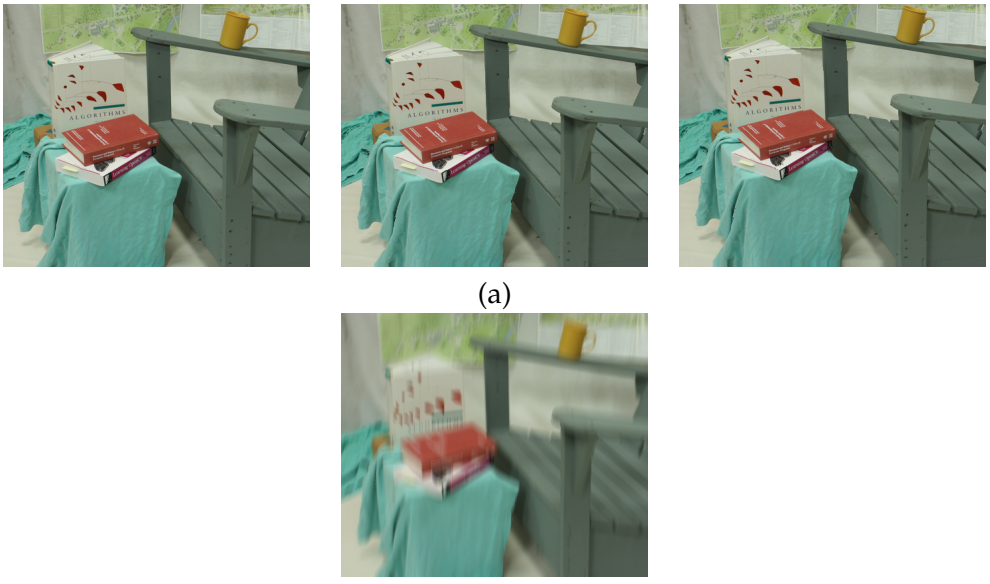
where $\text{vec}(\cdot)$ is the vectorize symbol, and $\mathbf{B}(\mathbf{x})$ is the intensity of pixel $\mathbf{x}$. Let $\mathbf{k}_\mathbf{x} \in \mathbb{R}^{h\times w}$ denotes the pixel-wise blur kernel at $\mathbf{x}$, it can be approximated by using bidirectional optical flows Kim and Lee [2015]; Pan et al. [2017b]. The blur kernel $\mathbf{K} \in \mathbb{R}^{h\times w\times(h\times w)}$ can be obtained by stacking $\mathbf{k}_\mathbf{x}$ (seeing Fig. 1.5). The assumptions here are 1) the motion is in a consistent velocity; 2) the motion is in a single direction.

Recently, researchers have studied to handle the blind motion deblurring with the powerful deep neural network. Most existing deep learning-based methods Tao et al. [2018]; Nah et al. [2017] put the image in different net architectures and use $\|\mathbf{L} - \bar{\mathbf{L}}\|$ as their loss function. Here, $\mathbf{L}$ denotes the ground-truth latent image, and $\bar{\mathbf{L}}$ is the deblurring result. This kind of solution achieves superior results. However, these blindly learning-based approaches not only require large scale training datasets but also lack interpretability. They implicitly extract the distribution of images directly from their training data without taking the physical image formation process into account. A few works Vasu et al. [2018]; Eboli et al. [2020] attempt to introduce blur kernel estimation to a deep neural network. However, they generally assumed the blur kernel to be spatially-invariant. This assumption reduces the generality of a deep neural network significantly, as we can hardly meet the assumption when rel-

ative motion between the camera and the objects occurred, especially in real-world applications. To improve their generalization ability while without losing the interpretability, several video-based methods introduced a self-supervised deblurring strategy Chen et al. [2018] by reblurring the output (deblurred image) with the optical flow between subsequent frames to match the input blurred image. However, it is still hard to embed the physical image formation process into a deep neural network.

With the above discussions, to tackle image restoration, in this thesis, we first propose a single image-based deblurring method by studying the problem in the frequency domain with the *phase-only image*. We further extend our approach to handle non-uniform blur, which involves spatially varying blur kernels. Then, with an RGBD image, we propose to jointly estimate the 6 DoF (degrees-of-freedom) camera motion and remove the non-uniform blur caused by camera motion by exploiting their underlying geometric relationships, with a single blurry image and its depth map (either direct depth measurements or a learned depth map) as input. Moreover, given consecutive blurred stereo video frames, we propose a method to recover latent clean images, estimate the 3D scene flow, and segment the multiple moving objects simultaneously.

1.1.3 Video reconstruction



(a)

(b)

Figure 1.6: *An example of generating a blurred image. (a) Samples of sharp frames for generating a blurred image. (b) The blurred image is generated based on the Middlebury dataset Scharstein et al. [2014]. (Best viewed on screen).*

Video reconstruction is another deblurring trend that reverses the blurring process by extracting a video from a single blurred image. The reconstructed video is

used to understand the dynamics of a scene better. Referring to the digital camera operation, the blurred image is modelled by the integration of light intensity emitted from the dynamic scene over the aperture time interval of the camera and is given by

$$\mathbf{B} = \frac{1}{T}\int_{f-T/2}^{f+T/2} \mathbf{L}(t)\, dt, \tag{1.7}$$

where $[f - T/2, f + T/2]$ is the exposure interval, and f is the reference timestamp (seeing Fig. 1.6 for details).

Learning-based approaches, such as Jin et al. [2018], use video to train multiple neural networks to estimate the underlying frames. They adopt Eq. (1.7) in discrete space, which is $\|\mathbf{B} - \frac{1}{N}\sum_0^N \bar{\mathbf{L}}n\|$, as their loss function, where $n \in [0, N]$ is the sample index and N is the sample number. However, the reconstructed videos usually do not obey the 3D geometry of the scene and camera motion.

Therefore, several researchers Pan et al. [2019c, 2020a] attempt to introduce the event camera into the deblurring field and reconstruct a high frame-rate sharp video using a single image that maintains temporal consistency.

Event cameras (such as the Dynamic Vision Sensor (DVS) Lichtsteiner et al. [2008] and the Dynamic and Active-pixel Vision Sensor (DAVIS) Brandli et al. [2014a]) measure intensity changes (called 'event') asynchronously at each pixel with microsecond temporal resolution. Unlike conventional cameras that produce the full image at a fixed frame-rate, event cameras trigger events whenever the change in intensity at a given pixel exceeds a preset threshold. Event cameras do not suffer from the limited dynamic ranges typical of sensors with synchronous exposure time, and are able to capture high-speed motion with microsecond accuracy (seeing Fig. 1.7 for details).

(a) DAVIS (b) Captured image (c) Recorded event

Figure 1.7: *Event cameras are bio-inspired sensors that asynchronously report logarithmic intensity changes. Each pixel in DAVIS contains circuitry that allows an active pixel sensor (APS) intensity image readout and a dynamic vision sensor (DVS) event generation from the same photoreceptor. The APS is a standard global shutter camera that operates independently of the DVS. As shown in (b) and (c), when APS captured only a heavily blurred image,* $>$ 4000 *red/blur rendered events have been recorded by the DVS of a* 20*ms time slice.*

Taking full advantage of the high temporal resolution event stream would contribute to the high frame-rate video reconstruction. As we mentioned in Section 1.1.2.2, blur causes undesired image degradation while also encodes the relative motion between the camera and the observed scene. Besides, the event stream encodes

the motion information by measuring the precise pixel-by-pixel intensity changes. Therefore, blur and event streams are connected under the framework of representing the dynamic scenes. In other words, the dynamic scene can be modelled through an event camera, while the blur uses convolution with a blur kernel to express the dynamic scene.

With the above discussion, in this thesis, we first propose a simple and effective approach the **Event-based Double Integral (EDI)** model, to reconstruct a high framerate, sharp video (>1000 fps) from a single blurry frame and its event data. The video generation is based on solving a simple non-convex optimization problem in a single scalar variable. Then, we improved the EDI model to the **multiple Event-based Double Integral (mEDI)** model by using multiple images and their events to handle flickering effects and noise in the generated video. Furthermore, we provide a new and more efficient solver to minimize the proposed energy model. Last, we propose a method that estimates the flow and the sharp image jointly, based on a single blur image and its events with an objective function that uses the primal-dual algorithm.

In summary, this thesis intends to explore the physical image formation process and to restore the sharp image/video with different data sources. For image restoration, we explore the physical image formation process and solve the problem with a single image, RGBD image, and stereo videos. For video reconstruction, which is reversing the blurring process, we use the event camera and proposed efficient yet simple approaches.

1.2 Related Work

We first review works for motion restoration based on a single blurred image or video. Then, we discuss works for video reconstruction with an event camera. We further discuss methods for optical flow estimation that closely related to our settings.

1.2.1 Single-image deblurring.

Existing blind single image-based deblurring methods tend to formulate the problem within the Maximum A Posteriori (MAP) framework, where the blur kernel and the latent sharp image are optimized jointly. To resolve the ill-posed underlining optimization problem, various assumptions, or regularizations, have been proposed for the blur kernel and the desired latent image, such as the dark channel prior Pan et al. [2016b], extreme channel prior Yan et al. [2017a], l_0 regularized prior Pan et al. [2014]; Xu et al. [2014], learned image prior using a CNN Li et al. [2018b], uniform blur Levin et al. [2009]; Xu et al. [2013], non-uniform blur from multiple homographies Hu et al. [2014]; Pan et al. [2016a], constant depth Gupta et al. [2010]; Xu and Jia [2012], in-plane rotation Sun et al. [2015], and forward motion Zheng et al. [2013]. The resultant optimisation problem is non-convex in general. The blur kernel and the latent image are usually solved in an alternating fashion. Thus, a proper and effective initialisation is demanded to achieve a good *local optimum solution* and makes

the algorithm converge quickly.

A few works have exploited the layer-wise scene structure to model the blur kernel Whyte et al. [2012]; Gupta et al. [2010]; Hu et al. [2014]; Pan et al. [2016a]. Gupta et al. [2010] represents the camera motion trajectory using a motion density function, which requires a constant depth or fronto-parallel scene assumption. Hu et al. [2014] jointly estimating the depth layering and remove the blur caused by an in-plane motion from a single blurred image. Pan et al. [2016a] estimate object segmentation and camera motion jointly by incorporating soft segmentation. Noting that both approaches require user input for initial depth layer segmentation, I recognise it is still hard for traditional single-image-based methods to model spatially-varying blur under high dynamic scenes.

Recently, learning-based methods have brought significant improvements in image deblurring. Sun et al. [2015] proposed a convolutional neural network (CNN) to estimate patch-level linear blur kernels and then restored the latent image by the estimated blur prior. Gong et al. [2017b] learned optical flow from a single blurred image through a fully-convolutional deep neural network. The blur kernel is then obtained from the estimated optical flow to restore the sharp image. Nah et al. [2017] proposed a multi-scale CNN that restores latent images in an end-to-end learning manner without assuming any restricted blur kernel model. Tao et al. [2018] proposed a scale-recurrent network that leverage the "coarse-to-fine" scheme to deblur an image. Zhang et al. [2019] presented a deep hierarchical multi-patch network inspired by Spatial Pyramid Matching to deal with blurred images via a fine-to-coarse hierarchical representation. Though the results are impressive, I suggest those approaches are not always faithful to the content of the latent image. Moreover, existing deep neural network-based motion deblurring methods either exploit more frames or exploit large scale datasets, which may hinder the generalisation ability.

1.2.2 Multi-image deblurring.

Multi-image provides flow or depth information which allows researchers to model the pixel-wise blur kernel better. Cho et al. [2012] proposed a method relying on the assumption that salient sharp frames frequently exist in videos, which only allows for slowly moving objects in dynamic scenes. Wulff and Black [2014] proposed a layered model to estimate both foreground motion and background motion. However, these motions are restricted to affine models, and it is difficult to extend them to multi-layer scenes due to the difficulty in the depth order. Kim and Lee [2014] incorporated optical flow estimation to guide the blur kernel estimation, which can deal with certain object motion blur. In Kim and Lee [2015], a new method is proposed to simultaneously estimate optical flow and tackle general blur by minimising a single non-convex energy function. Sellent et al. [2016] proposed a stereo video deblurring technique, where 3D scene flow is estimated from the blurred images using a piecewise rigid scene representation. Pan et al. [2017b] proposed a single framework to estimate the scene flow and deblur the images jointly.

Recently, video-based methods aim to use neural network architectures to exploit

the temporal relationship between neighbouring frames. Chen et al. [2018] fine-tune existing deblurring neural networks in a self-supervised fashion by enforcing that the output, when blurred based on the optical flow between sub-sequent frames, matches the input blurred image. Nah et al. [2019a] improve the accuracy of recurrent models by iteratively updating the hidden states transferred from past frames to the frame being processed so that the relations between video frames could be better used. Zhou et al. [2019a] propose a STFAN Network for the alignment and deblurring in a unified framework, where they take both blurred and restored images of the previous frame as well as the blurred image of the current frame as input. However, they train a block in the deep net to learn the warping and alignment ability that limits the generalisation ability of the proposed method. Zhou et al. [2019c] propose DAVANet to exploit the two-view nature of stereo images, where 3D scene cues from the depth and varying information from two views are incorporated to help to remove complex spatially-varying blur in dynamic scenes. Though these methods take into account the physical image formation process, the image warping and the alignment are not easily applied to the network architecture.

1.2.3 Event-based image reconstruction.

The event camera report log intensity changes and outputs a continuous, asynchronous stream of events that encodes non-redundant information about local brightness change. Estimating intensity images from events is important. The reconstructed images grant computer vision researchers a readily available high temporal resolution, high-dynamic-range imaging platform that can be used for tasks such as face-detection Barua et al. [2016], moving object segmentation Stoffregen et al. [2019], localization Liu et al. [2017b, 2019a]; Liu and Li [2019] and optical flow estimation Zhu et al. [2018a]; Pan et al. [2020b]. Although several works try to explore the advantages of the high temporal resolution provided by event cameras Zhu et al. [2017]; Gehrig et al. [2018]; Kueng et al. [2016a]; Gallego et al. [2019]; Brandli et al. [2014c], how to make the best use of the event camera has not yet been fully investigated.

A typical way for image reconstruction is achieved by processing a spatio-temporal window of events. Taking a spatio-temporal window of events imposes a latency cost at minimum equal to the length of the time window, and choosing a time-interval (or event batch size) that works robustly for all types of scenes is not trivial. Barua et al. [2016] generate image gradients by dictionary learning and obtain a logarithmic intensity image via Poisson reconstruction. Bardow et al. [2016] simultaneously optimise optical flow and intensity estimates within a fixed-length, sliding spatio-temporal window using the primal-dual algorithm Posch et al. [2010]. Cook et al. [2011] integrate events into interacting maps to recover intensity, gradient, and optical flow while estimating global rotating camera motion. Kim et al. [2014] reconstruct high-quality images from an event camera under a strong assumption that the only movement is pure camera rotation, and later extend their work to handle 6-degree-of-freedom motion and depth estimation Kim et al. [2016]. Reinbacher et al.

[2016] integrate events over time while periodically regularising the estimate on a manifold defined by the timestamps of the latest events at each pixel. Optimisation based event-only methods (i.e., without the process of learning from training data) will generate artefacts and lack of texture when event data is sparse because they cannot integrate sufficient information from the available sparse events. Recently, learning-based approaches have improved the image reconstruction quality significantly with powerful event data representations via deep learning Rebecq et al. [2019, 2020]; Wang et al. [2019]; Scheerlinck et al. [2020]. Rebecq et al. [2019] propose E2VID, a fully convolutional, recurrent UNet architecture to encode events in a spatio-temporal voxel grid. In Rebecq et al. [2020], they propose a recurrent network to reconstruct videos from a stream of events, and they incorporate stacked ConvLSTM gates, which prevent vanishing gradients during backpropagation for long sequences. Wang et al. [2019] form a 3D event volume by stacking event frame in a time interval. A reconstructed intensity frame is generated by summing events at each pixel in a smaller time interval.

To achieve more image details in the reconstructed images, several methods that aim to combine events with intensities have been proposed. The DAVIS Brandli et al. [2014a] uses a shared photo-sensor array to simultaneously output events (DVS) and intensity images (APS). Brandli et al. [2014b] combine images and event streams from the DAVIS camera to create inter-frame intensity estimates by dynamically estimating the contrast threshold (temporal contrast) of each event. Each new image frame resets the intensity estimate, preventing excessive growth of integration error. However, it also discards important accumulated event information. Scheerlinck et al. [2018] propose an asynchronous event-driven complementary filter to combine APS intensity images with events, and obtain continuous-time image intensities. Shedligeri and Mitra [2019] first exploit two intensity images to estimate depth. Second, they only use events to reconstruct a pseudo-intensity sequence (using method Reinbacher et al. [2016]) between the two intensity images. They, taking the pseudo-intensity sequence, they estimate the ego-motion using visual odometry. With the estimated 6-DOF pose and depth, they directly warp the intensity image to the intermediate location. Liu et al. [2017a] assume a scene should have a static background. Thus, their method needs an extra sharp static foreground image as input, and the event data are used to align the foreground with the background.

1.2.4 Event camera-based flow estimation.

Benosman et al. [2012] propose an adaptation of the gradient-based Lucas-Kanade algorithm based on DVS. In Benosman et al. [2013], they assume that the flow orientation and amplitude can be estimated using a local differential approach on the surface defined by coactive events. They work well for sharp edges and monochromatic blocks but fail with dense textures, thin lines, and more complicated scenes. Barranco et al. [2015] propose a more expensive phase-based method for high-frequency texture regions and reconstructing the intensity signals to avoid the problem with textured edges. Bardow et al. [2016] jointly reconstruct intensity image and estimate

flow based on events by minimising their objective function. However, accuracy relies on the quality of the reconstructed image. Gallego et al. [2018] present a unifying framework to estimate flow by finding the point trajectories on each image plane that are best aligned with events. Note that its computational complexity increases linearly with the number of events. Zhu propose the EV-FlowNet Zhu et al. [2018a], an event-based flow estimation approach using a self-supervised deep learning pipeline. The event data are represented as 2D frames to feed the network. While images from the sensor are used as a supervision signal, the blur effect is ignored which is shown to be useful for flow estimation in our framework. In Zhu et al. [2019], they further use another event format to train two networks to predict flow, camera ego-motion, and depth for static scenery. Then, they use predictions to remove motion blur from event streams which shows the potential of blurring to improve the flow estimate accuracy. However, flow computed at those constant brightness regions is still less reliable.

1.2.5 Image-based flow estimation.

Menze and Geiger [2015] proposed a novel model and dataset for 3D scene flow estimation with an application to autonomous driving. Pan et al. [2017b] proposed a single framework to estimate the scene flow and deblur the images jointly. Taniai et al. [2017] presented a multi-frame method for efficiently computing scene flow (dense depth and optical flow) and camera ego-motion for a dynamic scene observed from a moving stereo camera rig.

One promising direction is to learn optical flow with CNNs. Yin and Shi [2018] propose an unsupervised GeoNet for jointly estimating monocular depth, optical flow, and camera motion from video. PWC-Net Sun et al. [2018] use the current optical flow estimate to warp the CNN features of the second image. Then the warped features and features of the first image are applied to construct a cost volume, which is processed by a CNN to estimate the optical flow. The FlowNet by Dosovitskiy et al. [2015] represented a paradigm shift in optical flow estimation. The work shows the feasibility of directly estimating optical flow from raw images using a generic U-Net CNN architecture. FlowNet 2.0 Ilg et al. [2017] develop a stacked architecture that includes warping of the second image with the intermediate optical flow, which decreases the estimation error by more than 50% than the original FlowNet. SelFlow Liu et al. [2019b] is based on distilling reliable flow estimations from non-occluded pixels and using these predictions to guide the optical flow learning for hallucinated occlusions.

Several learning-based works attempt to use a single image to estimate flow Walker et al. [2015]; Rosello [2016]; Endo et al. [2019]. Walker et al. [2015] use CNN to predict dense flow, while they assume the image is static. Gong et al. [2017a] directly estimates the motion flow from a blurred image through a fully-convolutional deep neural network (FCN) and recover the unblurred image from the estimated motion flow. This is the first universal end-to-end mapping from the blurred image to the dense motion flow.

1.3 Thesis Outline

This thesis is formatted as a compilation of my publications during my PhD period at the Australian National University.

In this thesis, we tackle the following key challenges:

- Single image restoration in the frequency domain
- Camera motion estimation and image restoration with RGBD images
- Scene flow estimation and restoration with stereo videos
- High frame-rate video reconstruction with a DAVIS sensor
- Image reconstruction and flow estimation with a DAVIS sensor

In Chapter 2, we propose a single-image-based deblurring method by studying the problem in the frequency domain with the *phase-only image* and further extend our approach to handle non-uniform blur, which involves spatially varying blur kernels.

In Chapter 3, with a RGBD image, we then propose to jointly estimate the 6 DoF camera motion and remove the non-uniform blur caused by camera shake by exploiting their underlying geometric relationships, with a single blurry image and its depth map (either direct depth measurements or a learned depth map) as input.

In Chapter 4, given consecutive blurred stereo video frames, we propose a method to recover latent clean images, estimate the 3D scene flow, and segment the multiple moving objects simultaneously. By exploiting the blur model constraint, the moving objects and the 3D scene structure, we reach an energy minimization formulation for joint deblurring, scene flow estimation and moving object segmentation.

In Chapter 5, we first propose a simple and effective approach, the Event-based Double Integral (EDI) model, to reconstruct a high frame-rate, sharp video ($>$1000 fps) from a single blurry frame and its event data. The video generation is based on solving a simple non-convex optimization problem in a single scalar variable.

In Chapter 6, we improve the EDI model to the multiple Event-based Double Integral (mEDI) model by using multiple images and their events to handle flickering effects and noise in the generated video. Furthermore, we provide a more efficient solver to minimize the proposed energy model. We significantly reduce the computational complexity with the Fibonacci sequence.

In Chapter 7, we propose a method to estimate optical flow and the sharp image jointly, from a single blur image and its events with an objective function using the primal-dual algorithm. In doing so, we introduce an *event-based brightness constancy* constraint on absolute intensity and use the blur formation model in our objective function.

In Chapter 8, we conclude our thesis and provide some future research directions.

1.3.1 Contributions

Papers:

1. **Liyuan Pan**, Richard Hartley, Miaomiao Liu, Yuchao Dai. Phase-Only Image Based Kernel Estimation for Single Image Blind Deblurring. Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
2. **Liyuan Pan**, Yuchao Dai, Miaomiao Liu, Fatih Porikli. Joint Deblurring and Camera Motion Estimation from a Single Blurry Image. Winter Conference on Applications of Computer Vision (WACV), 2019.
3. **Liyuan Pan**, Yuchao Dai, Miaomiao Liu, Fatih Porikli, Quan Pan. Stereo Video Deblurring and Moving Objects Segmentation. Transactions on Image Processing (TIP), 2019.
4. **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. Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
5. **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 (TPAMI), 2020.
6. **Liyuan Pan**, Miaomiao Liu, and Richard Hartley. Single Image Optical Flow Estimation with an Event Camera. Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

Challenges:

1. **NaN_ROB**. 4^{th} place of Robust Vision Challenge (Stereo), 2018
2. Liu Liu, **Liyuan Pan**, Hongdong Li. 3^{rd} (/218) place of Google Landmark Retrieval Challenge, Kaggle, 2018

Chapter 2

Phase-only Image Based Kernel Estimation for Single Image Blind Deblurring

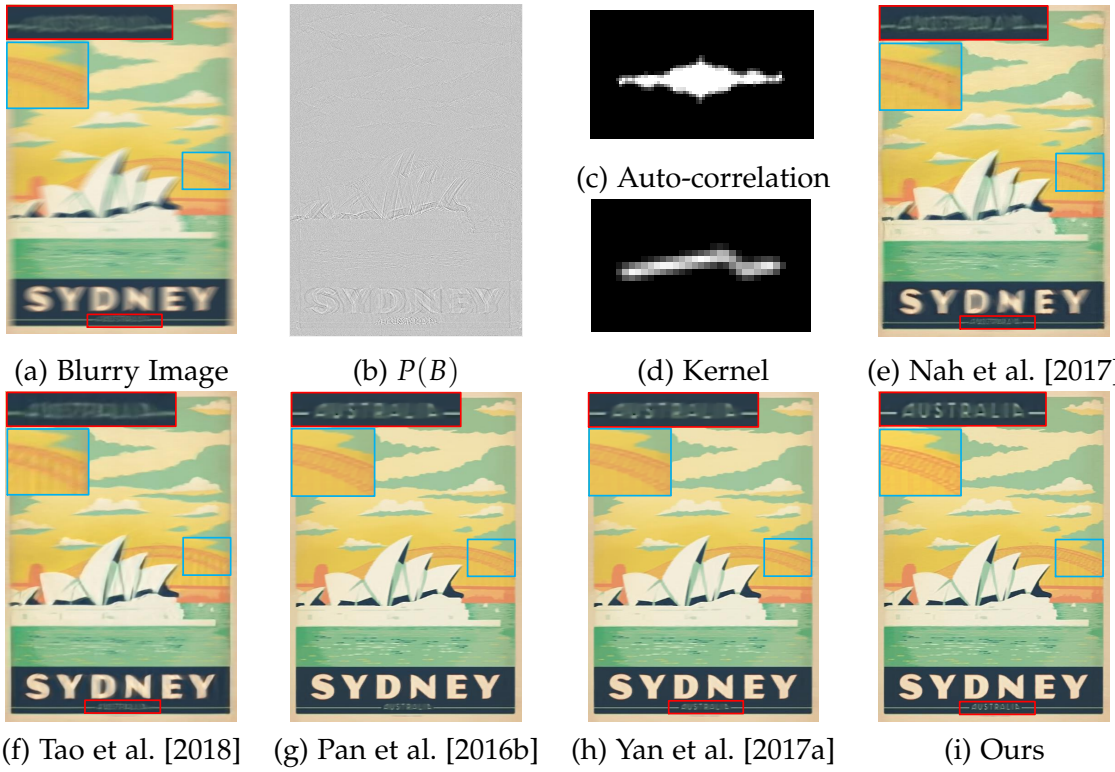
Single image blind deblurring is an ill-posed problem as infinite pairs of blur kernels and images could generate the same blurry image. This chapter aims to estimate a high-quality blur kernel directly from the input image with motion blur by studying the problem in the frequency domain to handle the difficulties and regularizing the solution space. We exploit the *phase-only image* of the blurred input image, reconstructed from the Fourier transformed image using the phase information only. The *phase-only image* provides information about the blur kernel, thereby leading to a new approach to estimating the blur kernel.

Liyuan Pan, Richard Hartley, Miaomiao Liu, Yuchao Dai. Phase-Only Image Based Kernel Estimation for Single Image Blind Deblurring. Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

2.1 Abstract

The image motion blurring process is generally modelled as the convolution of a blur kernel with a latent image. Therefore, the estimation of the blur kernel is essentially important for blind image deblurring. Unlike existing approaches which focus on approaching the problem by enforcing various priors on the blur kernel and the latent image, we are aiming at obtaining a high quality blur kernel directly by studying the problem in the frequency domain. We show that the auto-correlation of the absolute *phase-only image*[1] can provide faithful information about the motion (e.g., the motion direction and magnitude, we call it the *motion pattern* in this chapter.) that caused the

[1]Phase-only image means the image is reconstructed only from the phase information of the blurry image.



(a) Blurry Image (b) $P(B)$ (c) Auto-correlation (d) Kernel (e) Nah et al. [2017]

(f) Tao et al. [2018] (g) Pan et al. [2016b] (h) Yan et al. [2017a] (i) Ours

Figure 2.1: *Our deblurring result compared with the state-of-the-art methods. (a) Input blurry image. (b) The* phase-only image*. (c) The auto-correlation for the* phase-only image*. (d) The estimated blur kernel. (e) Deblurring result of Nah et al. [2017]. (f) Deblurring result of Tao et al. [2018]. (g) Deblurring result of Pan et al. [2016b]. (h) Deblurring result of Yan et al. [2017a]. (i) Our deblurring result. (Best viewed on screen).*

blur, leading to a new and efficient blur kernel estimation approach. The blur kernel is then refined and the sharp image is estimated by solving an optimization problem by enforcing a regularization on the blur kernel and the latent image. We further extend our approach to handle non-uniform blur, which involves spatially varying blur kernels. Our approach is evaluated extensively on synthetic and real data and shows good results compared to the state-of-the-art deblurring approaches.

2.2 Introduction

Blind image deblurring aims at estimating the blur kernel and the latent image from an input blurry image. This is an ill-posed problem as there are infinitely many pairs of blur kernels and images that could generate the same blurry image. Blind image deblurring has been extensively studied in computer vision and is still a very active research area Kim and Lee [2015]; Sellent

et al. [2016]; Gong et al. [2017b]; Pan et al. [2017b]; Nah et al. [2017]; Tao et al. [2018], where blur kernel estimation is essentially important in obtaining a high quality sharp image.

Existing blind image deblurring methods tend to formulate the problem within the Maximum A Posteriori (MAP) framework, where the blur kernel and the latent sharp image are optimized jointly. To resolve the ill-posed underlining optimization problem, various assumptions, or regularizations, have been proposed for the blur kernel and the desired latent image, such as the dark channel prior Pan et al. [2016b], extreme channel prior Yan et al. [2017a], l_0 regularized prior Pan et al. [2014]; Xu et al. [2014], learned image prior using a CNN Li et al. [2018b], uniform blur Levin et al. [2009]; Xu et al. [2013], non-uniform blur from multiple homographies Hu et al. [2014]; Pan et al. [2016a], constant depth Gupta et al. [2010]; Xu and Jia [2012], in-plane rotation Sun et al. [2015], and forward motion Zheng et al. [2013]. The resultant optimization problem is non-convex in general. The blur kernel and the latent image are usually solved in an alternating fashion. Thus, a proper and effective initialization is demanded to achieve a good *local optimum solution* and makes the algorithm converge quickly.

In this chapter, we aim at estimating a high-quality blur kernel directly from the input image with motion blur by studying the problem in the frequency domain. We exploit the *phase-only image* of the input blurry image, which is reconstructed from the Fourier transformed image using the phase information only. The *phase-only image* contains edge and texture information about the image structure Oppenheim and Lim [1981]; Papari and Petkov [2011]. The motion (either camera or object motion) information is encoded as repeated image edges in the *phase-only image* (see Fig. 2.1 for an example). We show that the auto-correlation of the absolute *phase-only image* reveals the motion information including the motion direction and motion magnitude, which is referred to as the *motion pattern* in this chapter. It provides information about the blur kernel, thereby leading to a new approach to estimating the blur kernel.

We further improve the blur kernel and latent image estimation by enforcing a spatial sparsity prior on the kernel as well as the latent image gradient in a simple optimization framework. Furthermore, our blur kernel estimation approach can be naturally extended to handle non-uniform blur in order to deal with the spatially-variant blur kernels that arise in complex image deblurring problems. Extensive experiment on both synthetic and real images demonstrate the superiority of our approach over the state-of-the-art methods.

Our main contributions are summarized as follows

1) We propose a new *phase-only image*-based approach to directly estimating the blur kernel from the input blurry image. The approach for *motion*

pattern estimation is easy and efficient, consisting of a few lines of code.

2) Our single-image blind deblurring model can be naturally extended to handle non-uniform blur in an effective manner. Furthermore, the estimated blur kernel can be easily refined by only enforcing spatial sparsity.

3) Evaluated on both synthetic and real images, our proposed approach shows impressive results compared to other state-of-the-art blind deblurring approaches.

2.3 Related Work

Single-image blind deblurring. Single-image deblurring jointly estimates the blur kernel and the latent sharp image from the blurry one, which is highly under-constrained since the blurry image could be explained by many pairs of blur kernel and sharp image Ji and Liu [2008]; Pan et al. [2019a]. In general, image deblurring is formulated in a MAP framework with *priors* on blur kernels or latent images. The Sparsity prior has proved effective in blur kernel estimation. For instance, Krishnan et al. [2011] applied normalized sparsity in their MAP framework to estimate the blur kernel. Xu et al. [2013] proposed an approximation of the l_0-norm as a sparsity prior in order to jointly estimate sharp image and blur kernels. Edge-based methods for blur kernel estimation have been exploited recently Xu and Jia [2010]; Joshi et al. [2008]; Cho and Lee [2009]; Sun et al. [2013]. Xu and Jia [2010] proposed a two-phase method for single-image deblurring. The blur kernel is first estimated based on the selected image edges and refined by ISD optimization. The latent sharp image is then restored by total-variation (TV)-l_1 deconvolution. In addition, a Gaussian prior is imposed to help the estimation of the blur kernel Joshi et al. [2008]; Cho and Lee [2009], which leads to an efficient solver. Moreover, the blur kernel has been modelled based on various motion assumptions, such as in-plane camera rotation Sun et al. [2015] or camera forward motion Zheng et al. [2013]. A few works have exploited the layer-wise scene structure to model the blur kernel Gupta et al. [2010]; Hu et al. [2014]; Pan et al. [2016a]. Gupta et al. [2010] represent the camera motion trajectory using a motion density function, which requires a constant depth or fronto-parallel scene assumption. Hu et al. [2014] proposed jointly estimating the depth layering and remove the blur caused by in-plane motion from a single blurry image. Pan et al. [2016a] proposed jointly estimating object segmentation and camera motion by incorporating soft segmentation. Note that both approaches require user input for initial depth layer segmentation.

Video image blind deblurring. In order to better model non-uniform blur, monocular video and stereo based deblurring approaches are proposed to handle blurring in realistic scenes Pan et al. [2018]; Xu and Jia [2012]. Cho

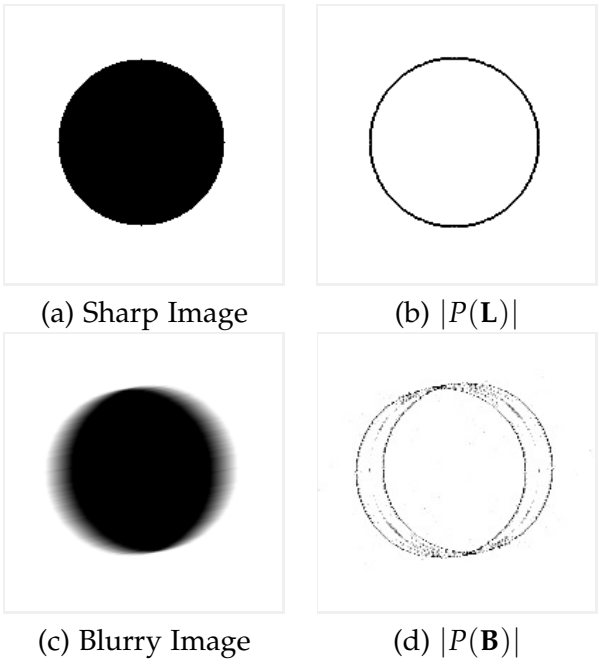
et al. [2012] proposed a method relying on the assumption that salient sharp frames frequently exist in videos, which only allows for slowly moving objects in dynamic scenes. Wulff and Black [2014] proposed a layered model to estimate both foreground motion and background motion. However, these motions are restricted to affine models, and it is difficult to extended them to multi-layer scenes due to the difficulty in depth ordering. Kim and Lee [2014] incorporated optical flow estimation to guide the blur kernel estimation, which is able to deal with certain object motion blur. In Kim and Lee [2015], a new method is proposed to simultaneously estimate optical flow and tackle general blur by minimizing a single non-convex energy function. Stereo images and videos can provide depth information which allows to better model pixel-wise blur kernel. Sellent et al. [2016] proposed a stereo video deblurring technique, where 3D scene flow is estimated from the blurry images using a piecewise rigid scene representation. Pan et al. [2017b] proposed a single framework to jointly estimate the scene flow and deblur the images.

Deep learning based image deblurring. Recently, image deblurring has greatly benefited from the great advances in deep learning Kupyn et al. [2018a]; Sun et al. [2015]; Zhang et al. [2018]; Tao et al. [2018]. Sun et al. [2015] proposed a convolutional neural network (CNN) to estimate locally linear blur kernels. Gong et al. [2017b] learned optical flow field from a single blurry image directly through a fully-convolutional deep neural network. The blur kernel is then obtained from the estimated optical flow which is applied in an MAP framework to restore the sharp image. Su et al. [2017] trained an end-to-end CNN to accumulate information across frames for video deblurring. Nah et al. [2017] proposed a multi-scale CNN that restores latent images in an end-to-end learning manner without any assumption on the blur kernel model. Li et al. [2018b] used a learned image prior to distinguish whether an image is sharp or not and embedded the learned prior into the MAP framework. Tao et al. [2018] proposed a light and compact network, SRN-DeblurNet, to deblur the image. While achieving reasonable performance on various scenarios, the success of these deep learning based methods depends on the consistency between the training datasets and the testing datasets, which can hinder the generalization ability.

2.4 Method

2.4.1 Fourier Theory of Phase-only Images

This section contains the main theoretical insights of this paper. Our goal is to find the latent sharp image from a single blurry image. The blurry image



(a) Sharp Image (b) $|P(\mathbf{L})|$

(c) Blurry Image (d) $|P(\mathbf{B})|$

Figure 2.2: *We use a circle image as an example. The image is blurred by a linear kernel, where the kernel length is 20 pixels and the direction is 10 degree.*

can be modelled as a convolution of the latent image with a blur kernel,

$$\mathbf{B} = \mathbf{L} \otimes \mathbf{k}, \tag{2.1}$$

where $\mathbf{B}$ is the known blurry image, $\mathbf{L}$ denotes the latent sharp image, $\mathbf{k}$ is the blur kernel, $\otimes$ is the convolution operator. Note that this problem is highly under-determined since multiple pairs of $\mathbf{L}$ and $\mathbf{k}$ can lead to the same blurry image.

In the Fourier domain, Eq. (2.1) corresponds to $\mathcal{F}(\mathbf{B}) = \mathcal{F}(\mathbf{L}) \odot \mathcal{F}(\mathbf{k})$, where $\odot$ represents the component-wise multiplication.

The phase and amplitude of a complex number $z = ke^{i\theta}$ are $e^{i\theta}$ and $k \geq 0$ respectively. Applying these component-by-component to a Fourier transformed image $\mathcal{F}(\mathbf{L})$ gives the phase and amplitude components. We denote taking the phase of a complex signal by $\mathcal{P}(\cdot)$. Taking the inverse Fourier transform of the phase-component gives the *phase-only image*, $P(\mathbf{L}) = \mathcal{F}^{-1}(\mathcal{P}(\mathcal{F}(\mathbf{L})))$. It is well known that the phase-only image bears more similarity to the original image than the analogously defined amplitude image. Fig. 2.2 shows an example of the phase-only image derived from a clean and blurry image. As may be observed, taking a phase-only image acts as a sort of edge-extractor. This is related to the fact, noted in Kovesi [2003] that the Fourier components of an edge tend to be in-phase with each other. For a real

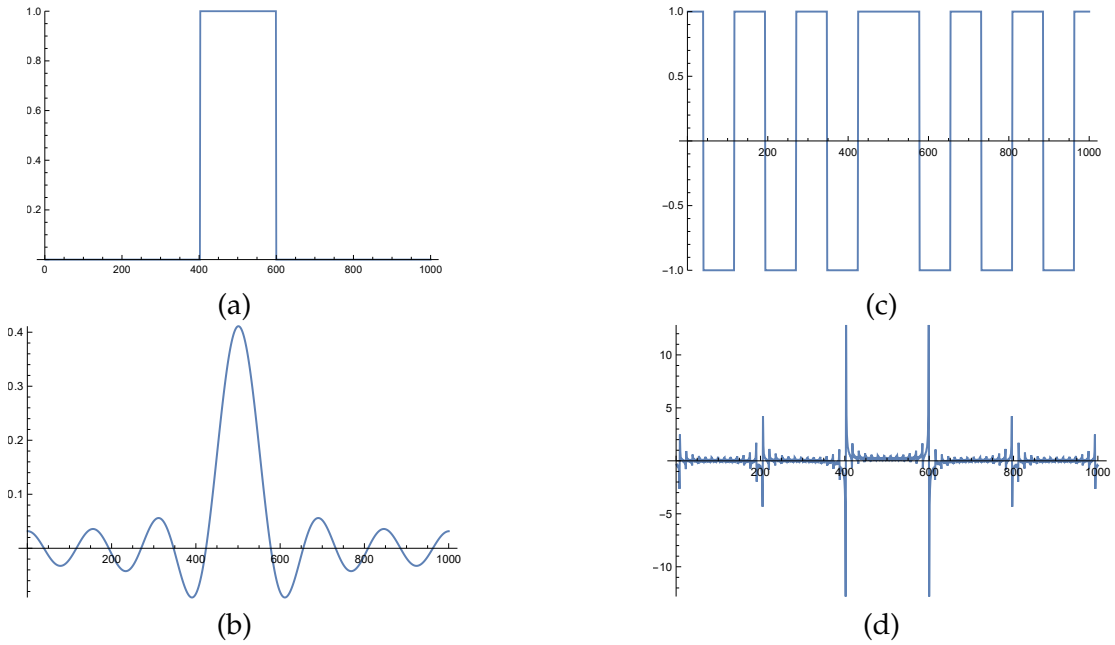


Figure 2.3: *Given a top-hat function (a), its fourier transform is a sinc shown in (b). (The central peak has twice the width of the others. Note that since the top-hat is symmetric, its Fourier transform is real, hence its phase is either $+1$ or -1 shown in (c).) The phase-only image of the top-hat shown in (d) is obtained by taking the inverse Fourier transform of the function in (c).*

image $\mathbf{L}$, the phase-only image will also be real. Another simple property is *rotation-covariance*: if R represents rotation then $P(R(\mathbf{L})) = R(P(\mathbf{L}))$. It is also shift-covariant.

We now make a basic observation regarding the phase-only image of a convolution.

lemma 1. *The phase-only image of a convolution $P(\mathbf{L} \otimes \mathbf{k})$, equals the convolution of the phase-only image and the phase-only kernel.*

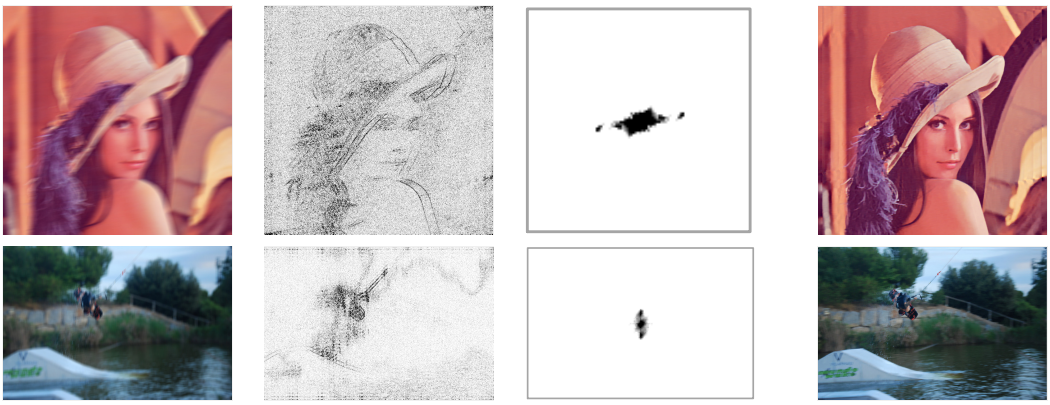
$$P(\mathbf{L} \otimes \mathbf{k}) = \mathcal{F}^{-1}(\mathcal{P}(\mathcal{F}(\mathbf{L} \otimes \mathbf{k}))) = P(\mathbf{L}) \otimes P(\mathbf{k}) \; . \tag{2.2}$$

This results from a simple calculation.

Linearly-blurred image. For a simple linear (straight-line) blur kernel, the form of $P(\mathbf{k})$ can be computed. By rotation and shift covariance, it may be assumed without loss of generality, that $\mathbf{k}$ is axis-aligned, in which case $\mathbf{k}(x, y) = \delta(y)H(x)$, where $\delta(y)$ is a Dirac delta function and $H(x)$ is a top-hat. The Fourier transform is separable, so it follows that $P(\mathbf{k})(x, y) = \delta(y)P(H)(x)$. Hence, we investigate what the $1D$ phase-only signal $P(H)$ is. The result is shown in Fig. 2.3.

According to Eq. (2.2), if $\mathbf{B} = \mathbf{L} \otimes \mathbf{k}$, then $P(\mathbf{B})$ is obtained by convolving

$P(\mathbf{L})$ in the orientation of the linear kernel with the phase-only kernel, shown in Fig. 2.3(d). This results in the creation of multiple copies ("ghosts"), of the phase-only image, $P(\mathbf{L})$, separated by the width of the filter. (The copies due to the principal peaks will be the most noticeable.)[2] This is shown in Fig. 2.4.



(a) Blurry Image (b) $|P(\mathbf{B})|$ (c) $\mathcal{A}(|P(\mathbf{B})|)$ (d) Deblurring Results

Figure 2.4: *(a) Input blurry images, the top one is a synthetic image created by ourselves and the bottom one is a real image from dataset Shi et al. [2014]. (b) The absolute phase-only image of the blurry image, $|P(\mathbf{B})|$, results in two principal copies (others more faint) of $P(\mathbf{L})$. (c) The autocorrelation of the absolute phase-only image, $\mathcal{A}(|P(\mathbf{B})|)$, showing two distinct peaks (separated by the length of the filter kernel). Distinguishing the two principal peaks of the autocorrelation (apart from the origin) can be used to determine the orientation and width of a linear (straight-line) blur kernel. (d) shows our deblurring results with sharp edges.*

The key advantage of phase-only image. This analysis and the examples show the advantage and purpose in considering the phase-only image as a means of determining the blur kernel, and subsequently deblurring the image. This is illustrated by the analysis of the linear kernel.

The effect of blurring is to smear the image in the blur direction, as shown in Fig. 2.4 (top left). From this image, it is not easy to discern the shape of the kernel, particularly the linear extent of the kernel. On the other hand, in the phase-only image, the effect of blurring is to create *two principal identical copies* of $P(\mathbf{L})$ separated by the extent of the blur kernel. This is immediately evident from Fig. 2.4(b), or Fig. 2.2(d). Thus, the continuous smear in the blurred image is replaced by a simple sum of two (principle) copies in the phase-only blurred image. This simplification of the effect of blurring makes the further image-processing to compute the blur-kernel much simpler.

This discovery of the application of the phase-only image to deblurring is the key original contribution of this paper.

[2] A more exact statement is that $P(\mathbf{B})$ consists of multiple ghosts, separated by the filter width, of the **gradient** of $P(\mathbf{L})$ in the filter direction.

2.4.2 Autocorrelation

Using phase-only to obtain $P(\mathbf{B})$ from a blurry image results in multiple (two principal) shifted copies of $P(\mathbf{L})$. Note that $P(\mathbf{L})$ is not known. However, this suggests the use of autocorrelation of $P(\mathbf{B})$.

Autocorrelation of a signal $\mathbf{I}$ (1 or 2-dimensional) is computed using Fourier transform as:

$$\mathcal{A}(\mathbf{I}) = \mathcal{F}^{-1}(\mathcal{F}(\mathbf{I}) \odot \overline{\mathcal{F}(\mathbf{I})}).$$

Unfortunately, if $\mathbf{I}$ is itself a phase-only image, derived from $\mathbf{J}$, then

$$\mathcal{F}(\mathbf{I}) = \mathcal{F}(\mathcal{F}^{-1}\mathcal{P}(\mathcal{F}(\mathbf{J}))) = \mathcal{P}(\mathcal{F}(\mathbf{J})).$$

So $\mathcal{A}(\mathbf{I}) = \mathcal{F}^{-1}(\mathcal{P}(\mathcal{F}(\mathbf{J})) \odot \overline{\mathcal{P}(\mathcal{F}(\mathbf{J}))}) = \mathcal{F}^{-1}(1) = \delta$
where δ is a Dirac delta function at the origin. In other words, a phase-only image **is completely un-selfcorrelated**.

In other words, we cannot derive any information whatever from the autocorrelation of a phase-only image. The solution is to use the absolute value of the phase-only image instead. In other words, we compute $\mathcal{A}(|P(\mathbf{B})|)$, which should show the desired behaviour.

Fig. 2.4 shows the absolute *phase-only image* $|P(\mathbf{B})|$ and its autocorrelation $\mathcal{A}(|P(\mathbf{B})|)$. It is noticed that multiple copies of $|P(\mathbf{L})|$ are shown in $|P(\mathbf{B})|$. The most noticeable repeated edges are due to the principal peak of $P(\mathbf{k})$ (as analyzed above) indicating the start and end point of the moving camera.

The autocorrelation of the absolute *phase-only image* shows several bright points that indicate the motion of the camera, e.g., the motion direction and magnitude, which is referred to as *motion pattern*. The autocorrelation image will consist of a central peak plus two side-peaks separated by the extent (and in the direction) of the blur-kernel.

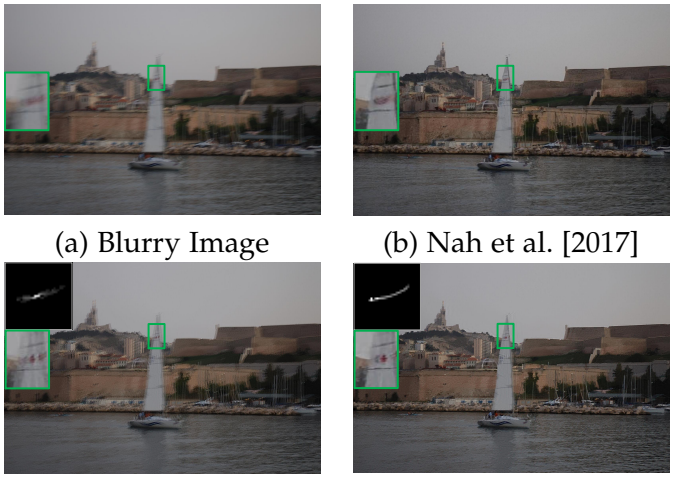
Consequently, the motion of the camera will provide faithful information for obtaining the blur kernel. Therefore, in the following section, we will present our approach to image deblurring based on the analysis of the autocorrelation of the absolute *phase-only image*.

2.5 Uniform Deblurring

Based on the analysis of the Fourier theory of phase-only images, we introduce our approach to estimate the blur kernel and deblur the images.

2.5.1 Uniform Blur from Linear Motion

Consider the blur caused by a pure linear motion. By computing the autocorrelation of the absolute *phase-only image*, the *motion pattern*, namely the motion direction and the motion magnitude, is extracted by directly connecting the



(a) Blurry Image (b) Nah et al. [2017]

(c) Coarse Kernel (d) Refined Kernel

Figure 2.5: *(a) The blurry image from dataset Pan et al. [2016b]. (b) Deblurring results of Nah et al. [2017]. (c) Our deblurring result with the coarse blur kernel built from the autocorrelation of the absolute phase-only image. (d) Our deblurring result with the refined kernel. The refined kernel can better improve the deblurring result by looking at the close-up of the part of the sail with detailed sharp edges. Note that the blur kernel is zoomed in the corner.*

two end bright points in $\mathcal{A}(|P(\mathbf{B})|)$. The blur kernel is then formed based on the extracted *motion pattern*. In particular, the motion magnitude determines the kernel size. The non-zero kernel values are uniformly distributed along the motion direction (see Fig. 2.4 the top row for an example). Given the built blur kernel, the latent image can be easily obtained by solving the Eq. (2.3) which will be introduced in the following section.

2.5.2 Uniform Blur from Non-linear Motion

The blurry image is formed by the integral of light intensity over the exposure period. For more complex motion, the autocorrelation image $\mathcal{A}(|P(\mathbf{B})|)$ will show more bright points representing high correlation values (see Fig. 2.1 (c) and Fig. 2.4 (c) for examples).

In general, in the case of uniform (spatially-invariant) blur, one may write $\mathbf{B} = \mathbf{k} \otimes \mathbf{L}$, so, allowing for the possibility of noise, the deblurring problem (with known kernel) may be formulated as finding $\operatorname{argmin}_{\mathbf{L}} \|\mathbf{k} \otimes \mathbf{L} - \mathbf{B}\|_2^2$. In most cases, however, blurring acts as a form of low-pass filter – high-frequency information is lost. Consequently, this problem is not well-conditioned. Thinking of convolution with known $\mathbf{k}$ as being a linear operator, there exist near-zero eigenvalues whose eigenvectors correspond to high-frequency components of the signal (image). The deblurring process is to restore the lost frequency components of the image. If high-frequency components are

over-emphasized in the deblurring process, the resulting latent image $\mathbf{L}$ will be noisy, or edges will show ringing. A common solution to this is to add a regularization term that discourages excessive high-frequency components. One is therefore led to the following minimization problem.

$$\min_{\mathbf{L}} \|\mathbf{k} \otimes \mathbf{L} - \mathbf{B}\|_2^2 + \mu_2 \, h(\nabla \mathbf{L}) \, , \tag{2.3}$$

where $h(\cdot)$ is a penalty term used to discourage excessive gradients, which are indicative of noise and over-emphasized edges.

In the case of non-linear motion, the kernel is not known exactly, but an initial value of $\mathbf{k}$ may be estimated directly from the autocorrelation of the absolute *phase-only image* as described previously. Our final goal is to further refine the kernel $\mathbf{k}$ and estimate the latent sharp image $\mathbf{L}$ by solving

$$\min_{\mathbf{L},\mathbf{k}} \|\mathbf{k} \otimes \mathbf{L} - \mathbf{B}\|_2^2 + \mu_1 \, \|\mathbf{k}\|_2^2 + \mu_2 \, h(\nabla \mathbf{L}) \, , \tag{2.4}$$

where μ_1 and μ_2 are weight parameters. The first term encodes the fact that the modelled blurry image should be similar to the observed image. The second term is to regularize the solution of the blur kernel. The third term prevents over-sharpening.

The optimization of our energy function defined in Eq. (2.4) involves two sets of variables, the kernel and the latent image. We perform the minimization iteratively starting with the initial estimate of $\mathbf{k}$ given by the phase-only technique. (See Fig. 2.5 for an example).

2.5.2.1 Estimating the Latent Image

The goal is to minimize Eq. (2.4) by alternation. If $\mathbf{k}$ is known, the problem comes down to minimizing Eq. (2.3).

Specifically, we use a truncated-quadratic gradient regularization term

$$h(\nabla \mathbf{L}) = \sum_{x,y} \min(\|\nabla_{xy} \mathbf{L}/\epsilon\|^2, 1) \tag{2.5}$$

where $\epsilon \in [0.1, 1]$ and $\nabla_{xy}\mathbf{L}$ represents the gradient of $\mathbf{L}$ at image coordinates (x, y). This regularization term smooths out small noise, while allowing occasional large gradients (intensity differences). This type of term, proposed by Blake and Zisserman [1987] is widely used to regularize noise and gradients in stereo Veksler [2001] and was also used in deblurring in Xu et al. [2013]). Because the truncated quadratic is non-convex, the optimization problem is non-convex. We use the method of half quadratic splitting, as in Xu et al. [2011], to minimize this cost function, though other methods such as Iterative Reweighted Least Squares could be used for such truncated-quadratic cost Aftab and Hartley [2015].

2.5.2.2 Refining the Kernel

Now, with $\mathbf{L}$ known, the motion blur kernel can be refined by solving

$$\min_{\mathbf{k}} \|\mathbf{k} \otimes \mathbf{L} - \mathbf{B}\|_2^2 + \mu_1 \|\mathbf{k}\|_2^2 \ .$$

This is a quadratic problem, and can be solved directly by taking gradients, which results in a set of linear equations. More efficiently, we solve it in the Fourier domain, in which case there is a closed-form solution

$$\mathcal{F}(\mathbf{k}) = \overline{\mathcal{F}(\mathbf{L})} \odot \mathcal{F}(\mathbf{B}) \ / \ \left(\overline{\mathcal{F}(\mathbf{L})} \odot \mathcal{F}(\mathbf{L}) + \mu_1\right) \ ,$$

where the division is carried out point-wise (as are the multiplications). Then $\mathbf{k}$ is found by the inverse transform, and then normalized to sum to 1.

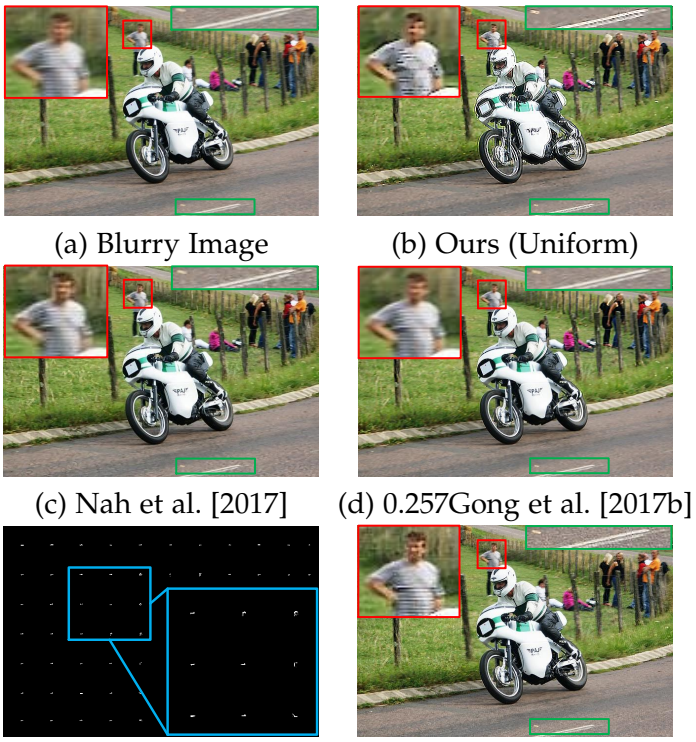
The algorithm alternates between recomputing $\mathbf{L}$ and $\mathbf{k}$ until convergence, or for a fixed number of steps.

2.6 Extension to Non-uniform Deblurring

Our method can be easily extended to handle non-uniform blur (e.g., the background and foreground undergo different blur) by deblurring the image patch-by-patch or layer-by-layer. Each patch or layer of the image corresponds to a different blur kernel. The new non-uniform blur model can be expressed as

$$\mathbf{B} = \sum_{i=1}^{N} \mathbf{k}_i \otimes \mathbf{l}_i, \tag{2.6}$$

where N denotes the number of segmented patches or layers, $\mathbf{l}_i = \mathbf{M}_i \odot \mathbf{L}$ is to extract the i-th patch or layer of the latent image, $\mathbf{M}_i$ is a binary mask with non-zeros values in the region corresponding to the i-th patch or layer in $\mathbf{L}$, and $\mathbf{k}_i$ denotes the blur kernel corresponding to the i-th patch. Similary, we define $\mathbf{B}_i = \mathbf{k}_i \otimes \mathbf{l}_i$ and $\mathbf{B} = \sum_{i=1}^{N} \mathbf{B}_i$. Each layer can be handled using our proposed uniform deblurring approach in Section 2.5. The final latent image $\mathbf{L}$ is $\sum_{i=1}^{N} \mathbf{l}_i$. In Fig. 2.6, we give an example of the deblurring results for uniform and non-uniform blur models. The image is a real blurry image from dataset Gong et al. [2017b]. Clearly, our non-uniform deblurring achieves better results than our uniform-deblurring model and the other existing non-uniform deblurring methods which either use additional depth, camera pose information Hu et al. [2014]; Gupta et al. [2010]; Whyte et al. [2012] or use deep convolutional neural networks Gong et al. [2017b]; Nah et al. [2017].



(a) Blurry Image (b) Ours (Uniform)

(c) Nah et al. [2017] (d) 0.257Gong et al. [2017b]

(e) Blur Kernel (f) Ours (Non-uniform)

Figure 2.6: *Example of our non-uniform blur kernel where the real blurry image is from Gong et al. [2017b]. (a) Input blurry image. (b) Our deblurring results by using uniform blur model and its blur kernel. We can see clearly that the man in a plaid shirt seems not deblurred because of the improper kernel. (c) Deblurring result of Nah et al. [2017]. (d) Deblurring result of Gong et al. [2017b]. (e) Non-uniform blur kernel. (f) Our deblurring result by using non-uniform blur model and kernel.*

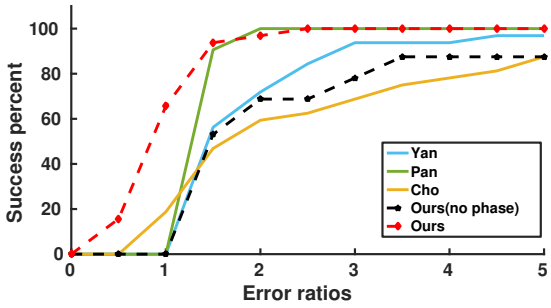


Figure 2.7: *Quantitative evaluations on dataset Levin et al. [2009]. We report the experimental results with and without using the blur kernel estimated from the phase-only image ('*Ours(no phase)*'). The results further demonstrate the effectiveness of blur kernel estimation from the phase-only image.*

Table 2.1: *Quantitave comparison on the dataset Levin et al. [2009].*

	0.257Cho et al. [2011]	Pan et al. [2016b]	Yan et al. [2017a]	Our (no phase)	Our
PSNR(dB)	25.63	27.54	24.70	25.74	**28.38**
SSIM	0.7907	0.8626	0.8760	0.7842	**0.9250**
SSD	2.6688	1.2747	1.6802	3.2517	**0.8776**

(a) Blurry Image (b) Yan et al. [2017a] (c) Pan et al. [2016b] (d) Ours

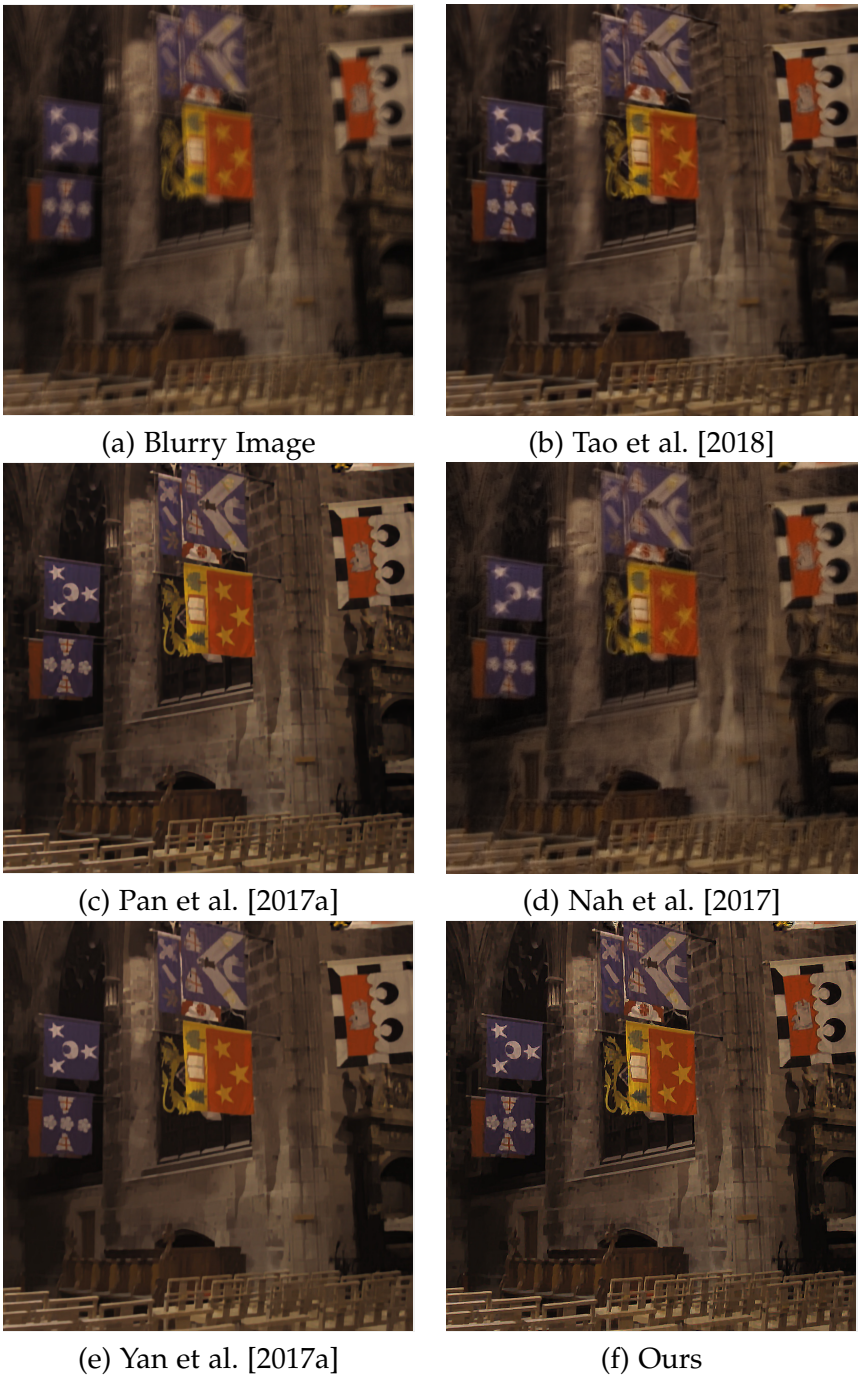
Figure 2.8: *Qualitative comparison on example images from dataset Köhler et al. [2012](top), Levin et al. [2009](bottom) and image taken by ourselves (middle). (a) Input blurry images. (b) Deblurring results of Yan et al. [2017a]. (c) Deblurring results of Pan et al. [2016b]. (d) Our deblurring result. (Best viewed on screen).*

2.7 Experiment

2.7.1 Experimental Setup

Dataset. We evaluate our approach on the datasets provided by Köhler et al. [2012]; Pan et al. [2016b]; Sturm et al. [2012]; Gong et al. [2017b]; Levin et al. [2009] and images captured by ourselves, which covers images from man-made scene, natural scene and images containing text (see Fig. 2.5, 2.6, 2.8 for examples).

Baselines and evaluation metric. Since our proposed approach can handle both uniform and non-uniform blurs, we compare with state-of-the-art methods for both cases separately. For traditional methods (non-deep learning methods), we compare with Yan et al. [2017a]; Pan et al. [2016b]; Cho and

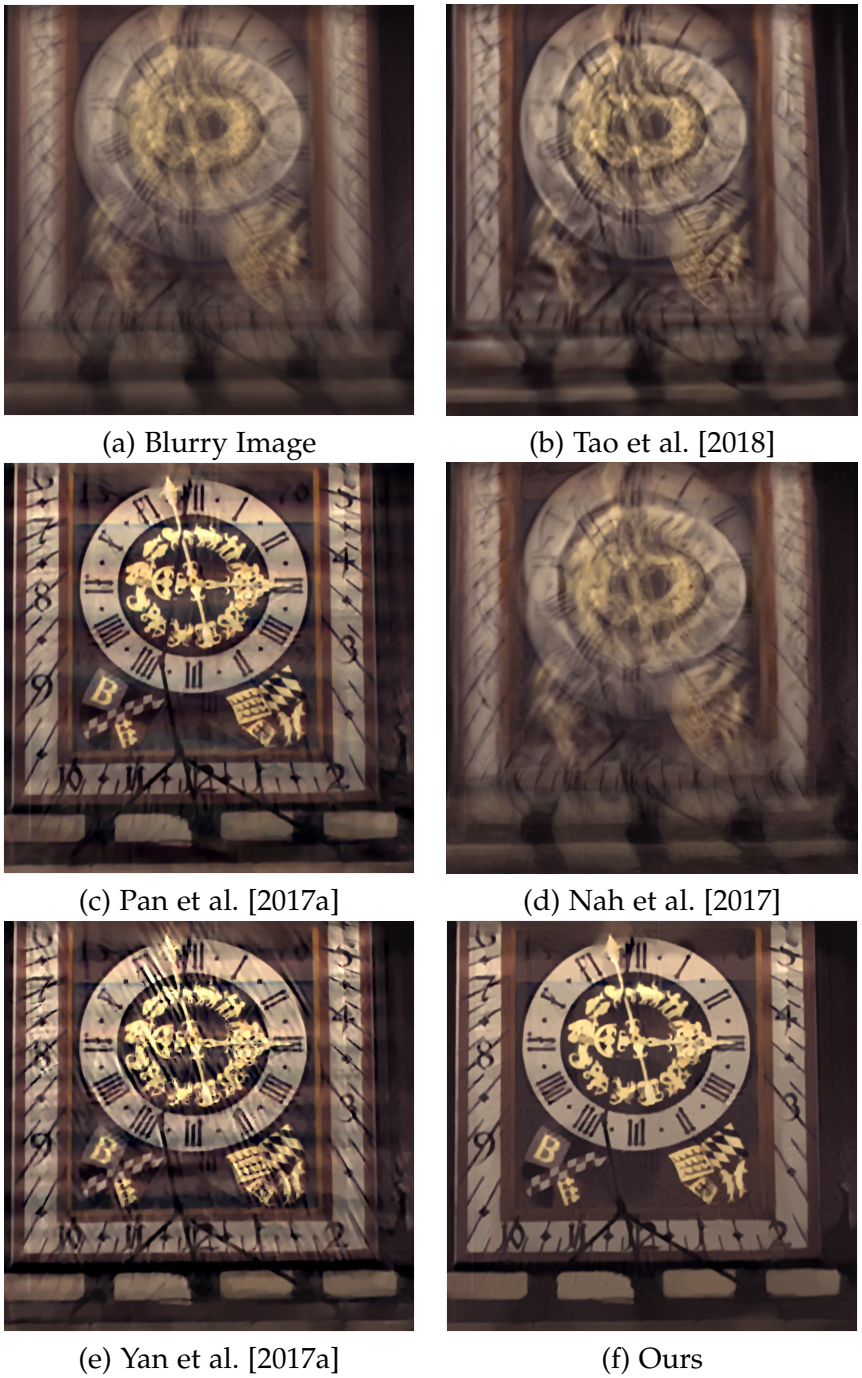


(a) Blurry Image (b) Tao et al. [2018]

(c) Pan et al. [2017a] (d) Nah et al. [2017]

(e) Yan et al. [2017a] (f) Ours

Figure 2.9: *Example of deblurring result on Köhler et al. [2012] dataset with kernel estimated by our method. (a) Input blurry images. (b) Deblurring results of Tao et al. [2018]. (c) Deblurring results of Pan et al. [2017a]. (d) Deblurring results of Nah et al. [2017]. (e) Deblurring results of Yan et al. [2017a]. (f) Our deblurring result. (Best viewed on screen).*

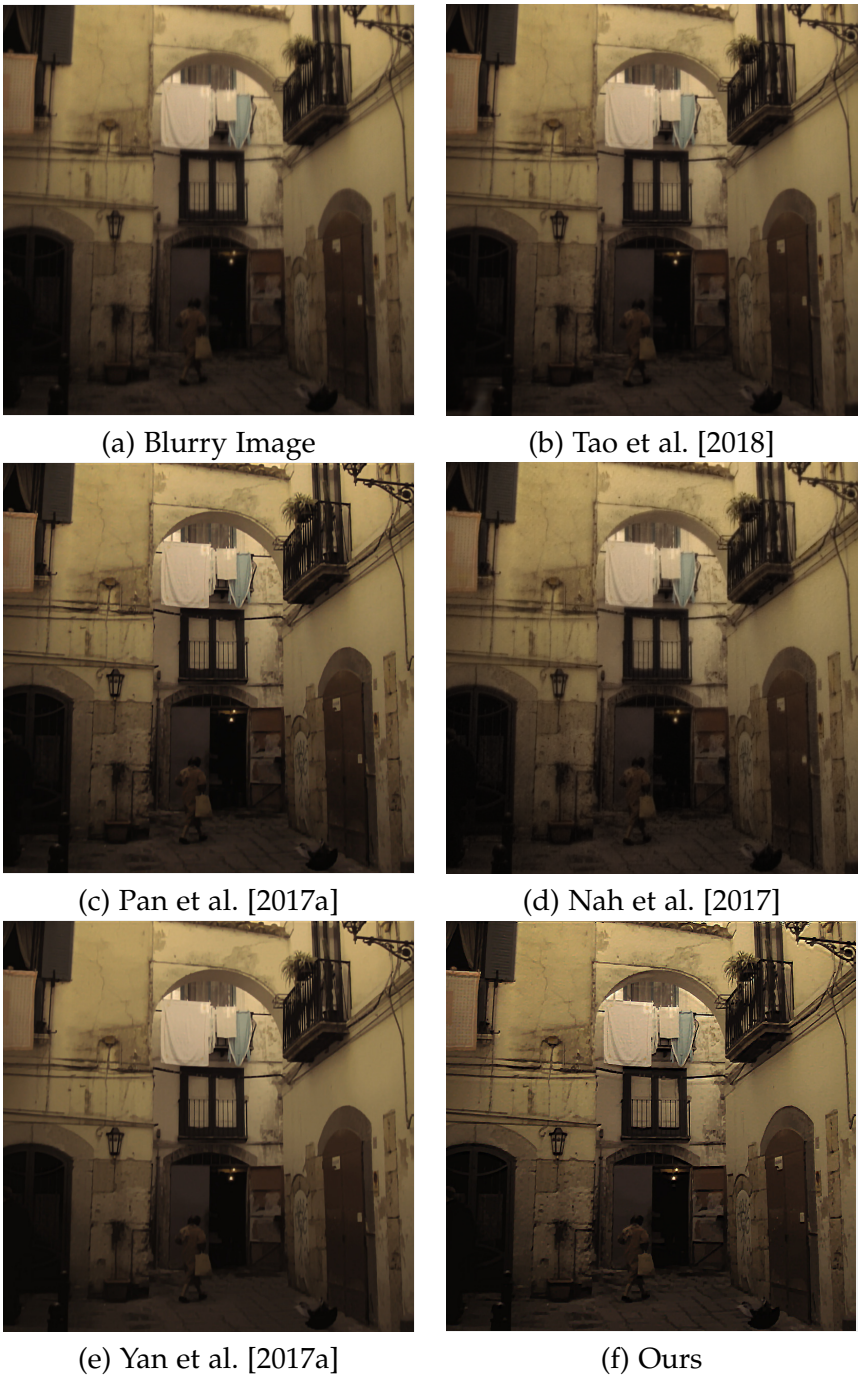


(a) Blurry Image (b) Tao et al. [2018]

(c) Pan et al. [2017a] (d) Nah et al. [2017]

(e) Yan et al. [2017a] (f) Ours

Figure 2.10: *Example of deblurring result on Köhler et al. [2012] dataset with kernel estimated by our method. (a) Input blurry images. (b) Deblurring results of Tao et al. [2018]. (c) Deblurring results of Pan et al. [2017a]. (d) Deblurring results of Nah et al. [2017]. (e) Deblurring results of Yan et al. [2017a]. (f) Our deblurring result. (Best viewed on screen).*

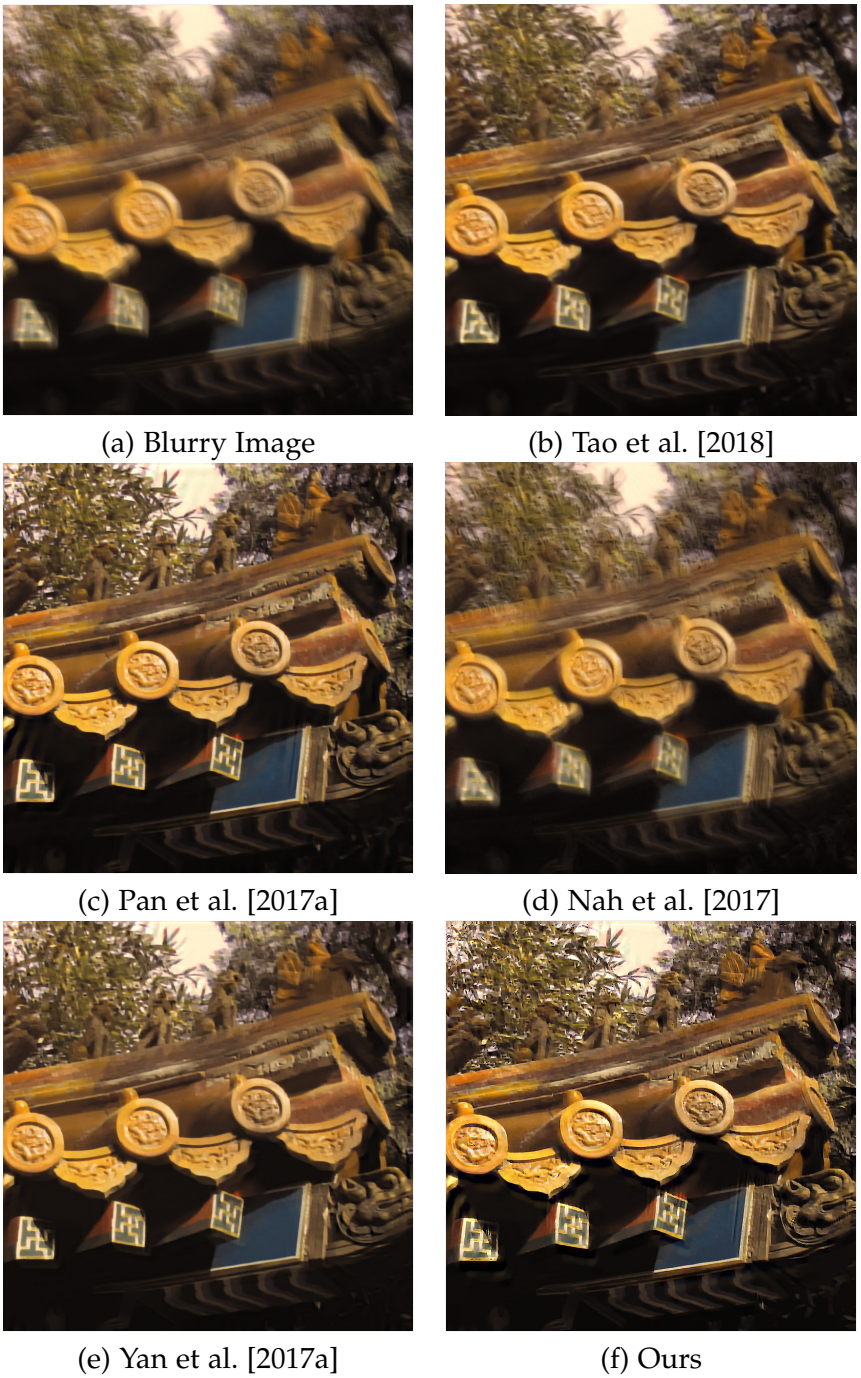


(a) Blurry Image (b) Tao et al. [2018]

(c) Pan et al. [2017a] (d) Nah et al. [2017]

(e) Yan et al. [2017a] (f) Ours

Figure 2.11: *Example of deblurring result on Köhler et al. [2012] dataset with kernel estimated by our method. (a) Input blurry images. (b) Deblurring results of Tao et al. [2018]. (c) Deblurring results of Pan et al. [2017a]. (d) Deblurring results of Nah et al. [2017]. (e) Deblurring results of Yan et al. [2017a]. (f) Our deblurring result. (Best viewed on screen).*



(a) Blurry Image (b) Tao et al. [2018]

(c) Pan et al. [2017a] (d) Nah et al. [2017]

(e) Yan et al. [2017a] (f) Ours

Figure 2.12: *Example of deblurring result on Köhler et al. [2012] dataset with kernel estimated by our method. (a) Input blurry images. (b) Deblurring results of Tao et al. [2018]. (c) Deblurring results of Pan et al. [2017a]. (d) Deblurring results of Nah et al. [2017]. (e) Deblurring results of Yan et al. [2017a]. (f) Our deblurring result. (Best viewed on screen).*

Table 2.2: *Quantitative comparisons on the dataset Köhler et al. [2012], where Nah et al. [2017]; Kupyn et al. [2018a] are deep based methods.*

	Blurry Image	Whyte et al. [2012]	Xu et al. [2013]	Pan et al. [2016b]
PSNR(dB)	24.93	27.03	27.47	29.95
SSIM	0.783	0.809	0.811	0.932
	Yan et al. [2017a]	Nah et al. [2017]	Kupyn et al. [2018a]	Ours
PSNR(dB)	28.42	26.48	26.10	**30.18**
SSIM	0.897	0.807	0.816	**0.933**

Lee [2009]; Whyte et al. [2012]; Xu et al. [2013]. For deep learning based methods, we compare with Gong et al. [2017b]; Nah et al. [2017]; Kupyn et al. [2018a] which can handle spatially-variant blur. We report the PSNR, SSIM on datasets Levin et al. [2009]; Köhler et al. [2012] and *error ratio*[3] on dataset Levin et al. [2009] which provides the ground truth blur kernels for evaluation.

Implementation details. We validate the parameters in our model on three reserved images for each dataset and use coarse-to-fine strategy for deblurring. We set $\mu_1 = 2$, $\mu_2 = 0.005$ for our experiment. Our framework is implemented using MATLAB®. It takes around 40 second to process one image (800×800) on a single i7 core running at 3.6 GHz.

2.7.2 Experimental Results

The dataset introduced in Levin et al. [2009] is a widely used uniform blur dataset, which contains 32 blurry images generated by 4 ground truth images and 8 blur kernels. We perform the quantitative and qualitative evaluation on this dataset. Results are shown in Fig. 2.7, 2.8 and Table 2.1, which demonstrates that our proposed approach achieves competitive results.

The *Natural dataset* is generated by Köhler et al. [2012] with camera motion measured and controlled by a Vicon tracking system. Specifically, the dataset provides blurry image, its latent image, and ground truth blur kernel, which allows the quantitative comparison of our approach with baselines. The captured images are of size 800×800. In Table 2.2, we show the quantitative comparison with the state-of-the-art Single-image deblurring approaches on dataset Köhler et al. [2012]. More results are shown in Fig. 2.9, Fig. 2.10, Fig. 2.11, and Fig. 2.12. It demonstrates that our approach can achieve the best performance on the PSNR and SSIM score.

We further show the corresponding qualitative comparison results on example images in Köhler et al. [2012] in Fig. 2.8. It clearly shows that our approach can recover more sharp details and with less ringing artifacts than

[3] *Error ratio* is introduced in Levin et al. [2009] which measures the ratio between the SSD (Sum of Squared Distance) of the deconvolution error computed with the estimated kernel and the ground truth kernel.

other approaches, which are highlighted in the presented results. We also report our deblurring result in Fig. 2.1, 2.4, 2.5 and 2.6, respectively. Note that our deblurring results can recover the color more faithfully than the baselines.

2.8 Conclusions

Our proposed *phase-only image* based kernel estimation approach is simple (implemented in a few lines of code). The resulted image deblurring algorithm achieves better quantitative results (using PSNR, SSIM, and SSD), than the state-of-the-art methods by extensive evaluation on the benchmark datasets. While our approach can handle the general blur cases, it still suffers from low lighting condition like other deblurring methods. Our future work will explore how to remove blurs less sensitive to lighting conditions.

Chapter 3

Single Image Deblurring and Camera Motion Estimation with Depth Map

Camera shake during exposure is a major problem in hand-held photography. This chapter focuses on estimating and removing the spatially-varying motion blur caused by camera shake during the exposure time. It proposes to achieve blind image deblurring by explicitly exploiting the 6 DoF (degrees-of-freedom) camera motion. In our formulation, the observed blurred image is formed by a composition of both the 6 DoF camera motion and the 3D scene structure, enabling us to capture the real blurred image generation process, especially due to camera shake.

Liyuan Pan, Yuchao Dai, Miaomiao Liu, Fatih Porikli. Joint Deblurring and Camera Motion Estimation from a Single Blurry Image. Winter Conference on Applications of Computer Vision (WACV), 2019.

3.1 Abstract

Camera shake during exposure is a major problem in hand-held photography, as it causes image blur that destroys details in the captured images. In the real world, such blur is mainly caused by both the camera motion and the complex scene structure. While considerable existing approaches have been proposed based on various assumptions regarding the scene structure or the camera motion, few existing methods could handle the real 6 DoF camera motion. In this chapter, we propose to jointly estimate the 6 DoF camera motion and remove the non-uniform blur caused by camera motion by exploiting their underlying geometric relationships, with a single blurred image and its depth map (either direct depth measurements, or a learned depth map) as input. We formulate our joint deblurring and 6 DoF camera motion estimation as an energy minimization problem which is solved in an alternative manner. Our

(a) Blurred Image (b) Our Result (c) Blurred Image (d) Our Result

Figure 3.1: *(a), (c) are the input blurred images from Köhler et al. [2012] dataset. (b), (d) are our deblurring results. We first use the blurred image to learn a depth map by using Godard et al. [2017]. Then, we jointly estimate camera motion and deblur the image with the learned depth map. With the depth map and the 6 Dof camera pose, we can project the recovered image to a sharp image sequence. We display one image of our deblurring sequence (during the exposure time). (Best view in Adobe Reader)*

model enables the recovery of the 6 DoF camera motion and the latent clean image, which could also achieve the goal of generating a sharp sequence from a single blurred image. Experiments on challenging real-world and synthetic datasets demonstrate that image blur from camera shake can be well addressed within our proposed framework.

3.2 Introduction

Image blurs are mainly caused by camera motions or motion of the objects in the scene during the long exposure time, which is generally required under the low-light condition. It is a common problem for hand-held photography and becomes increasingly important due to the popularity of mobile devices such as smartphones in recent years. Blind image deblurring targets at recovering the latent clean images from the blur ones. It has been an active research field in computer vision and image processing community Kim and Lee [2015]; Sellent et al. [2016]; Gong et al. [2017b]; Pan et al. [2017b]; Su et al. [2017].

Blind image deblurring is a very challenging task since it is highly under-constrained as multiple pairs of blur kernels and latent images can generate the same blurred image. A single blur kernel cannot model the complex blurs in real-world scenarios. Existing methods have exploited various constraints to model the characteristics of blur and use different natural image priors to regularize the solution space Lai et al. [2016]. However, these assumptions, such as uniform blur Xu et al. [2013], non-uniform blur from multiple homography Hu et al. [2014]; Pan et al. [2016a], with moving objects Pan et al. [2016a], constant depth Gupta et al. [2010]; Xu and Jia [2012], in-plane rota-

tion Sun et al. [2015], forward motion Zheng et al. [2013] may not be satisfied and applicable in practice.

In this chapter, we focus on estimating and removing the spatially-varying motion blur caused by camera shake during the exposure time and propose to achieve blind image deblurring by explicitly exploiting the 6 DoF (degrees-of-freedom) camera motion (see Fig. 3.1 for an example). In our formulation, the observed blurred image is formed by a composition of both the 6 DoF camera motion and the 3D scene structure, which enables us to capture the real blurred image generation process especially due to camera shake.

In order to handle the real world spatially-variant blur, we make the following assumptions regarding the scene structure and the camera motion:

1) **Availability of depth map of the scene.** As more and more consumer cameras are now equipped with depth sensors such as iPhone X, the availability of depth map becomes a rather reasonable and realistic assumption. Furthermore, the advent of deep learning also enables the estimation of a dense depth map from a single color image (monocular depth estimation) Ranftl et al. [2016]; Liwicki et al. [2016]; Li et al. [2018a].

2) **Small camera motion.** Due to the short exposure time and the high sampling rate of modern video cameras, the *camera shake process* can be modeled as the camera essentially undergoes a motion with small rotation angle and linear translation. We thus adopt the small angle approximation of rotation Yu and Gallup [2014].

The above assumptions naturally lead to a few legitimate queries:

1) **Why the 6 DoF camera motion is needed?** Recently, several deep learning based approaches Jin et al. [2018]; Purohit et al. [2019] could restore a video from a single blur image. However, the restored video sequence is not guaranteed to respect the 3D geometry of the scene as well as the camera motion. Instead, we target at recovering the 6 DoF camera motion which allows the recovery of a sharp video sequence from a single blurred image as well as the capacity of novel view synthesis for high frame rate video sequences. In Fig. 3.1, we illustrate the recovered video sequence from a single blurred image, which clearly demonstrates the benefit of our camera motion model.

2) **Why the small camera motion model is useful?** For small rotation model, the simplified rotation matrix is robust to noise as the second-order Taylor expansion of the rotation matrix has been ignored. The small motion model has been proven to be the key in estimating the camera poses and the 3D structure in the context of 3D reconstruction from accidental motion Im et al. [2015]. More complex camera trajectories could be exploited with the cost of increasing computational complexity.

Building upon the above assumptions regarding the camera motion and the scene structure, we formulate blind image deblurring as the task of joint latent clean image recovery and 6 DoF camera motion estimation [1]. Our unified framework naturally relates camera motion estimation and image deblurring, where the solution of one sub-task benefits the solution of the other sub-task. Specifically, we present an energy minimization based framework which involves both a unary term in explaining the observed blurred image and regularization terms on the camera motion and the desired latent clean image. To speed up the implementation and provide effective optimization, we apply a coarse-to-fine strategy to the energy minimization, where in each level we perform camera motion estimation and image deblurring in an alternative manner.

Our main contributions can be summarized as:

- We propose to jointly estimate the 6 DoF camera motion and deblur the image from a blurred image while giving its depth map (the depth information is from depth measurements or learned from the color image);
- We propose to use the small motion camera model which not only simplifies the motion estimation problem but also leads to an efficient solution;
- Extensive experiments on both synthetic and real images prove the effectiveness of our method especially its robustness against noisy depth maps.

3.3 Related Work

Recently, significant progress has been made in blind image deblurring. As there is a rich family of image deblurring methods, here we confine ourself to the most related ones. Blind image deblurring methods could be roughly categorized into two groups: monocular methods (image and video) and multi-view methods. Besides, we will also briefly cover deep learning based deblurring approaches.

Monocular image deblurring. Blind image deblurring is highly ill-posed, therefore various constraints on the blur kernels or the latent images have been proposed to regularize the solution space, which include the gradient based regularizers such as total variation Pan et al. [2017b], Gaussian scale

[1] The most similar work to ours seems to be Park and Lee Park and Lee [2017a], which solves for camera pose, scene depth, deblurring image and super-resolution under a unified framework from a **image sequence**. Different from Park and Lee [2017a], our method takes **a single blurred image and a depth map** as input to achieve camera motion estimation and image deblurring

mixture Fergus et al. [2006], $l_1 \backslash l_2$ norm Krishnan et al. [2011], and the l_0-norm regularizer Xu et al. [2013]. Besides, non-gradient-based priors such as the color line based prior Lai et al. [2015], and the extreme channel (dark-/bright channel) prior Pan et al. [2016b, 2017a]; Yan et al. [2017a] have also been explored. The fact that blur caused by camera shake in images are usually non-uniform motivates a series of work in modeling the spatially-variant blur. Whyte et al. [2012] approximated the blur kernels by discretization in the space of 3D camera rotations. Gupta et al. [2010] used a motion density function to represent the camera motion trajectory for the non-uniform deblurring, which requires the constant depth or fronto-parallel scene assumption. Hirsch et al. [2011] assumed that blur is locally invariant and proposed a fast non-uniform framework based on efficient filter flow. Zheng et al. [2013] considered only discretized 3D translations. Hu et al. [2014] proposed to jointly estimate the depth layering and remove non-uniform blur caused by in-plane motion from a single blurred image, which, however, requires user input for depth layers partition and known depth values a prior. Pan et al. [2016a] proposed to jointly estimate object segmentation and camera motion by incorporating soft segmentation, but requires user input. In practical settings, it is still challenging to remove strongly non-uniform motion blur in complex scenes.

Video deblurring. Single image based deblurring has been extended to video sequence to better remove blurs in dynamic scenesCho et al. [2012]; Kim and Lee [2014, 2015]; Pan et al. [2017b]. Wulff and Black Wulff and Black [2014] proposed a layered model to estimate both foreground motion and background motion. However, these motions are restricted to affine models, and it is difficult to be extended to multi-layer scenes due to the requirement of depth ordering of the layers. Kim and Lee [2015] proposed to simultaneously estimate optical flow and tackle the case of general blur by minimizing a single non-convex energy function. As depth can significantly simplify the deblurring problem, multi-view deblurring methods have been proposed to leverage the depth information. Xu and Jia [2012] inferred depth from two blurred images captured by a stereo camera and proposed a hierarchical estimation framework to remove motion blur caused by in-plane translation. Sellent et al. [2016] proposed a stereo video deblurring technique, where 3D scene flow is estimated from the blur images using a piecewise rigid scene representation. Pan et al. [2017b] proposed a single framework to jointly estimate the scene flow and deblur the images. Lee et al. [2018] proposed to estimate all blur model variables jointly, including latent sub-aperture image, camera motion, and scene depth from the blurred 4D light field.

Deep learning based image deblurring. Recently, the success of deep learning in high-level vision tasks have also been extended to low-level vision tasks such as image deblurring Vasu and Rajagopalan [2017]; Kim et al. [2017]; Su

et al. [2017]; Nah et al. [2017]; Tao et al. [2018]. Sun et al. [2015] proposed a convolutional neural network (CNN) to estimate locally linear blur kernels. Gong et al. [2017b] learned optical flow field from a single blurred image directly through a fully-convolutional deep neural network and recovered the clean image from the learned optical flow. Jin et al. [2018] extracted a video sequence from a single motion-blurred image by introducing loss functions invariant to the temporal order. Li et al. [2018b] used a learned image prior to distinguish whether an image is sharp or not and embedded the learned prior into the MAP framework. Tao et al. [2018] proposed a light and compact network, SRN-DeblurNet, to deblur the image. With the supervised learning nature of these deep learning based deblurring methods, the success strongly depends on the statistical consistency between the training datasets and the testing datasets, which could hinder the generalization ability for real world applications.

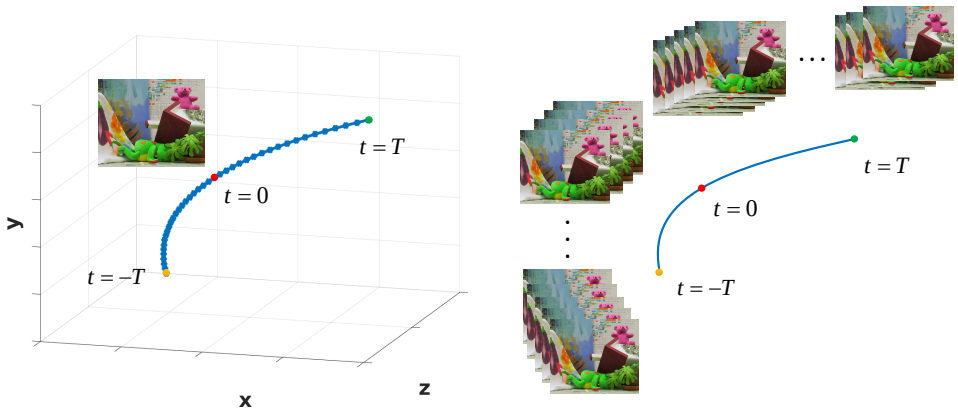


Figure 3.2: *Example of our blur model. We approximate the blurred image by averaging the images sequence during the exposure time* $2T$*, where the spatially-variant blur kernel induced by the 6 DoF camera motion. (Best viewed on screen).*

3.4 A Unified Spatially-varying Camera Shake Blur Model

In this section, we develop a unified spatially-varying camera shake blur model, which explicitly relates the 6 DoF camera motion (including in/out-of-plane rotation and translation), and the latent clean image. In particular, we formulate our problem as a joint estimation of 6 DoF camera motion and image deblurring for depth-varying scenery.

3.4.1 Blur Model

Given a single blurred image **B** and its corresponding depth map **D** (either from depth sensors or learned through a deep neural network), our goal is

to find a clean (latent) image $\mathbf{L}$ and its corresponding camera motion during image capture. The blurred image can be modeled as a convolution of the latent image with a spatially-varying blur kernel $\mathbf{k_x}$,

$$\mathbf{B}(\mathbf{x}) = (\mathbf{k_x} \otimes \mathbf{L})(\mathbf{x}) + z, \tag{3.1}$$

where $\mathbf{k_x}$ denotes the blur kernel at pixel location $\mathbf{x} \in \mathbb{R}^2$, $\otimes$ is the convolution operator, $z \sim \mathcal{N}(0, \sigma^2)$ is defined as the Gaussian noise. Note that this problem is highly under-determined since multiple pairs of $\mathbf{L}$ and $\mathbf{k_x}$ could lead to the same blurred image. We therefore make assumptions on the generation process of the blur image that, for complex dynamic settings such as outdoor traffic scenes, the spatially-varying blur kernels are determined by the 6 DoF camera motion and the scene structure.

The blurred image is generally modeled as the integration of the images during the exposure time $2T$. In our model, we will explicitly model the blurred image generation process with respect to the 6 DoF camera motion. Given the depth map $\mathbf{D}$ corresponding to the latent image $\mathbf{L}$ and the camera motion $\mathbf{p_t}$, the image at time t is defined as $w(\mathbf{p}_t, \mathbf{D}, \mathbf{L})$. $w(\cdot)$ is referred as the warping function which is defined by the back-projection of the latent image to 3D points based on the depth $\mathbf{D}$ followed by a forward projection to image frame at time t based on $\mathbf{p}_t$. The blurred image is therefore generated as

$$\mathbf{B} = \lambda_T \int_{t=-T}^{T} w(\mathbf{p}_t, \mathbf{D}, \mathbf{L}) dt + z, \tag{3.2}$$

where $\lambda_T = \frac{1}{2T}$. In general, we handle the problem in discrete space with

$$\mathbf{B} = \lambda_N \sum_{n=-N}^{N} w(\mathbf{p}_t, \mathbf{D}, \mathbf{L}) + z = \mathbf{A_p}(\mathbf{L}) + z,$$

where sample frequency $\lambda_N = \frac{1}{2N+1}$, N is the sample number, and n is the sample index.

3.4.2 Camera Motion Model

We further assume that the camera performs uniform out-of-plane rotation and translation. Let $\mathbf{p} = (\theta_x, \theta_y, \theta_z, v_x, v_y, v_z)^T$ represent the absolute motion during the exposure time $2T$. The camera motion at time t, is then defined as $\mathbf{p}_t = (t/2T) * \mathbf{p}$. Let $\grave{} = (\theta_x, \theta_y, \theta_z)^T$ be the rotation parameters (Rodrigues' rotation formula Belongie [1999]), and $\mathbf{v} = (v_x, v_y, v_z)^T$ be the translation vector. Since the camera exposure time is usually very short (several milliseconds), we assume that the camera performs small rotation motion, thus the rotation matrix can be approximated as

$$\mathbf{R} = \mathbf{I} + [\grave{}]_\times = \begin{bmatrix} 1 & -\theta_z & \theta_y \\ \theta_z & 1 & -\theta_x \\ -\theta_y & \theta_x & 1 \end{bmatrix},$$

where $[\cdot]_\times$ denotes the cross-product operator, and $\mathbf{I}$ is the identity matrix. The small rotation motion assumption results in a first-order approximation of the rotation matrix.

Based on the above blur model and small motion model, we define our energy functions for deblurring and camera motion estimation in the following sections.

3.4.3 Energy Formulation

Our energy function is defined on the latent clean image and the 6 DoF camera motion. We formulate our problem in a unified framework to jointly estimate the camera motion and deblur the image. Our energy function is defined as

$$E = E_{\text{blur}}(\mathbf{L}, \mathbf{p}) + E_{\text{reg}}(\mathbf{L}, \mathbf{p}), \tag{3.3}$$

which consists of a data term for deblurring, a regularization term enforcing the smoothness in camera motion, induced optical flow and the latent clean image. The energy function terms are further discussed in the following sections.

3.4.3.1 Data Term for Deblurring.

Our data term for deblurring involves two terms, which is defined as

$$E_{\text{blur}}(\mathbf{L}, \mathbf{p}) = \left\|\mathbf{A}_{\mathbf{p}}(\mathbf{L}) - \mathbf{B}\right\|_{\text{F}}^2 + \left\|\nabla\mathbf{A}_{\mathbf{p}}(\mathbf{L}) - \nabla\mathbf{B}\right\|_{\text{F}}^2. \tag{3.4}$$

The first term encodes the fact that the estimated blur image from spatially-varying blur kernel should be similar to the observed blurred image. The second term encourages the intensity changes (gradient) in the estimated blurred image should be close to that of the observed blurred image.

3.4.3.2 Regularization Terms.

Our regularization terms explore the small motion constraints on the camera motion model, spatial smoothness constraints on the latent image and optical flow induced by the camera motion. The first one is to avoid the trivial solution of $\mathbf{p} = \mathbf{0}$. The second one is to enforce the optical flow generated from the camera motion and the depth map to be smooth across the image and respect the image and depth discontinuities. The third term is to suppress the noise in the latent image and penalize the spatial fluctuations. To this end, our potential function is defined as

$$E_{\text{reg}}(\mathbf{p}, \mathbf{L}) = \mu_1 \left\|\mathbf{p}\right\|_2^2 + S(\mathbf{p}) + \mu_4 \left\|\nabla\mathbf{L}_{(i,j)}\right\|_1, \tag{3.5}$$

where $S(\mathbf{p})$ is defined as

$$S(\mathbf{p}) = \mathcal{E}(B, D) \left\| \nabla F(\mathbf{p})_{(i,j)} \right\|_2^2,$$
$$\mathcal{E}(B, D) = \sum_{i,j \in \Omega} \mu_2 \, e^{\left(-\frac{\left\| \nabla \mathbf{B}_{(i,j)} \right\|_2^2}{\sigma_{\mathrm{B}}^2} \right)} + \mu_3 \, e^{\left(-\frac{\left\| \nabla \mathbf{D}_{(i,j)} \right\|_2^2}{\sigma_{\mathrm{D}}^2} \right)}.$$

Ω denotes the image region. $\mu_{\{1,2,3,4\}}$ are weight parameters with $\mu_1 < 0$. $\sigma_{\{\mathrm{B,D}\}}$ are parameters for balancing the influence of the image and depth discontinuity on the spatial smoothness constraints. $F(\mathbf{p})$ denotes the optical flow field induced by camera motion $\mathbf{p}$ and depth map $\mathbf{D}$, which is obtained by forward projection of the 3D points corresponding to $\mathbf{t} = \mathbf{0}$ to the camera motion $\mathbf{p}$.

3.5 Solution

The optimization of our energy function defined in Eq. (3.3), is to solve two different sets of variables, which are the camera motion $\mathbf{p}$ and the latent image $\mathbf{L}$, respectively. In order to solve the variables more efficiently, we perform the optimization alternatively through the following steps,

- Fix the latent image $\mathbf{L}$, solve for the camera motion $\mathbf{p}$ by optimizing Eq. (3.6) (See Section 3.5.1).
- Fix the motion parameters $\mathbf{p}$, solve for the latent image $\mathbf{L}$ by optimizing Eq. (3.7) (See Section 3.5.2).

In the following sections, we describe the details for each optimization step.

3.5.1 Camera motion estimation

We fix the latent image, namely $\mathbf{L} = \tilde{\mathbf{L}}$, then Eq. (3.3) reduces to

$$\min_{\mathbf{p}} \left\| \mathbf{A}_{\mathbf{p}}(\tilde{\mathbf{L}}) - \mathbf{B} \right\|_{\mathrm{F}}^2 + \left\| \nabla \mathbf{A}_{\mathbf{p}}(\tilde{\mathbf{L}}) - \nabla \mathbf{B} \right\|_{\mathrm{F}}^2 + \mu_1 \left\| \mathbf{p} \right\|_2^2 + S(\mathbf{p}). \tag{3.6}$$

This is a non-linear and non-convex optimization problem. Fortunately, the solution space (6 DoF camera motion) is very small. We solve the problem by a nonlinear least-squares method Moré [1978] to find the solution.

3.5.2 Image deblurring

Given the 6 DoF camera motion parameters, namely $\tilde{\mathbf{p}}$, the blur image is derived based on Eq. (3.3). The objective function in Eq. (3.3) becomes convex with respect to $\mathbf{L}$ and is expressed as

$$\min_{\mathbf{L}} \left\| \mathbf{A}_{\tilde{\mathbf{p}}}(\mathbf{L}) - \mathbf{B} \right\|_{\mathrm{F}}^2 + \left\| \nabla \mathbf{A}_{\tilde{\mathbf{p}}}(\mathbf{L}) - \nabla \mathbf{B} \right\|_{\mathrm{F}}^2 + \mu_4 \left\| \mathbf{L} \right\|_{\mathrm{TV}}. \tag{3.7}$$

Table 3.1: *Comparison of flow error and deblurring results on different datasets (Middlebury, KITTI and TUM).*

		Pan et al. [2017a]	Yan et al. [2017a]	Kim	Our
PSNR (dB)	Middleburry	25.44	24.98	-	**26.16**
	KITTI	22.78	23.28	-	**26.21**
SSIM	Middleburry	0.7962	0.7822	-	**0.8357**
	KITTI	0.7615	0.7715	-	**0.8289**
Flow Error	TUM	-	-	31.95	**27.57**

In order to obtain the latent clean image $\mathbf{L}$, we adopt the conventional primal-dual optimization method Chambolle and Pock [2011] and derive the updating scheme as follows

$$\begin{cases} \mathbf{q}^{r+1} = \dfrac{\mathbf{q}^r + \gamma \nabla \mathbf{L}^r}{\mathbf{max}(1, |(\mathbf{q}^r + \gamma \nabla \mathbf{L}^r)|)}, \\ \mathbf{L}^{r+1} = \arg\min_{\mathbf{L}} \left\|\mathbf{A}_{\tilde{\mathbf{p}}}(\mathbf{L}) - \mathbf{B}\right\|^2 + \left\|\nabla \mathbf{A}_{\tilde{\mathbf{p}}}(\mathbf{L}) - \nabla \mathbf{B}\right\|^2 + \dfrac{\left\|\mathbf{L}^{r+1} - (\mathbf{L}^r - \eta(\mu_4 \nabla \mathbf{q}^{r+1})\right\|^2}{2\eta}, \end{cases} \tag{3.8}$$

where r is the iteration number, q^r denotes the dual variable, $\eta = 10$ and $\gamma = 0.005$ are update step parameters. More details are referred to Chambolle and Pock [2011].

To further speed up the alternative optimization, we propose to apply a coarse-to-fine strategy to the energy minimization. Specifically, we perform camera motion estimation and image deblurring in an alternative manner in each level. The results from the coarse levels can be used as initialization for the following fine levels.

3.6 Experiments

3.6.1 Experimental Setup

Synthetic Datasets. To the best of our knowledge, there are no realistic benchmark datasets that provide blurred images, their corresponding ground-truth depth maps, and the latent clean images. We thus make use of the KITTI Geiger et al. [2013] and Middlebury dataset Scharstein et al. [2014] to create synthetic datasets on realistic scenery. Since the camera shake always involves small rotation and translation, we thus sample the rotation angle for each image from a Gaussian distribution with the standard deviation $\sigma_a = 0.05$ rad and translation vector from a Gaussian distribution with $\sigma_t = 0.4$m for KITTI and $\sigma_t = 0.05$m for Middlebury. The difference in the standard deviation is to match the different depth range in two datasets, which is 3m for Middlebury and 40m for KITTI dataset, respectively.

The blurred image is generated by averaging the *captured clean images* at $N = 20$ uniformly distributed camera motion and locations within the expo-

(a) Input blurred Image (b) Pan et al. [2017a] (c) Yan et al. [2017a] (d) Ours

Figure 3.3: *Example deblurring results on the Middlebury dataset. (a) Input blurred color images. (b) Deblurring results of Pan et al. [2017a]. (c) Deblurring results of Yan et al. [2017a]. (d) Our deblurring results. (Best viewed on screen).*

sure time $T = 0.23$ (see Eq. (3.3) and Fig. 3.2 for details). In particular, the clean images are rendered based on the camera motion in 3D space. Note that the blurred image rendering process requires a dense depth map. Instead of filling in holes for the sparse raw depth map in KITTI, we adopt the unsupervised stereo matching approach Zhong et al. [2017], which ranks among the methods of top performance on KITTI dataset with a pre-trained model available, to estimate the dense disparity map referred to as oracle depth. We create our testing set using 200 images chosen from different image sequences in KITTI. We similarly generate the testing set with 14 images from Middlebury 2014 using the depth maps provided by the dataset.

Real Dataset. We further evaluate our method on the TUM RGB-D dataset Sturm et al. [2012], which includes both depth maps and real blurred images. The captured depth maps and color images are of size 640×480. The measurements from the depth sensors are imperfect, which are noisy and contaminated with large holes due to the reflective surfaces and distant objects in the scene. We thus pre-process the depth maps by filling in those holes using a traditional depth completion method Yang et al. [2014]. We test our algorithm on 300 images chosen from the 'bear' and 'walkman' sequences. Since the TUM dataset does not include ground truth sharp images, we thus only provide qualitative comparison with the state-of-the-art blind deblurring approaches.

Implementation Details. We validate the parameters in our model on three

reserved images for each dataset. We set $\mu_1 = -20$, $\mu_2 = \mu_3 = 0.2$, $\mu_4 = 0.05$, $\sigma_B = 0.01$, $\sigma_D = 0.02$ for all of our experiments. In order to give a better initialization for our method, we first apply a conventional blind de-convolution approach Krishnan et al. [2011] to estimate a uniform blur kernel of size 25×25 to provide a prior on our 6 DoF pose $\mathbf{p}$. Our experiments show that such initialization is more robust than initializing the algorithm randomly. We further implement our algorithm in the traditional coarse-to-fine manner to achieve fast convergence. In particular, the image pyramid is built with 11 levels and the scale factor is set as 0.9. The motion parameters and the latent image estimated from coarse resolution are propagated as initialization to the next pyramid level. Our framework is implemented using MATLAB® with C++ wrappers. It takes around 5 minutes to process one image on a single i7 core running at 3.6 GHz.

Baselines and Evaluation Metric. We compare our approach with the state-of-the-art blind deblurring methods, such as Yan et al. [2017a], Pan et al. [2017a] and Hu et al. [2014], which handle spatially variant blur from a single image. We further compare with a video method Kim and Lee [2015] and two learning based methods Gong et al. [2017b]; Nah et al. [2017]which can handle non-uniform blur on the TUM dataset.

We report the PSNR and SSIM on our deblurred images. Instead of directly evaluating the rotation and translation estimation, we report the optical flow errors which are introduced by the errors in the camera motion estimation. In particular, the error metric is computed by counting the number of pixels which have errors more than 3 pixels and 5% of its ground-truth.

3.6.2 Experimental Results

For the above datasets we used, the depth is from stereo matching, depth sensor and learned by neural network. In Table 3.1, we compare our approach with the state-of-the-art single image deblurring methods, Yan et al. [2017a] and Pan et al. [2017a], for spatially-variant blurs on Middlebury, KITTI, and TUM dataset, based on the PSNR, SSIM and Flow Error metric. Note that experiments on Middlebury and TUM used depth with high accuracy (provided by the dataset) as input for deblurring. In order to evaluate the robustness of our approach w.r.t. the depth quality, we adopted the most recent unsupervised monocular depth estimation method Godard et al. [2017] to learn the depth maps for KITTI dataset as input to remove the blurs generated based on oracle depth from stereo matching approach Zhong et al. [2017]. We further provide an example for visual comparison in Fig. 3.5(e),(f) to show the difference of the deblurring results from the oracle depth and the learned depth, respectively. Note that our approach outperforms all the baseline approaches which do not reason about the camera motion, by a large margin.

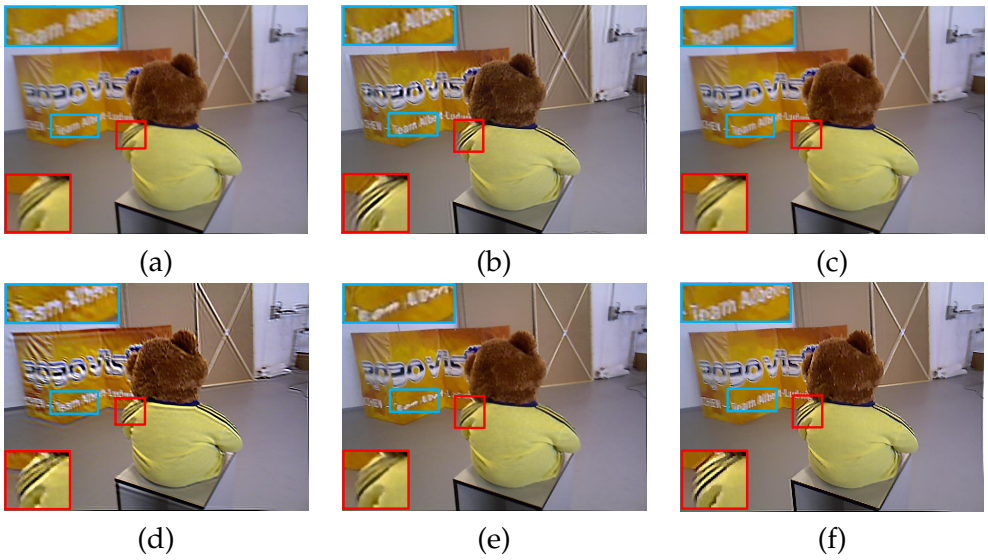


Figure 3.4: *Comparison with the state-of-the-art non-uniform deblurring methods using real blurred image from the TUM dataset. The depth is from the Kinect sensor. (a) Blurred image. (b) Video based deblurring result Kim and Lee [2015]. (c) Learning based result Gong et al. [2017b]. (d) Single image based deblurring result Hu et al. [2014], which also considers depth in their formulation. (e) Learning based result Nah et al. [2017]. (f) Our deblurring result.*

This evidences the importance of our joint camera motion estimation and image deblurring framework. We further compare our approach with the image deblurring approach from monocular video sequence Sturm et al. [2012]. This again shows the importance of including depth information and performing 6 DoF camera motion estimation for blind deblurring.

The qualitative comparisons on the three datasets are shown in Fig. 3.3, 3.5, and 3.4, respectively. The qualitative results show that our approach can recover more sharp details than other competing approaches, which are highlighted in the reported results. Note that our deblurring results can recover the color images more faithfully than the baselines. It further evidences the quantitative improvements as shown in Table 3.1. Last but not least, as we have recovered the 6 DoF camera motion, we can generate a sharp video sequence correspondingly as illustrated in Fig. 3.1, where each novel frame is generated by warping the latent clean image with the corresponding camera motion estimation and estimated/measured depth maps.

3.7 Conclusions

In this chapter, we have presented a joint optimization framework to estimate the 6 DoF camera motion and deblur the image from a single blurred image.

(a) Input blurred Image (b) Pan et al. [2017a]

(c) Yan et al. [2017a] (d) Ours

(e)Input with oracle depth from Zhong et al. [2017] **(f) Input with learned depth from Godard et al. [2017]**

Figure 3.5: *Example deblurring results on the KITTI dataset. (a) Input blurred color images. (b) Deblurring results of Pan et al. [2017a]. (c) Deblurring results of Yan et al. [2017a]. (d) Our deblurring results with learned depth map as input. In order to compare the results with respect to different input depth map,* **(e) and (f) show our deblurring results with oracle depth map and learned depth map as inputs, respectively.** *Compared with the two state-of-the-art deblurring methods, our method achieves the best performance (Best viewed on screen).*

To alleviate the difficulties, we exploit the availability of depth maps (either from noisy measurements or learned through a deep neural network) and a small motion model for the camera. Under our formulation, the solution of one sub-task benefits the solution of the other sub-task. Extensive experiments on both synthetic and real image datasets demonstrate the superiority of our framework over very recent state-of-the-art blind image deblurring methods such as dark channel prior Pan et al. [2017a] and extreme channel prior Yan et al. [2017a]). In the future, we plan to exploit more general parametric camera trajectories to further improve the performance in real world challenging scenarios.

Chapter 4

Joint Stereo Video Deblurring, Scene Flow Estimation and Moving Object Segmentation

In this chapter, to tackle non-uniform deblurring difficulty, we leverage the stereo video frames to restore the latent sharp images. With the stereo video frames, depth and the optical flow can be estimated, which are useful for the spatially variant blur kernel estimation. We also use the motion boundary information provided by semantic segmentation as prior.

Liyuan Pan, Yuchao Dai, Miaomiao Liu, Fatih Porikli, Quan Pan. Joint Stereo Video Deblurring, Scene Flow Estimation and Moving Object Segmentation. Transactions on Image Processing (TIP), 2019.

4.1 Abstract

Stereo videos for the dynamic scenes often show unpleasant blurred effects due to the camera motion and the multiple moving objects with large depth variations. Given consecutive blurred stereo video frames, we aim to recover the latent clean images, estimate the 3D scene flow and segment the multiple moving objects. These three tasks have been previously addressed separately, which fail to exploit the internal connections among these tasks and cannot achieve optimality. This chapter proposes to jointly solve these three tasks in a unified framework by exploiting their intrinsic connections. To this end, we represent the dynamic scenes with the piece-wise planar model, which exploits the local structure of the scene and expresses various dynamic scenes. These three tasks are naturally connected under our model and expressed as the parameter estimation of 3D scene structure and camera motion (structure and motion for the dynamic scenes). By exploiting the blur model constraint, the moving objects and the 3D scene structure, we reach an energy minimization formulation for joint deblurring, scene flow and segmentation. We evalu-

ate our approach extensively on both synthetic datasets and publicly available real datasets with fast-moving objects, camera motion, uncontrolled lighting conditions and shadows. Experimental results demonstrate that our method can achieve significant improvement in stereo video deblurring, scene flow estimation and moving object segmentation, over state-of-the-art methods.

4.2 Introduction

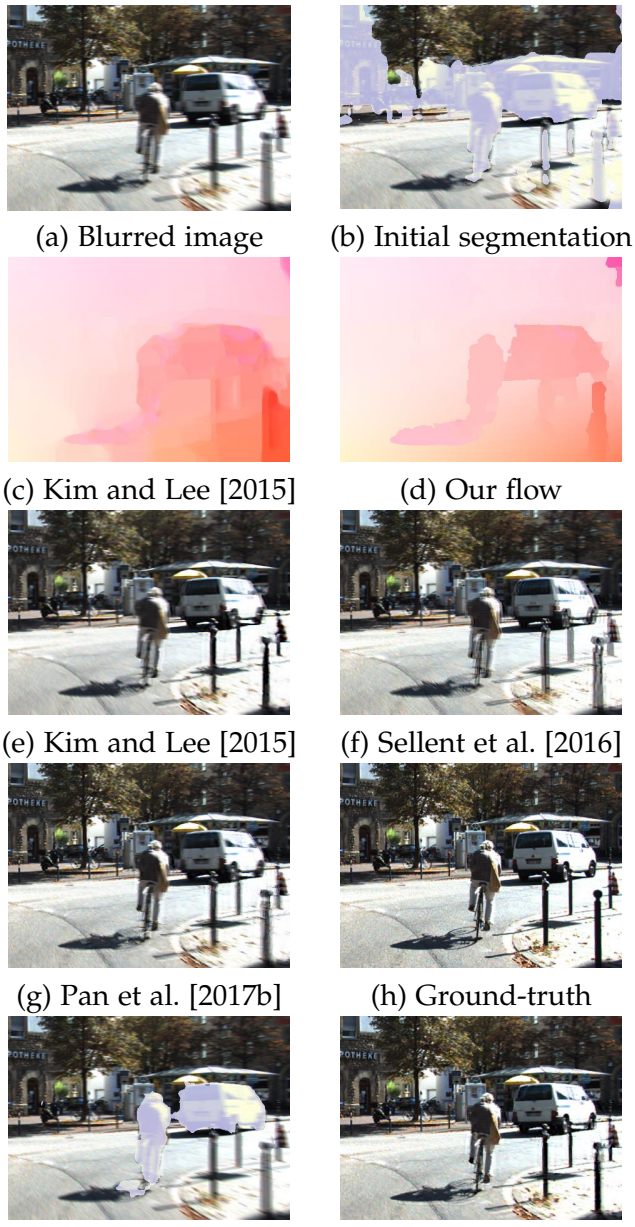
Image deblurring aims at recovering the latent clean image from single or multiple blurred images, which is a classic and fundamental task in image processing and computer vision. Image blur could be caused by various reasons, for example, optical aberration, medium perturbation, defocus, and motion Schuler et al. [2012]; Shi et al. [2015]; Gupta et al. [2010]; Jia [2014]; Sun et al. [2015]. In this work, we only focus on motion blur, which is widely encountered in real-world applications such as autonomous driving Franke and Joos [2000]; Geiger et al. [2012]; Liu et al. [2017c]. The effects become more apparent when the exposure time increased due to low-light conditions.

Motion deblurring has been extensively studied, and various methods have been proposed in the literature. It is common to model the blur effect using kernels Jia [2014]; Seok Lee and Mu Lee [2013]. Early deblurring methods mainly focus on the blur caused by camera shake with constant depth or static scenes with moving objects Hu et al. [2014]; Xu et al. [2013]. We focus on a more generalized motion blur caused by both camera motion and moving objects in this work. Therefore, conventional blur removal methods, such as Gupta et al. [2010]; Krishnan et al. [2011], cannot be directly applied since they are restricted to a single or a fixed number of blur kernels, making them inferior in tackling general motion blur problems.

For a scenario where both camera motion and multiple moving objects exist, the blur kernel is, in principle, defined for each pixel individually. Recently, several researchers have studied to handle the blurred images with *spatially-variant blur* Kim and Lee [2015]; Sellent et al. [2016]; Pan et al. [2017b] which uses accurate motion estimation to model the blur kernel. The phenomenon around motion and blur can be viewed as a chicken-egg problem: effective motion blur removal requires accurate motion estimation. Yet, the accuracy of motion estimation highly depends on the quality of the images.

It is a problem for any of the algorithms exploiting motion information as the condition is a major challenge to reliable flow computation.

In this chapter, we aim to tackle a 'generalized stereo deblurring' problem. The moving stereo cameras observe a dynamic scene with varying depth, and the moving objects' boundaries are mixed with the background pixels. Thus we propose to use the motion boundary information provided by semantic segmentation Wu et al. [2019]. In our approach, we jointly estimate



(a) Blurred image (b) Initial segmentation

(c) Kim and Lee [2015] (d) Our flow

(e) Kim and Lee [2015] (f) Sellent et al. [2016]

(g) Pan et al. [2017b] (h) Ground-truth

(i) Our segmentation (j) Our deblurred result

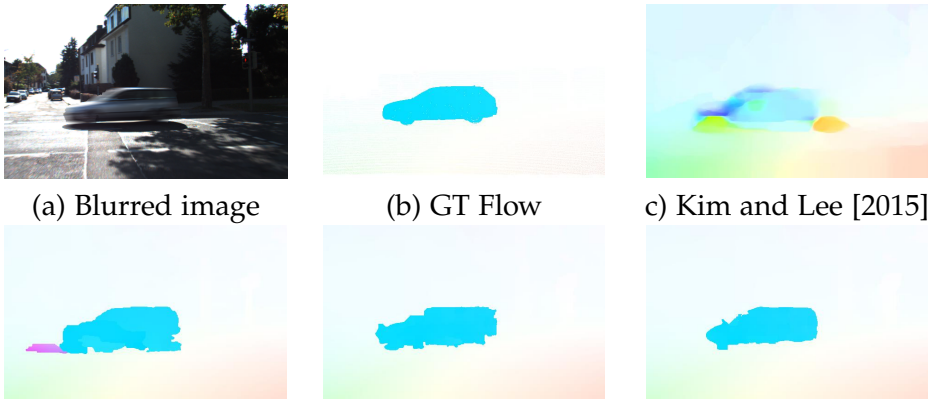
Figure 4.1: *Stereo deblurring, scene flow estimation and moving object segmentation results with (a) and (b) as input. (a) Blurred image. (b) Initial segmentation prior. (c) Flow estimation by Kim and Lee [2015]. (d) Our flow estimation result. (e) Deblurring results by Kim and Lee [2015]. (f) Stereo deblurring results by Sellent et al. [2016] which uses Vogel et al. [2015] to estimate scene flow. (g) Deblurring results by Pan et al. [2017b]. (h) Ground-truth latent image. (i) Our moving object segmentation result. (j) Our stereo deblurring result. Best viewed in colour on the screen.*

scene flow, segment the moving objects and deblur the images under a unified framework. Using our formulation, we attain significant improvement in numerous real challenging scenes, as illustrated in Fig. 4.1.

We would like to argue that, the scene flow estimation approaches that make use of colour brightness constancy may be hindered by the blurred images. Existing optical flow methods make generic, spatially homogeneous, assumptions about the spatial structure of the flow. Due to the inherent correlation between semantic segmentation and moving object segmentation (for example, the movement of pixels a vehicle tends to be the same and be different from the background), semantic segmentation has been used to provide motion segmentation prior. Thus, we investigate the benefits of semantic grouping Wu et al. [2019] which are more beneficial for the scene flow estimation task. Here, we only need a coarse and simple semantic segmentation prior to distinguish foreground and background. The more of the boundary information can be detected during the deconvolution process, the better quality of the estimated results Zhou and Komodakis [2014]; Pan et al. [2016a]. In Fig. 4.2, we compare the scene flow estimation results with the state-of-the-art solutions on different blurred images. It could be observed that the scene flow estimation performance deteriorates quickly w.r.t. the image blur because of the inaccuracy at boundaries.

On the other hand, motion segmentation or moving object segmentation alone is also very challenging as the objects could be rigid, non-rigid, and deformable. How to unify these different scene models and achieve moving object segmentation is an active research direction. This chapter focuses on outdoor traffic scenes with multiple moving objects, such as vehicles, cyclists, and pedestrians. Specifically, we exploit both the semantic cue and 3D geometry cue to better handle moving object segmentation together with scene flow estimation and stereo deblurring.

Furthermore, existing works fail to exploit the connections between stereo deblurring, scene flow estimation and moving object segmentation, which actually are closely connected. Specifically, better scene flow estimation and moving object segmentation will enable better stereo deblurring. Correspondingly, stereo deblurring and moving object segmentation also help scene flow estimation. However, building their intrinsic connections is not easy as the dynamic scenes could be rather generic, from a static scene to a highly dynamic scene consisting of multiple moving objects (vehicles, pedestrians and *etc.*). Having a unified formulation for the dynamic scenes is highly desired. We propose to exploit the piecewise plane model for the dynamic scene structure, and under this formulation, the joint task of scene flow estimation, stereo deblurring and moving object segmentation has been expressed as the parameter estimation for each planar, the camera motion and pixel labelling. Therefore, we put these three tasks in a loop under a unified energy minimization formulation in which the intra-relation has been effectively exploited.



(a) Blurred image (b) GT Flow c) Kim and Lee [2015]

(d) Sellent et al. [2016] (e) Pan et al. [2017b] (f) Our flow estimation

Figure 4.2: *Scene flow estimation results for an outdoor scene. (a) Blurred reference image from* **BlurData-1**. *(b) Ground truth optical flow for the scene. (c) Estimated flow by Kim and Lee [2015]. (d) Estimated flow by Sellent et al. [2016] which uses Vogel et al. [2015] to estimate scene flow. This approach ranks as one of the top 3 approaches on KITTI scene flow benchmark Geiger et al. [2013]. (e) Estimated flow by Pan et al. [2017b]. (f) Our flow estimation result. Compared with these state-of-the-art methods, our method achieves the best performance.*

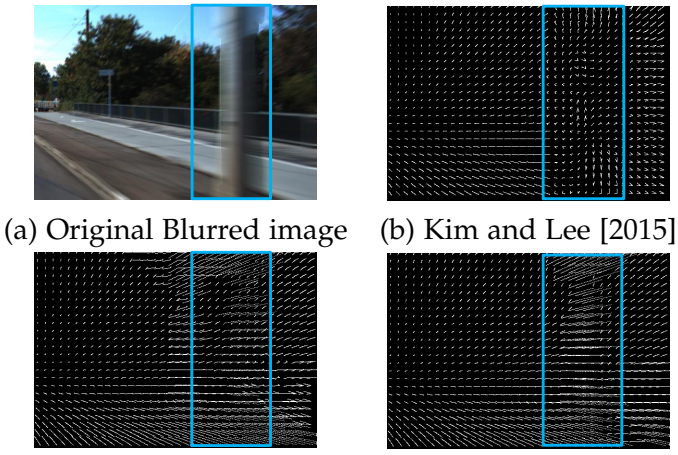
In our previous work Pan et al. [2017b], we only consider the relationship between optical flow and deblurring without adding segmentation information. We extend the previous work significantly in the following ways:

- We propose a novel joint optimization framework to estimate the scene flow, segment moving objects and restore the latent images for generic dynamic scenes. Our deblurring objective benefits from improved boundaries information and the estimated scene structure.

- We integrate high-level semantic cues for camera motion and scene structure estimation by exploiting the intrinsic connection between semantic segmentation and moving object segmentation.

- We propose a method to exploit motion segmentation information in aiding the challenging video deblurring task. Similarly, the scene flow and objects boundary objective allow deriving more accurate pixel-wise spatially varying blur kernels (see Section.4.4.2).

- Extensive experiments demonstrate that our method can successfully handle complex real-world scenes depicting fast-moving objects, camera motions, uncontrolled lighting conditions, and shadows.



(a) Original Blurred image (b) Kim and Lee [2015]

(c) Sellent et al. [2016] (d) Ours

Figure 4.3: *Blur kernel estimation for an outdoor scene. (a) Blurred reference image from* **BlurData-1**. *(b) Blur kernel estimation by Kim and Lee [2015]. (c) Blur kernel estimation by Sellent et al. [2016]. (d) Our blur kernel estimation. Compared with these monocular and stereo deblurring methods, our method achieves more accurate blur kernel estimation.*

4.3 Related Work

Image deblurring (even under stereo configuration) is generally an ill-posed problem. Thus certain assumptions or additional constraints are required to regularize the solution space. Numerous methods have been proposed to address the problem Kim and Lee [2015]; Sellent et al. [2016]; Pan et al. [2017b]; Li et al. [2018c]; Hu et al. [2014]; Sun et al. [2015]; Pan et al. [2016a]; Ren et al. [2017]; Gong et al. [2017a]. As per the system configuration, the methods can be roughly categorized into two groups: monocular based approaches and binocular or multi-view based approaches. We also briefly discuss recent efforts in deep learning-based deblurring, moving object segmentation, semantic segmentation, and scene flow estimation.

Single view deblur Monocular based deblurring approaches often assume that the captured scene is static or has uniform blur kernel Gupta et al. [2010], or need user interaction Pan et al. [2016a]. A series of widely-used priors and regularizers are based on image gradient sparsity, such as the total variational regularizer Perrone and Favaro [2014], the Gaussian scale mixture prior Fergus et al. [2006], the $l_1 \backslash l_2$ norm based prior Krishnan et al. [2011], and the l_0-norm regularize Xu et al. [2013]; Pan et al. [2014]. Non-gradient-based priors have also been proposed, such as the edge-based patch prior Sun et al. [2013], the colour line based prior Lai et al. [2015], and the dark/white channel prior Pan et al. [2016b]; Yan et al. [2017a]. Hu et al. [2014] proposed to jointly estimate the depth layering and remove non-uniform blur caused by

(a) Initial mask (b) Menze and Geiger [2015] (c) Kim and Lee [2015]

(d) Ours (no seg) (e) Ours (one layer) (f) Ours (two layer)

(g) Blurred image (h) Ground-truth (i) Kim and Lee [2015]

(j) Sellent et al. [2016] (k) Ours (one layer) (l) Ours (two layer)

Figure 4.4: *Scene flow results for an outdoor scenario. (a) and (g) The initial segmentation and blurred reference image from* **BlurData-1**. *(b) Estimated flow by Menze and Geiger [2015]. (c) Estimated flow by Kim and Lee [2015]. (d)-(f) Our flow estimation result. (d) Without semantic segmentation. (e) With semantic segmentation, one layer StereoSLIC. (f) With semantic segmentation, two-layer StereoSLIC. (h) The ground-truth latent image. (i) Deblurred result by Kim and Lee [2015]. (j) Deblurred result by Sellent et al. [2016]. (k) and (l) Our deblurred result. (k) Without semantic segmentation. (l) With semantic segmentation. The results show that our two-layer StereoSLIC could preserve edge information. Compared with both these state-of-the-art methods, our method achieves competitive performance. Best viewed in colour on the screen.*

the in-plane motion from a single blurred image. While this unified framework is promising, user input for depth layers partition is required, and potential depth values should be known in advance. Pan et al. [2016a] proposed an algorithm to jointly estimate object segmentation and camera motion by incorporating soft segmentation, but require user input. In practical settings, it is still challenging to remove strongly non-uniform motion blur captured in complex scenes.

Since blur parameters and a latent image are difficult to be estimated from a single image, the monocular based approaches are extended to video to remove blurs in dynamic scenes. In the work of Wulff and Black Wulff and Black [2014], a layered model is proposed to estimate the different motions of both foreground and background layers. Kim and Lee [2014] proposed a method based on a local linear motion without segmentation. This method incorporates optical flow estimation to guide the blur kernel estimation and is able to deal with certain object motion blur. In Kim and Lee [2015], a new method is proposed to simultaneously estimate optical flow and tackle the case of general blur by minimization a single non-convex energy function. Park and Lee [2017b] estimate camera poses and scene structures from severely blurred images and deblurring using the motion information.

Multi-view deblur As depth factor can significantly simplify the deblurring problem, multi-view deblurring methods have been proposed to leverage available depth information. Ezra and Nayar Nayar and Ben-Ezra [2004] proposed a hybrid imaging system, where a high-resolution camera captures the blurred frame and a low-resolution camera with faster shutter speed is used to estimate the camera motion. Xu and Jia [2012] inferred depth from two blurred images captured by a stereo camera and proposed a hierarchical estimation framework to remove motion blur caused by the in-plane translation. Sellent et al. [2016] proposed a video deblurring technique based on a stereo video, where 3D scene flow is estimated from the blurred images using a piecewise rigid 3D scene representation. Along the same line, Ren et al. [2017] proposed an algorithm where accurate semantic segmentation is known. In their work, they also used the pixel-wise non-linear kernel model to approximate motion trajectories in the video. While the performance of their experiments shows limited effective for images which included multiple types of moving objects. We Pan et al. [2017b] proposed a single framework to jointly estimate the scene flow and deblur the images in CVPR 2017, where the motion cues from scene flow estimation and blur information could reinforce each other. These two methods represent the state-of-the-art in multi-view video deblurring and will be used for comparisons in the experimental section.

Deep learning-based deblurring methods Recently, deep learning-based methods have been used to restore clean latent images. Gong et al. [2017a] esti-

mated flow from a single blurred image caused by camera motion through a fully convolutional deep neural network and recovered a clean image from the estimated flow. Su et al. [2017] introduced a deep learning solution to video deblurring, where a CNN is trained end-to-end to learn how to accumulate information across frames. However, they aimed to tackle motion blur from camera shake. Nah et al. [2017] proposed a multi-scale convolutional neural network that restores latent images in an end-to-end manner without assuming any restricted blur kernel model. Kim et al. [2017]; Kim et al. [2018] proposed a novel network layer that enforces temporal consistency between consecutive frames by dynamic temporal blending which compares and adaptively shares features obtained at different time steps. Kupyn et al. [2018b] presented an end-to-end learning approach for motion deblurring. The model they used is Conditional Wasserstein GAN with gradient penalty and perceptual loss based on VGG-19 activations. Tao et al. [2018] propose a light and compact network, SRN-DeblurNet, to deblur the image. Jin et al. [2018] proposed to restore a video with fixed length from a single blurred image. However, deep deblurring methods generally need a large dataset to train the model and usually require sharp images provided as supervision. In practice, blurred images do not always have corresponding ground-truth sharp images.

Moving object segmentation According to the level of supervision required, video segmentation techniques can be broadly categorized as unsupervised, semi-supervised and supervised methods. Unsupervised methods Papazoglou and Ferrari [2013] use a rapid technique to produce a rough estimate of which pixels are inside the object based on motion boundaries in pairs of subsequent frames. It then automatically bootstraps an appearance model based on the initial foreground estimate, and uses it to refine the spatial accuracy of the segmentation and segment the object in frames where it does not move. The works Faktor and Irani [2014]; Wang et al. [2015]; Wang et al. [2018] extend the concept of salient objects detection Sundaram et al. [2010] as prior knowledge to infer the objects. Semi-supervised video segmentation, which also refers to label propagation, is usually achieved via propagating human annotation specified on one or a few key-frames onto the entire video sequence Hariharan et al. [2015]; Shankar Nagaraja et al. [2015]; Tsai et al. [2016a]. The idea of combining the best from both the CNN model and MRF/CRF model is not new. A video object segmentation method by Jang and Kim Jang and Kim [2017] performs MRF optimization to fuse the outputs of a triple-branch CNN. However, the loosely-coupled combination cannot fully exploit the strength of MRF/CRF models. Supervised methods require tedious user interaction and iterative human corrections. These methods can attain high-quality boundaries while needing human supervision Wang et al. [2014]; Fan et al. [2015]. Yan Yan et al. [2017b] proposed a multi-task ranking

model for the higher-level weakly-supervised actor-action segmentation task.

Semantic segmentation Another crucial factor in computing latent clean image is detecting moving objects boundaries. The general problem is that the object boundaries with mixed foreground and background pixels can lead to severe ringing artefacts. Semantic segmentation can help to provide objects information as initialization. He et al. [2016] proposed the ResNets to combat the vanishing gradient problem in training very deep convolutional networks. Wu et al. [2019] obtain the semantic segmentation masks with the ResNet-38 network. Lin et al. [2017] present RefineNet with multi-resolution fusion (MRF) to combine features at different levels, chained residual pooling (CRP) to capture background context, and residual convolutional units (RCUs) to improve end-to-end learning. Tsai et al. [2016b] first generated the object-like tracklets and then adopted a sub-modular function to integrate object appearances, shapes and motions to co-select tracklets that belong to the common objects. Taking one step further, the Deep Parsing Network (DPN) Liu et al. [2018] is designed to approximate the mean-field inference for MRFs in one pass.

Optical flow estimation Menze and Geiger [2015] proposed a novel model and dataset for 3D scene flow estimation with an application to autonomous driving. Pan et al. [2017b] proposed a single framework to jointly estimate the scene flow and deblur the images. Taniai et al. [2017] presented a multi-frame method for efficiently computing scene flow (dense depth and optical flow) and camera ego-motion for a dynamic scene observed from a moving stereo camera rig. Yin and Shi [2018] proposed an unsupervised learning framework GeoNet for jointly estimating monocular depth, optical flow and camera motion from video. Gong et al. [2017a] directly estimate the motion flow from the blurred image through a fully-convolutional deep neural network (FCN) and recover the unblurred image from the estimated motion flow. PWC-Net Sun et al. [2018] uses the current optical flow estimate to warp the CNN features of the second image. It then uses the warped features and features of the first image to construct a cost volume processed by a CNN to estimate the optical flow. The FlowNet by Dosovitskiy et al. [2015] represented a paradigm shift in optical flow estimation. The work shows the feasibility of directly estimating optical flow from raw images using a generic U-Net CNN architecture. FlowNet 2.0 Ilg et al. [2017] develop a stacked architecture that includes warping of the second image with the intermediate optical flow, which decreases the estimation error by more than 50% than the original FlowNet.

4.4 Problem Formulation

In this chapter, we propose to solve the challenging and practical problem of stereo deblurring by using consecutive stereo image pairs of a calibrated camera in complex dynamic environments, where the blur is caused by the camera motion and the objects' motion. Under the problem setup, stereo deblurring and the scene flow estimation is already deeply coupled, i.e., , stereo deblurring depends on the solution of the scene flow estimation while the scene flow estimation also needs the solution of stereo deblurring. Besides, with the multiple moving objects representation of the observed scene, moving object segmentation also closely relates to both scene flow estimation and stereo deblurring, i.e., , improper moving object segmentation could result in dramatical changes in scene flow estimation and stereo deblurring especially along the object boundaries Sevilla-Lara et al. [2016]. Therefore, we could conclude that the scene flow estimation, Moving object segmentation and video deblurring are deeply coupled under our problem setup.

To better exploit the deeply coupling nature of the problem, we propose to formulate our problem as a joint estimation of scene flow, Moving object segmentation and stereo image deblurring for complex dynamic scenes. In particular, we rely on the assumptions that the scene can be well approximated by a collection of 3D planes Yamaguchi et al. [2013] belonging to a finite number of objects [1] performing rigid motions individually Menze and Geiger [2015]. Therefore, the problem of scene flow estimation can be reformulated as the task of geometric and motion estimation for each 3D plane. The rigid motion is defined for each moving object, which naturally encodes the Moving object segmentation information. The blurred stereo images are generated due to the camera motion, multiple moving objects motion and the 3D scene structure, which are all characterized by the scene flow estimation and the Moving object segmentation. Specifically, our structured blur kernels are expressed with the geometry and motion of each 3D plane.

4.4.1 Blurred Image Formation based on the Structured Pixel-wise Blur Kernel

Blurred images are formed by the integration of light intensity emitted from the dynamic scene over the aperture time interval of the camera. We assume that the blurred image $\mathbf{B}$ can be generated by the integral of the latent high frame-rate image sequence $\{\mathbf{L}_n\}$ during the exposure time. This model follows by Kim and Lee [2014]; Gupta et al. [2010]; Whyte et al. [2012]; Dai and Wu [2008], which supposes the integration of light intensity happens in pixel

[1] The background is regarded as a single 'object' due to the camera motion only.

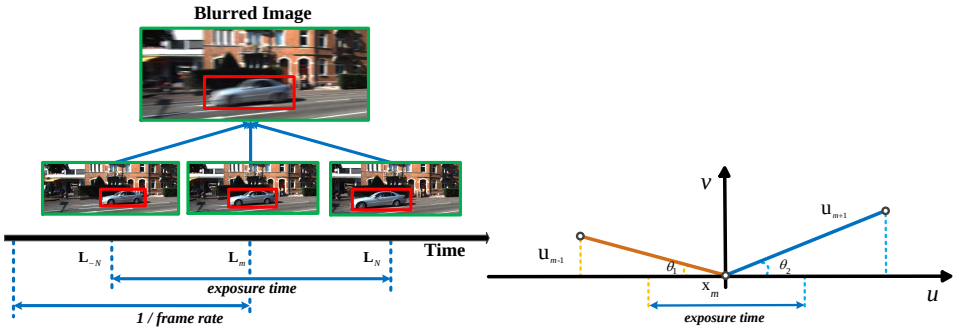


Figure 4.5: *The pipeline of generating blurred images. We approximate the motion blur kernel as a piece-wise linear function based on bi-direction optical flows and generate blurred images by averaging consecutive frames whose relative motions between two neighbouring frames are known. Notably, ground truth sharp image is chosen to be the middle one.*

colour space over the shutter time of the camera.[2] This defines the blurred image frame in the video sequence as

$$\mathbf{B}_m = \frac{1}{2N+1} \sum_{n=-N}^{N} \mathbf{L}_n, \tag{4.1}$$

where $\mathbf{B}_m$ is the m^{th} blurred image in the video sequence, $\mathbf{L}_n$, $n \in [-N, N]$ denotes latent frames that generate the blurred image. The middle frame $\mathbf{L}_m$ among the latent frames is defined as the deblurred image, which associated with $\mathbf{B}_m$. This integration model has been widely used in the image/video deblurring literature Seok Lee and Mu Lee [2013]; Kim and Lee [2014]; Gong et al. [2017a], which has also been used in Nah et al. [2017]; Su et al. [2017]; Kim et al. [2017] to generate realistic blurred images from high frame-rate videos. With optical flow, we can transform $\mathbf{L}_n$ with $\mathbf{L}_m$. Thus, the blur can be modelled by bi-directional optical flows. We approximate the kernel as piece-wise linear using bidirectional optical flows, where the kernel $\mathbf{A}_m^{\mathbf{x}}$ is spatially varying for each pixel.

$$\mathbf{B}_m(\mathbf{x}) = \text{vec}(\mathbf{A}_m^{\mathbf{x}})^T \text{vec}(\mathbf{L}_m), \tag{4.2}$$

where $\mathbf{x} \in \mathbb{R}^2$ denotes the pixel location in the image domain, vec denotes the vectorization operator, $\mathbf{A}_m^{\mathbf{x}} \in \mathbb{R}^{h \times w}$ is the blur kernel for each pixel $\mathbf{x}$, where h, w are the image size. In order to handle multiple types of blurs, we assumed that the blur kernel $\mathbf{A}_m^{\mathbf{x}}$ can be linearized in terms of a motion

[2] We notice that several methods model the integration in the raw sensor value and consider the effects of CRFs (camera response function) on motion deblurring. These yield a slightly different solution for deblurring Nah et al. [2017]; Tai et al. [2013].

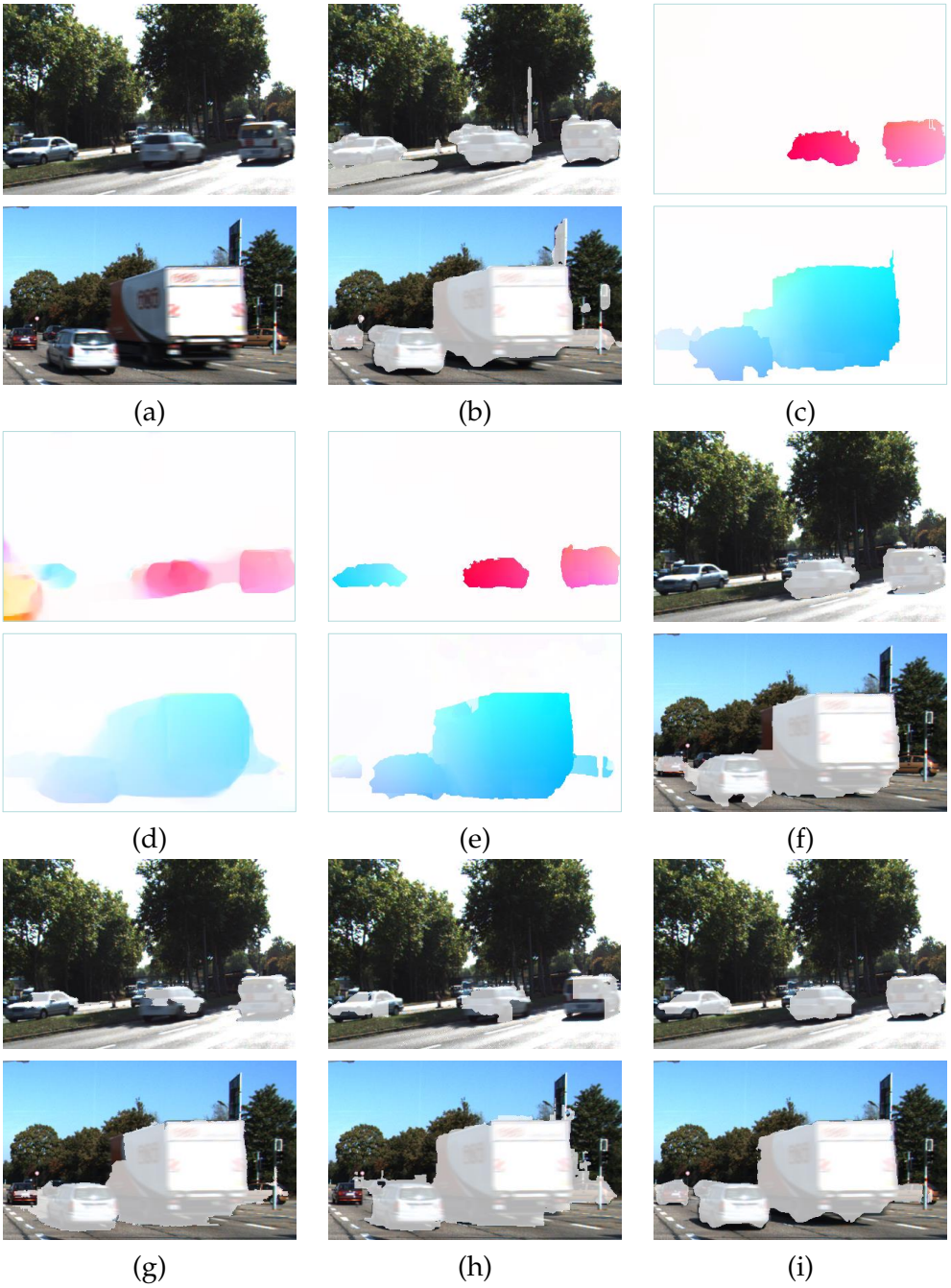


Figure 4.6: *Scene flow and moving object segmentation results for an outdoor scenario from* **BlurData-1**. *(a) Input blurred image. (b) Input semantic segmentation. (c) Estimated flow by Menze and Geiger [2015]. (d) Estimated flow by Kim and Lee [2015]. (e) Our flow estimation result. (f) Segmentation result by Menze and Geiger [2015]. (g) Segmentation result by Papazoglou and Ferrari [2013]. (h) Segmentation result by Faktor and Irani [2014]. (i) Our segmentation result. Compared with both these state-of-the-art methods, our method achieves competitive performance. Best viewed in colour on the screen.*

vector, which can be expressed as Kim and Lee [2014],

$$\mathbf{A}_m^{\mathbf{x}}(\tilde{u},\tilde{v}) = \begin{cases} \dfrac{\delta(\tilde{u}v_{m+1} - \tilde{v}u_{m+1})}{\tau||\mathbf{u}_{m+1}||}, & \text{if } \tilde{\mathbf{u}} \in [\mathbf{0}, \boldsymbol{\text{ø}}\mathbf{u_{m+1}}], \\ \dfrac{\delta(\tilde{u}v_{m-1} - \tilde{v}u_{m-1})}{\tau||\mathbf{u}_{m-1}||}, & \text{if } \tilde{\mathbf{u}} \in [\mathbf{0}, \boldsymbol{\text{ø}}\mathbf{u_{m-1}}], \\ \mathbf{0}, & \text{otherwise}, \end{cases} \tag{4.3}$$

where $\tau = \frac{1}{2} \times$ exposure time $\times$ frame rate, δ denotes the Kronecker delta function, $\mathbf{u}_{m+1}$ and $\mathbf{u}_{m-1}$ are the bidirectional optical flows at frame m. In particular, $\tilde{\mathbf{u}} = (\tilde{u}, \tilde{v})$ which denotes the motion between exposure time, the kernel model is shown in Fig. 4.5. We obtain the blur kernel matrix $\mathbf{A}_m \in \mathbb{R}^{(h\times w)\times(h\times w)}$ by stacking $\text{vec}(\mathbf{A}_m^{\mathbf{x}})$ over the whole image domain. This leads to the blur model for the image as

$$\text{vec}(\mathbf{B}_m) = \mathbf{A}_m \text{vec}(\mathbf{L}_m). \tag{4.4}$$

We omit the vectorize symbol in the following sections. We can cast the kernel estimation problem as a motion estimation problem.

In our setup, the stereo video provides the depth information for each frame. Based on our piece-wise planar assumptions on the scene structure, optical flows for pixels lying on the same plane are constrained by a single homography. In particular, we represent the scene in terms of superpixels and a finite number of objects with rigid motions. We denote $\mathcal{S}$ and $\mathcal{O}$ as the set of superpixels and moving objects, respectively. Each superpixel $i \in \mathcal{S}$, is associated with a region $\mathcal{S}_i$ in the image, each region is denoted by a plane variable $\mathbf{n}_{i,k_i} \in \mathbb{R}^3$ in 3D ($\mathbf{n}_{i,k_i}^T\mathbf{x} = 1$ for $\mathbf{x} \in \mathbb{R}^3$), where $k_i \in \{1, \cdots, |\mathcal{O}|\}$ denotes the i^{th} superpixel associated with the k^{th} object. Object inheriting its corresponding motion parameters $\mathbf{o}_{k_i} = (\mathbf{R}_k, \mathbf{t}_k) \in \mathbb{SE}(3)$, where $\mathbf{R}_k \in \mathbb{R}^{3\times3}$ is the rotation matrix and $\mathbf{t}_k \in \mathbb{R}^3$ is the translation vector. Note that $(\mathbf{n}, \mathbf{o})$ encodes the scene flow information Menze and Geiger [2015], where $\mathbf{n} = \{\mathbf{n}_{i,k_i} | i \in \mathcal{S}\}$ and $\mathbf{o} = \{\mathbf{o}_{k_i} | k_i \in \mathcal{O}\}$. Given the motion parameters $\mathbf{o}_{k_i}$, we can obtain the homography defined by superpixel i as

$$\mathbf{H}_i = \mathbf{K}(\mathbf{R}_k - \mathbf{t}_k\mathbf{n}_{i,k_i}^T)\mathbf{K}^{-1}, \tag{4.5}$$

where $\mathbf{K} \in \mathbb{R}^{3\times3}$ is the camera calibration matrix. We note that, $\mathbf{H}_i$ relates corresponding pixels across two frames.

The optical flow is then defined as

$$\mathbf{u}_i = \mathbf{x} - \pi(\mathbf{H}_i\mathbf{x}), \tag{4.6}$$

where we denote $\mathbf{x}^* = \pi(\mathbf{H}_i\mathbf{x})$. $\pi(\cdot)$ is the perspective division such that

$\pi([x, y, z]^T) := [x/z, y/z]^T$. This shows that the optical flows for pixels in a superpixel are constrained by the same homography. Thus, it leads to a structured version of blur kernel defined in Eq. (4.3).

4.4.2 Moving object segmentation

Semantic segmentation breaks the image into semantically consistent regions such as road, car, person, sky, *etc.*. Our algorithm computes each region independently based on the semantic class label, resulting in more precise moving object segmentation and flow estimation, particularly at object boundaries. The provided additional information about object boundaries contributes to avoiding ringing and boundary artefact.

A general problem in motion deblurring is that the moving object boundaries with mixed foreground and background pixels can lead to severe ringing artefacts (see Fig. 4.1 for details). Most motion deblurring methods address this problem by segmenting blurred images into regions or layers where different kernels are estimated and applied for image restoration Tai et al. [2010]; Wulff and Black [2014]; Pan et al. [2016a]. Segmentation on blurred images is difficult due to ambiguous pixels between regions, but it plays an important role in motion deblurring.

In our formulation, we use ResNet38 Wu et al. [2019] to predict the semantic label map $\mathbf{M} \in \mathbb{N}^{w \times h}$ as initialization for our "generalized stereo deblur" model. This approach ranks higher on Cityscapes Cordts et al. [2016] where the image is captured on an urban street. A $\mathbf{M}$ determines the predicted semantic instance label for each pixel in each frame, which provides strong prior for boundary detection, motion estimation, and label classification for superpixels.

We first set roads, sky and trees are static background layer, and assume other things have a higher moving possibility to be the foreground layer. Here, a convincing background layer will provide the inline feature points on the background for ego-motion estimation. Then, we can estimate the disparity map and the 6-DOF camera motion using stereo matching and visual odometry with coarse background segmentation. We identify regions inconsistent with the estimated camera motion and estimate the motion at these regions separately. Each motion parameter $\mathbf{o}$ is generated by moving clusters from sparse features points. In particular, the motion hypothesis is then generated using the 3-point RANSAC algorithm implemented in Geiger et al. [2011]. These inconsistent regions can match with our prior $\mathbf{M}$. This helps to maintain the boundary information for moving objects and avoid ringing artefacts (see Fig. 4.4 for details).

Each slanted plane in the image is labelled as moving or static according to the ego-motion estimation. With the semantic segmentation masks, we can give each superpixel an additional label, foreground or background. We then

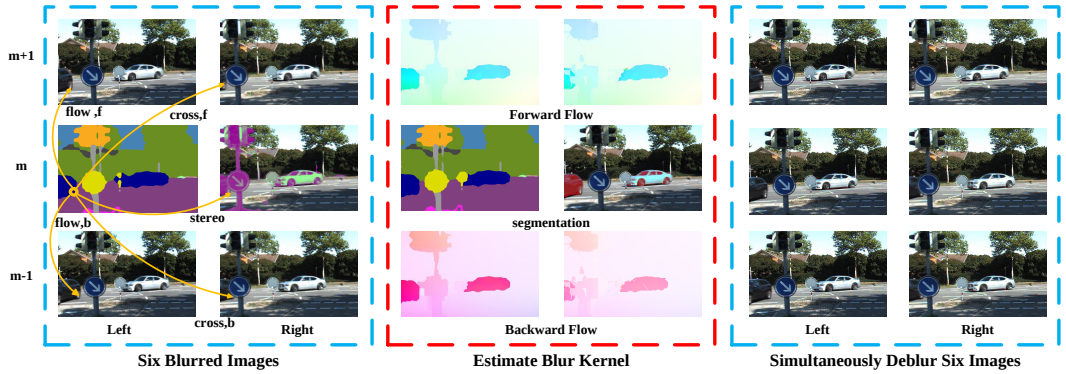


Figure 4.7: *Illustration of our 'generalized stereo deblurring' method. We simultaneously compute four scene flows (in two directions and in two views), moving object segmentation and deblur six images. In case the input contains only two images, we use the reflection of the flow forward as the flow backward in the deblurring part.*

use the label map to initialize object label k_i for each superpixel i. If most pixels' semantic label in i^{th} superpixel are fore/background, the superpixel is more likely to belongs to the fore/background.

$$k_i(\mathbf{x}) \in \begin{cases} \{1\} & \text{, if } \mathbf{M}(\mathbf{x}) = \text{Background} \\ \{2, \cdots, |\mathcal{O}|\} & \text{, if } \mathbf{M}(\mathbf{x}) = \text{Foreground.} \end{cases} \tag{4.7}$$

Although we provide over segmentation initially as shown in Fig. 4.1(a), our algorithm can precisely segment the moving objects after optimization (Fig. 4.1(b)) and provide more accurate motion boundaries information for optical flow estimation (Fig. 4.1(d)), and thereby facilitates stereo video deblurring (Fig. 4.1(h)).

With the semantic segmentation prior, we label each superpixel and objects more accurately, and our approach obtains superior results in moving object segmentation and scene flow estimation (see Fig. 4.6 for details).

In the optimization part, instead of giving sample k_i for every superpixel randomly, we use the semantic segmentation prior $\mathbf{M}$ to give a more reliable sample for each superpixel (see Section 4.5.1 for detail).

4.4.3 Energy Minimization

We formulate the problem in a single framework as a discrete-continuous optimization problem to jointly estimate the scene flow, moving object segmentation and deblur the stereo images. Specifically, our model is defined

as

$$\mathbf{E}(\mathbf{n}, \mathbf{o}, \mathbf{L}) = \underbrace{\sum_{i \in \mathcal{S}} \phi_i(\mathbf{n}_i, \mathbf{o}, \mathbf{L})}_{\text{data term}} + \underbrace{\sum_{i,j \in \mathcal{S}} \phi_{i,j}(\mathbf{n}_i, \mathbf{n}_j, \mathbf{o})}_{\substack{\text{scene flow} \\ \text{smoothness term}}} + \underbrace{\sum_{m} \psi_m(\mathbf{L})}_{\substack{\text{latent image} \\ \text{regularisation}}}, \tag{4.8}$$

where $i,\ j$ denotes the set of adjacent superpixels in $\mathcal{S}$. The function consists of a data term, a smoothness term for scene flow, and a spatial regularization term for latent images. Our model is initially defined on three consecutive pairs of stereo video sequences. It can also allow the input with two pairs of frames. Details are provided in Section 4.6. The energy terms are discussed in Section 4.4.4, Section 4.4.5, and Section 4.4.6, respectively.

In Section 4.5, we perform the optimization in an alternative manner to handle mixed discrete and continuous variables, thus allowing us to jointly estimate scene flow, moving object segmentation and deblur the images.

4.4.4 Data Term

Our data term involves mixed discrete and continuous variables, and are of three different kinds. The first kind encodes the fact that the corresponding pixels across the six latent images should have a similar appearance, i.e., , brightness constancy. This lets us write the term as

$$\phi_i^1(\mathbf{n}_i, \mathbf{o}, \mathbf{L}) = \theta_1 \sum_{\mathbf{x} \in \mathcal{S}_i} |\mathbf{L}(\mathbf{x}) - \mathbf{L}^*(\mathbf{x}^*)|_1, \tag{4.9}$$

where $\mathbf{L}$ denotes the reference image, $\mathbf{L}^*$ denotes the target image, the superscript $* \in \left\{\mathbf{stereo}, \mathbf{flow}_{f,b}, \mathbf{cross}_{f,b}\right\}$ denote the warping direction to other images and $(\cdot)_{f,b}$ denotes the forward and backward direction, respectively (see Figure 4.7). The terms is defined by summing the matching costs of all pixels inside superpixel i. We adopt the robust ℓ_1 norm to enforce its robustness against noise and occlusions.

Our second potential, similar to one term used in Menze and Geiger [2015], is defined as

$$\phi_i^2(\mathbf{n}_i, \mathbf{o}) = \begin{cases} \theta_2 \sum_{\mathbf{x} \in \mathcal{S}_i} \rho_{\alpha_1}(||\mathbf{x} - \mathbf{x}^*||_2) & \text{, if } \mathbf{x} \in \Pi_{\mathbf{x}}, \\ 0 & \text{, otherwise,} \end{cases} \tag{4.10}$$

where $\rho_\alpha(\cdot) = \min(|\cdot|, \alpha)$ denotes the truncated l_1 penalty function. More specifically, it encodes the information that the warping of feature points $x \in \Pi_x$ based on $\mathbf{H}^*$ should match its extracted correspondences $\mathbf{x}^*$ in the target view. In particular, $\Pi_{\mathbf{x}}$ is obtained in a similar manner as Menze and Geiger [2015].

The third data term, making use of the observed blurred images, is defined as

$$\mathbf{E}_i^3(\mathbf{n}_i, \mathbf{o}, \mathbf{L}) = \theta_3 \sum_m \sum_\partial \|\partial \mathbf{A}_m(\mathbf{n}_i, \mathbf{o}) \mathbf{L}_m - \partial \mathbf{B}_m\|_2^2, \tag{4.11}$$

where ∂ denotes the Toeplitz matrices corresponding to the horizontal and vertical derivative filters. This term encourages the intensity changes in the estimated blurred image to be close to that of the observed blurred image.

4.4.5 Smoothness Term for Scene Flow

Our energy model exploits a smoothness potential that involves discrete and continuous variables. It is similar to the ones used in Menze and Geiger [2015]. In particular, our smoothness term includes three different types.

The first one is to encode the compatibility of two superpixels that share a common boundary by respecting the depth discontinuities. We define our potential function as

$$\phi_{i,j}^1(\mathbf{n}_i, \mathbf{n}_j) = \theta_4 \sum_{\mathbf{x} \in \mathcal{B}_{i,j}} \rho_{\alpha_2}(\omega_{i,j}(\mathbf{n}_i, \mathbf{n}_j, \mathbf{x})), \tag{4.12}$$

where $d(\mathbf{n}_i, \mathbf{x})$ is the disparity of pixel $\mathbf{x}$ in superpixel i in the reference disparity map, $\omega_{i,j}(\mathbf{n}_i, \mathbf{n}_j, \mathbf{x}) = d(\mathbf{n}_i, \mathbf{x}) - d(\mathbf{n}_j, \mathbf{x})$ are the dissimilarity value of disparity for pixel $\mathbf{x} \in \mathcal{B}_{i,j}$ on the boundary.

The second potential is to encourage the neighbouring superpixels to orient in similar directions. It is expressed as

$$\phi_{i,j}^2(\mathbf{n}_i, \mathbf{n}_j) = \theta_5 \rho_{\alpha_3} \left(1 - \frac{|\mathbf{n}_i^T \mathbf{n}_j|}{\|\mathbf{n}_i\| \, \|\mathbf{n}_j\|} \right). \tag{4.13}$$

The shadows of moving objects have motion boundaries but no disparity discontinuities. However, the motion boundaries are co-aligned with disparity discontinuities in general. Thus, we use the third and fourth potential encodes these discontinuities. This potential can be expressed as

$$\begin{aligned} &\phi_{i,j}^3(\mathbf{n}_{i,k_i}, \mathbf{n}_{j,k_j}) = \\ &\exp\left\{ -\frac{\lambda}{|\mathcal{B}_{i,j}|} \sum_{\mathbf{x} \in \mathcal{B}_{i,j}} \omega_{i,j}(\mathbf{n}_i, \mathbf{n}_j, \mathbf{x})^2 \right\} \times \frac{|\mathbf{n}_i^T \mathbf{n}_j|}{\|\mathbf{n}_i\| \, \|\mathbf{n}_j\|} \times [k_i \neq k_j], \end{aligned} \tag{4.14}$$

where $|\mathcal{B}_{i,j}|$ denotes the number of pixels shared along boundary between

superpixels i and j.

$$\phi^4_{i,j}(\mathbf{n}_{i,k_i}, \mathbf{n}_{j,k_j}, \mathbf{o}_{k_i}, \mathbf{o}_{k_j}) = \exp\left\{ -\frac{\lambda}{|\mathcal{B}_{i,j}|} \sum_{\mathbf{x}\in\mathcal{B}_{i,j}} G(\mathbf{o}_{k_i}, \mathbf{o}_{k_j}) \right\} \times \frac{|\mathbf{n}^T_{i,k_i}\mathbf{n}_{j,k_j}|}{\left\|\mathbf{n}_{i,k_i}\right\| \left\|\mathbf{n}_{j,k_j}\right\|} \times [k_i \neq k_j], \tag{4.15}$$

$$G(\mathbf{o}_{k_i}, \mathbf{o}_{k_j}) = \theta_r(\text{trace}(\mathbf{R}^T_{k_i}\mathbf{R}_{k_j}) - 1)/2 + \theta_t(\exp(-\left\|\mathbf{t}_{k_i} - \mathbf{t}_{k_j}\right\|)),$$

where $[\cdot]$ denotes the Iverson bracket. This encodes our belief that motion boundaries are more likely to occur at 3D folds or discontinuities than within smooth surfaces.

4.4.6 Regularization Term for Latent Images

Several works Krishnan and Fergus [2009]; Krishnan et al. [2011] have studied the importance of spatial regularization in image deblurring. In our model, we use a total variation term to suppress the noise in the latent image while preserving edges, and penalize spatial fluctuations. Therefore, our potential takes the form

$$\psi_m = \sum_{\mathbf{x}} |\nabla \mathbf{L}_m|. \tag{4.16}$$

Note that the total variation is applied to each colour channel separately.

4.5 Solution

The optimization of our energy function defined in Eq.(4.8), involving discrete and continuous variables, is very challenging to solve. Recall that our model involves two set of variables, namely scene flow variables and latent clean images. Fortunately, given one set of variables, we can solve the other efficiently. Therefore, we perform the optimization iteratively by the following steps,

- Fix latent clean image $\mathbf{L}$, solve scene flow by optimizing Eq.(4.17) (See Section 4.5.1).
- Fix scene flow parameters, $\mathbf{n}$ and $\mathbf{o}$, solve latent clean images by optimizing Eq.(4.18) (See Section 4.5.2).

In the following sections, we describe the details for each optimization step.

4.5.1 Scene flow estimation

We fix latent images, namely $\mathbf{L} = \tilde{\mathbf{L}}$. Eq.(4.8) reduces to

$$\min_{\mathbf{n},\mathbf{o}} \sum_{i\in\mathcal{S}} \sum_{m=1}^{3} \mathbf{Œ}_i^m(\mathbf{n}_i, \mathbf{o}, \tilde{\mathbf{L}}) + \sum_{i,j\in\mathcal{S}} \sum_{m=1}^{4} \mathbf{Œ}_{i,j}^m(\mathbf{n}_i, \mathbf{n}_j, \mathbf{o}), \tag{4.17}$$

which becomes a discrete-continuous CRF optimization problem.

We use the sequential tree-reweighted message passing (TRW-S) method in Menze and Geiger [2015] to find an approximate solution. Since the label k of $\mathbf{n}_i$ of each superpixel is drawing randomly, we use the semantic segmentation prior $\mathbf{M}$ to give a more reliable sample of each superpixel. We modify their sampling strategy as shown in Algorithm 1.

Algorithm 1: TRW-S Optimization

Input : $\tilde{\mathbf{L}}$, $\mathbf{M}$, $\mathbf{B}$.
1 Initialize $\mathbf{n}$ and $\mathbf{o}$ as described in 'Initialization'.
2 Iteration times = 3
3 For all $i \in S$
4 Draw sample for $\mathbf{n}_i$ (Gaussian)
5 Draw sample for $\mathbf{k}_i$(M)
6 For all $k \in \mathcal{O}$
7 Draw sample for $\mathbf{o}_k$ (MCMC)
8 Run TRW-S Kolmogorov [2006] on discretized problem
Output: $\mathbf{n}_{i,k_i}, \mathbf{o}_{k_i}$

4.5.2 Deblurring

Given the scene flow parameters, namely $\mathbf{n} = \tilde{\mathbf{n}}$, and $\mathbf{o} = \tilde{\mathbf{o}}$, the blur kernel matrix, $\mathbf{A}_m$ is derived based on Eq.(4.3), and Eq.(4.6). The objective function in Eq. (4.8) becomes convex with respect to $\mathbf{L}$ and is expressed as

$$\min_{\mathbf{L}} \sum_{\mathcal{S}_i\in\mathcal{S}} \mathbf{Œ}_i^1(\tilde{\mathbf{n}}_i, \tilde{\mathbf{o}}, \mathbf{L}) + \mathbf{Œ}_i^3(\tilde{\mathbf{n}}_i, \tilde{\mathbf{o}}, \mathbf{L}) + \psi_m(\mathbf{L}). \tag{4.18}$$

In order to obtain sharp image $\mathbf{L}$, we adopt the conventional convex optimization method Chambolle and Pock [2011] and derive the primal-dual

updating scheme as follows

$$\begin{cases} \mathbf{p}^{r+1} = \dfrac{\mathbf{p}^r + \gamma \nabla \mathbf{L}_m^r}{\max(1, \mathrm{abs}(\mathbf{p}^r + \gamma \nabla \mathbf{L}_m^r))} \\ \mathbf{q}^{r+1} = \dfrac{\mathbf{q}^r + \gamma \theta_1 (\mathbf{L}_m^r - \mathbf{L}_*^r)}{\max(1, \mathrm{abs}(\mathbf{q}^r + \gamma \theta_1 (\mathbf{L}_m^r - \mathbf{L}_*^r))} \\ \mathbf{L}_m^{r+1} = \arg\min_{\mathbf{L}_m} \sum_i \theta_3 \sum_\partial \|\partial \mathbf{A}_m \mathbf{L}_m - \partial \mathbf{B}_m\|_2^2 + \\ \dfrac{\left\| [\mathbf{L}_m - \eta((\nabla \mathbf{p}_m^{r+1})^T + \theta_1 (\mathbf{q}^{r+1} - \mathbf{q}_*^{r+1})^T)] - \mathbf{L}_m^r \right\|^2}{2\eta}, \end{cases} \tag{4.19}$$

where $\mathbf{p}_m$, $\mathbf{q}_{m,*}$ are the dual variables, γ and η are the step variants which can be modified at each iteration, and r is the iteration number.

Algorithm 2: Proposed deblurring system

Input : Stereo Blurred Image Sequences **B**, Semantic Segmentation of Reference Image Pair.

1 Initialize **n** and **o** as described in 'Initialization'.
2 Run Algorithm 1 minimize Eq. (4.17). Estimate scene flow and moving object segmentation map.
3 Run Primal-Dual Chambolle and Pock [2011] minimize Eq. (4.18). Restoration clean image.
4 Repeat steps 2,3 until reaches a preset iteration number (3 in our experiment).

Output: Latent Images **L**, Moving object Segmentation Map, Scene Flow

4.6 Experiments

To demonstrate the effectiveness of our method, we evaluate it based on two datasets: the synthetic chair sequence Sellent et al. [2016] and the KITTI dataset Geiger et al. [2013]. We report our results on both datasets in the following sections.

4.6.1 Experimental Setup

Initialization. Our model in Section 4.4 is formulated on three consecutive stereo pairs. In particular, we treat the middle frame in the left view as the reference frame. We adopt StereoSLIC Yamaguchi et al. [2013] to generate superpixels. Given the stereo images, we apply the approach in Geiger et al. [2011] to obtain sparse feature correspondences. The traditional SGM Hirschmuller [2008] method is applied to obtain the disparity map. We further leverage the semantic segmentation results to provide priors for motion segmentation. In particular, we applied the pre-trained model from the high-accuracy

Table 4.1: *Quantitative comparisons on disparity, optical flow and deblurring results on the KITTI dataset (BlurData-1).*

KITTI Dataset	Disparity		Flow		PSNR	
	m	m+1	Left	Right	Left	Right
Vogel et al. [2015]	8.20	8.50	13.62	14.59	/	/
Kim and Lee [2015]	/	/	38.89	39.45	28.25	29.00
Sellent et al. [2016]	8.20	8.50	13.62	14.59	27.75	28.52
Kupyn et al. [2018b]	/	/	/	/	28.34	28.73
Tao et al. [2018]	/	/	/	/	29.55	29.95
Pan et al. [2017b]	6.82	8.36	10.01	11.45	29.80	30.30
Ours	**6.18**	**7.49**	**9.83**	**11.14**	**29.85**	**30.50**
Baseline						
Vogel et al. [2015] and Kim and Lee [2015]	/	/	22.42	/	28.11	/

method Wu et al. [2019] on our blurred image. Based on the obtained semantics, we generate a binary map $\mathbf{M}$ which indicates the foreground as 1 and background as 0 by grouping the estimated semantics (see Section 4.4.2 for details.) The motion hypotheses are first generated using RANSAC algorithm implemented in Geiger et al. [2011]. Regarding the model parameters, we perform grid search on 30 reserved images. In our experiments, we fix the model parameters as $\theta_1 = 0.7$, $\theta_2 = 5.5$, $\theta_3 = 0.7$, $\gamma = 250$, $\theta_4 = 0.37$, $\theta_5 = 17$, $\lambda = 0.13$, $\alpha_1 = 3.39$, $\alpha_2 = 2.5$, $\alpha_3 = 0.25$, $\theta_r = 0.05$, $\theta_t = 0.1$.

Evaluation metrics. Since our method estimates the scene flow, segments moving objects and deblurs images, we thus evaluate multiple tasks separately. As for the scene flow estimation results, we evaluate both the optical flow and disparity map by the same error metric, which is by counting the number of pixels having errors more than 3 pixels and 5% of its ground-truth. We adopt the PSNR to evaluate the deblurred image sequences for left and right view separately. We report precision (P), recall (R) and F-measure (F) for our motion segmentation results. Those metrics are defined as:

$$P = \frac{t_p}{t_p + f_p}, \quad R = \frac{t_p}{t_p + f_n}, \quad F = \frac{2R * P}{R + P}, \tag{4.20}$$

where the true positive t_p represents the number of pixels that have been correctly detected as moving objects; false positive f_p are defined as pixels that have been mis-detected as moving pixels; false negative f_n are denoted as moving pixels that have not been detected correctly. Thus, we report disparity errors for three stereo image pairs for each sequence, flow errors in forward and backward directions, and PSNR values for six images, and precision, recall, and F-measure for the moving object segmentation results.

Baselines. We first compare our scene flow results with piece-wise rigid scene flow method (PRSF) Vogel et al. [2015], whose performance ranks as one of the top 3 approaches on KITTI optical flow benchmark Geiger et al.

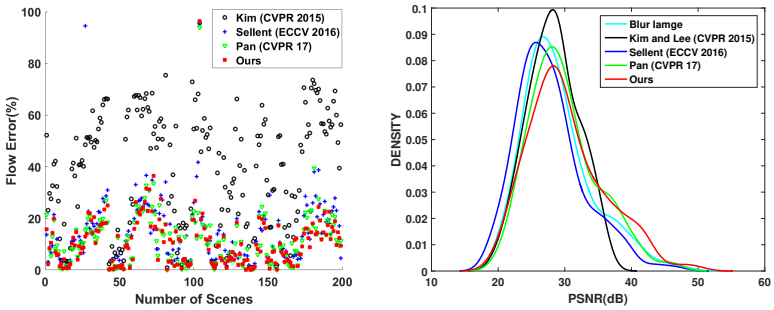


Figure 4.8: *Left: The flow estimation errors for 199 scenes in the KITTI dataset. Our method clearly outperforms the monocular and stereo video deblurring methods. Right: The distribution of the PSNR scores for 199 scenes in the KITTI dataset(BlurData-1). The probability distribution function for each PSNR was estimated using kernel density estimation with a normal kernel function. The heavy tail of our method means larger PSNR can be achieved using our method.*

[2013]. We then compare our results with the state-of-the-art stereo deblurring approach Sellent et al. [2016], monocular deblurring approach Kim and Lee [2014] and deep-learning-based deblurring approaches Tao et al. [2018]; Kupyn et al. [2018b]. We compare our moving object segmentation results with the state-of-the-art approach using sharp stereo video sequences Zhou et al. [2017]. Besides, we further choose NLC Faktor and Irani [2014] and FST Papazoglou and Ferrari [2013] as baselines since they are more robust to occlusions, motion blur and illumination changes according to the comprehensive evaluations in Perazzi et al. [2016]. We make the quantitative comparison of our model w/o explicitly imposing semantics priors for our flow and deblurring results in Fig 4.8. In addition, we compare with our previous method (Pan et al. [2017b]) that has no semantics priors. The comparison clearly shows that the performance is improved significantly with the introduction of semantics as priors.

Runtime: In all experiments, we simultaneously compute two directions, namely forward and backward, scene flows, restore six blurred images and segment all moving objects. Our MATLAB® implementation with C++ wrappers requires a total running time of 35 minutes for processing one scene (6 images, 3 iterations) on a single i7 core running at 3.6 GHz.

4.6.2 Results on KITTI

Currently, there are no realistic benchmark datasets that provide blurred images and corresponding ground-truth deblurring and scene flow to the best of our knowledge. We take advantage of the KITTI dataset Geiger et al. [2013] to create a synthetic **blurry image dataset** on realistic scenery, which con-

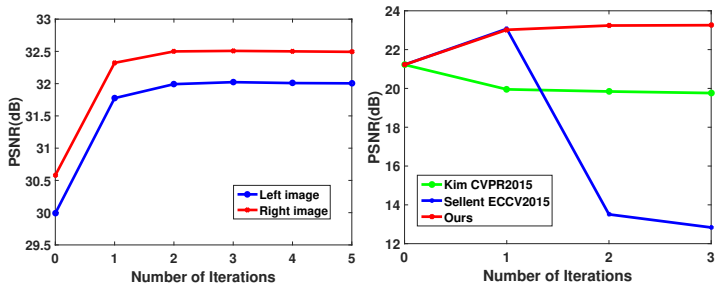
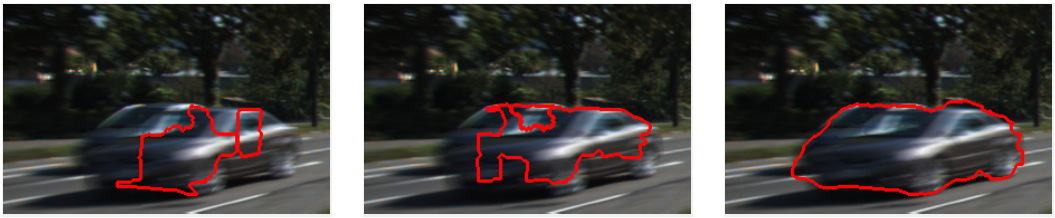


Figure 4.9: *The deblurring performance of our approach with respect to the number of iterations. (left) Our method on our dataset with the gap of 0.3 dB between the first and the last iteration. (right) Several baselines on 'Chair'.*



iterations = 1 iterations = 2 iterations = 3

Figure 4.10: *The moving object segmentation result with respect to the number of iterations*

tains 199 challenging outdoor sequences. Each sequence includes 6 images (375×1242). Our blur image dataset is generated in two different ways. First, we follow the general practice in image deblurring and generate the blur image dataset, referred to as **BlurData-1**, using the piecewise linear 2D kernel in Eq. (4.3) which is defined on the dense scene flow. We use method Menze and Geiger [2015] to generate dense ground-truth flows. In addition, $\tau = 0.23$ and the number of frame is $N = 20$ (see Fig. 4.5 for details).

Second, we follow the way of generating blurry image in Kim et al. [2018], by averaging the reference image together with its neighbouring frames. In particular, we average 7 frames in total (3 on either side of the reference frame). Note that the image sequence in KITTI, in general, has large relative

Table 4.2: *Moving object segmentation evaluation on the KITTI dataset BlurData-1.*

Methods	Recall(R)	Precision (P)	F-measure (F)
Menze and Geiger [2015]	0.7995	0.5841	0.6045
Zhou et al. [2017]	0.7641	0.6959	0.7284
Papazoglou and Ferrari [2013]	0.5945	0.3199	0.2938
Faktor and Irani [2014]	0.4761	0.3148	0.3339
Baseline	0.7633	0.6113	0.6789
Our	**0.8520**	**0.7281**	**0.7426**

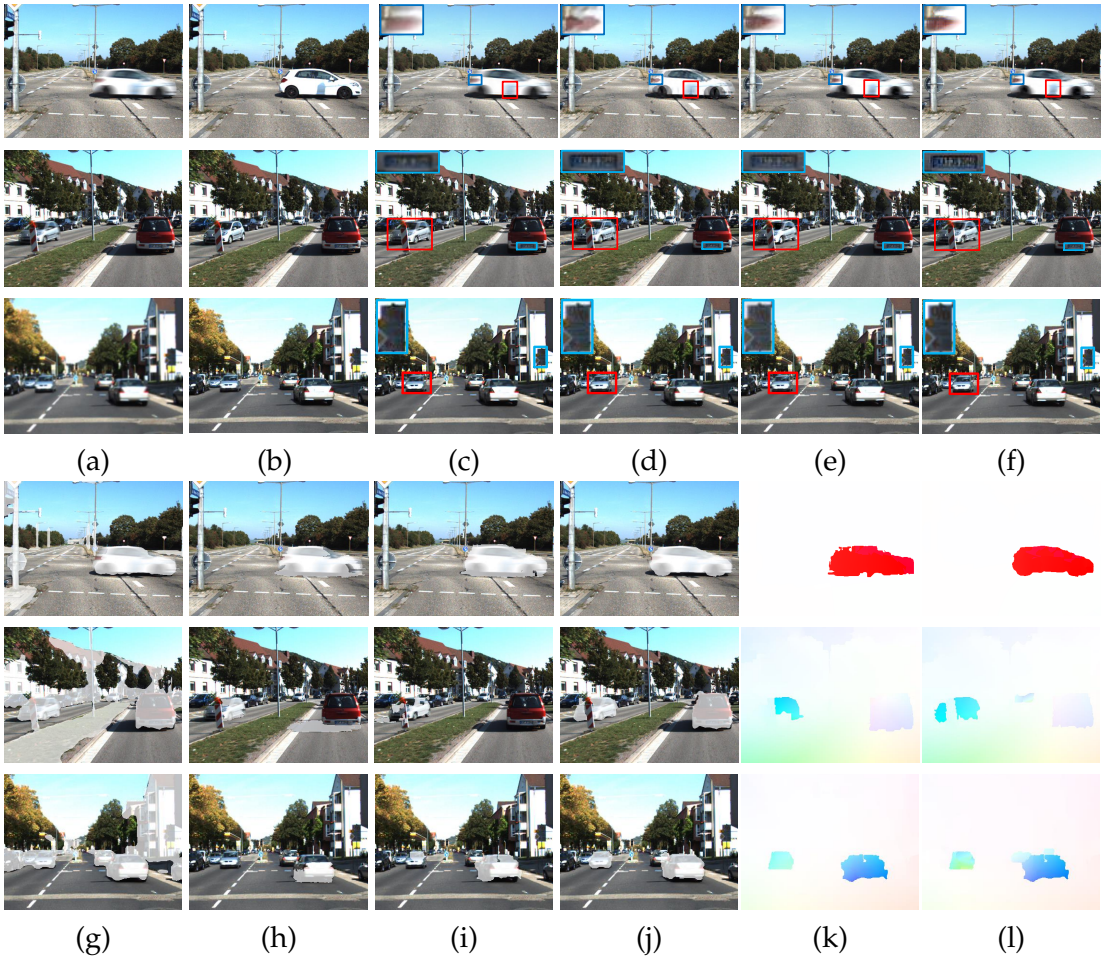
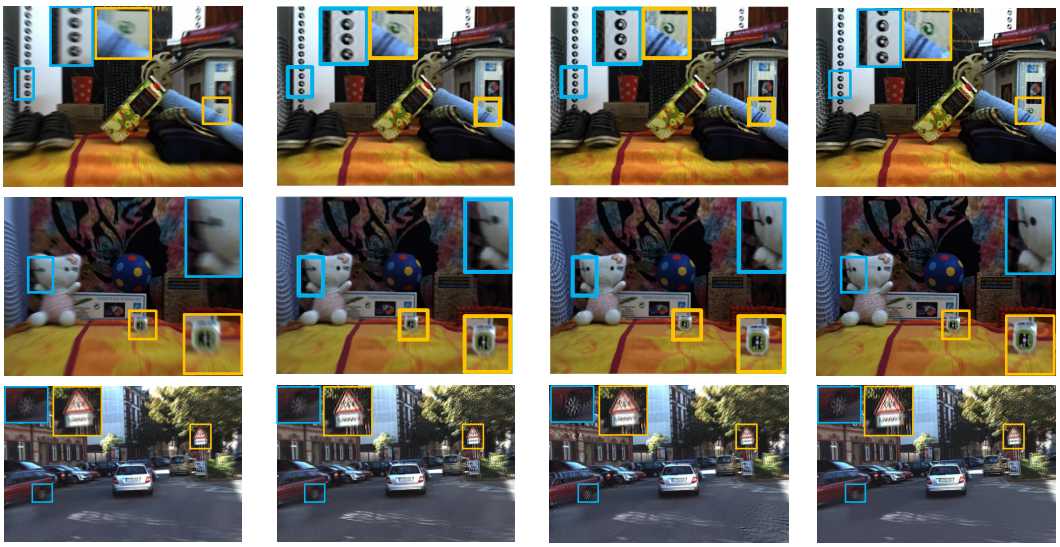
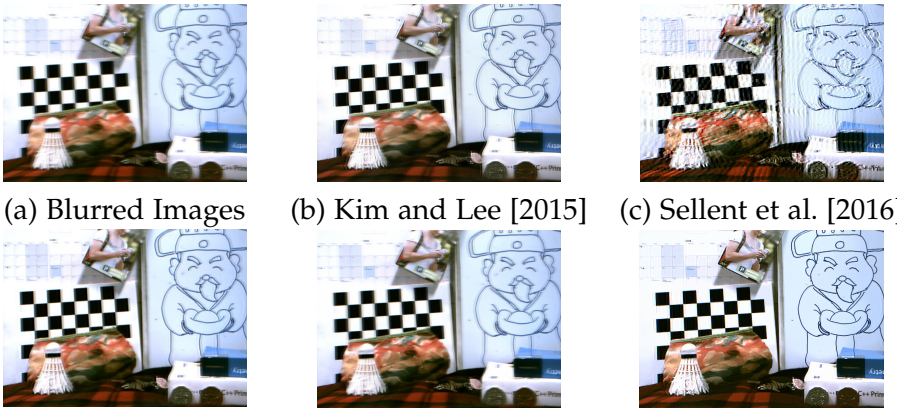


Figure 4.11: *Qualitative comparison of our approach with baselines for deblurring, moving object segmentation, and flow estimations. Our method use (a) blurred image and (g) Initial semantic prior from* **BlurData-1** *as input. (b) Ground-truth latent image. (c) Deblurring results by Kim and Lee [2015]. (d) Stereo deblurring results by Sellent et al. [2016]. (e) and (f) show our deblurring results w/o imposing semantic priors, respectively; (h) Segmentation result by Papazoglou and Ferrari [2013]. (i) Segmentation result by Faktor and Irani [2014]. (j) Our segmentation result. (k) and (l) show the optical flow estimation results w/o imposing semantic priors. Best viewed in colour on the screen.*



(a) Blurred Images (b) Kim and Lee [2015] (c) Sellent et al. [2016] (d) Our results

Figure 4.12: *Sample deblur results on the real image dataset from Sellent et al. [2016] in 1^{st} and 2^{nd} row, and average model dataset in $3^{r}d$ row. It shows that our 'generalized stereo deblur' model can tackle different kinds of motion blur model and get better results. Best viewed in colour on the screen.*



(a) Blurred Images (b) Kim and Lee [2015] (c) Sellent et al. [2016]

(d) Pan et al. [2017b] (e) Tao et al. [2018] (f) Our results

Figure 4.13: *Deblurring results on our Blur dataset. (a) The blurred image. (b) Deblurring results by Kim and Lee [2015]. (c) Stereo deblurring results by Sellent et al. [2016]. (d) Deblurring results by Pan et al. [2017b]. (e) Deblurring results by Tao et al. [2018]. (f) Our result. It shows that our 'generalized stereo deblur' model can get competitive result compared with the state-of-the-art deblurring methods results. Best viewed in colour on the screen.*

motion. We therefore only choose 10 sequences to generate blurry images based on averaging, which is denoted as **BlurData-2**. In the following, we report results on our generated two synthetic datasets, respectively.

Deblurring and Scene Flow Results. We evaluated our approach by averaging errors and PSNR scores over m and $m+1$ stereo image pairs. Table 4.1 shows the PSNR values, disparity errors, and flow errors averaged over 199 test sequences on **BlurData-1**. Note that our method consistently outperforms all baselines. We achieve the minimum error scores of 9.83% for optical flow and 6.18% for the disparity in the reference view. Figure 4.8 and Figure 4.8 show the estimated flows and deblurring results of the KITTI stereo flow benchmark, which includes 199 scenes. Figure 4.9 (left) shows the performance of our deblurring stage with respect to the number of iterations. While we use 5 iterations for all our experiments, our experiments indicate that only 3 iterations are sufficient in most cases to reach an optimal performance under our model. In Figure 4.11, we show qualitative results.

Moving Object Segmentation Results. We report the quantitative comparison of our results with the baselines in Table 4.2. It shows that our approach significantly outperforms the baselines by a large margin. Fig. 4.11(g-k) show the qualitative comparison of our approach with baselines. The results show that our final segmentation follows the boundary of the moving objects very well. It further demonstrates that our approach can segment the moving objects more accurately than other approaches. Therefore, we can achieve a conclusion that joint scene flow estimation, deblurring, and moving object segmentation benefit each task.

4.6.3 Results on Other Dataset

In order to evaluate the generalization ability of our approach on different images, we use the datasets based on the 3D kernel model and average kernel model which is different from our **Blurred image dataset**. In order to compare our performance on images blurred by the 3D kernel model, we also use the data courtesy of Sellent Sellent et al. [2016]. Those sequences contain four real and four synthetic scenes and each of them have six blurred images with its sharp images. The synthetic sequences are blurred by the 3D kernel model and have ground-truth for those sequences. Figure 4.9 (right) shows the performance of several baselines on synthetic dataset. This plot affirms our assumption that jointly and simultaneously solving scene flow and video deblur that contribute to each other. It also shows that a simple combination of two stages cannot achieve the targeted results. Real scenes use real images captured with a stereo camera that moves forward very slowly and attached to a motorized rail. By averaging the frames, they obtain motion blurred images where all objects in the scene are static and the camera moves toward

the scene. For these reasons, we give the semantic segmentation map as all background (see Figure 4.12 1^{st} and 2^{nd} rows show the performance of the result of the real scene).

In Fig. 4.12(the 3^{rd} and 4^{th} rows.), we show qualitative results of our method and other methods on sample sequences from this two datasets, where our method again achieves the best performance.

4.6.4 Limitations

Our method is based on calibrated stereo cameras which sometimes seem not convenient for routine application. The framework may fail in the texture-less case, the scene with strong reflection or under low lighting conditions. Besides, the occlusion will also reduce the accuracy of the segmentation boundaries. Our model cannot handle images with defocus blur and scenery with transparency or translucency. Following the recent deblurring works such as Kim and Lee [2014]; Gupta et al. [2010]; Dai and Wu [2008]; Whyte et al. [2012], we make the similar assumption that the intensity integral happens in colour space during the exposure time, while we are aware of several methods model the integration in the raw sensor value and consider the effects of CRFs on motion deblurring Nah et al. [2017]; Tai et al. [2013]. We leave these limitations as future works.

4.7 Conclusion

This chapter presents a joint optimization framework to tackle the challenging task of stereo video deblurring where scene flow estimation, moving object segmentation, and video deblurring are solved in a coupled manner. Under our formulation, the motion cues from scene flow estimation and blur information could reinforce each other, and produce superior results than conventional scene flow estimation or stereo deblurring methods. We have demonstrated the benefits of our framework on extensive synthetic and real stereo sequences. In future, we plan to extend our method to deal with multiple frames to achieve better stereo deblurring.

Chapter 5

Bringing a Blurry Frame Alive at High Frame-Rate with an Event Camera

Video reconstruction is another deblurring trend that reverses the blurring process by extracting a video from a single blurred image. This chapter introduces the event camera (Dynamic and Active-pixel Vision Sensor, DAVIS) to this research field. Event cameras are gaining attention, for they can measure intensity changes (called '*events*') with microsecond accuracy under high-speed motion and challenging lighting conditions. A blurred image can be regarded as the integral of a sequence of latent images, while the events indicate the changes between the latent images. Therefore, we can model the blur-generation process by associating event data to a latent image.

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. Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

5.1 Abstract

Event-based cameras can measure intensity changes (called '*events*') with microsecond accuracy under high-speed motion and challenging lighting conditions. With the active pixel sensor (APS), the event camera allows simultaneous output of the intensity frames. However, the output images are captured at a relatively low frame-rate and often suffer from motion blur. A blurry image can be regarded as the integral of a sequence of latent images, while the events indicate the changes between the latent images. Therefore, we are able to model the blur-generation process by associating event data to a latent image. In this chapter, we propose a simple and effective approach, the **Event-based Double Integral (EDI)** model, to reconstruct a high frame-rate,

sharp video from a single blurry frame and its event data. The video generation is based on solving a simple non-convex optimization problem in a single scalar variable. Experimental results on both synthetic and real images demonstrate the superiority of our EDI model and optimization method in comparison to the state-of-the-art.

5.2 Introduction

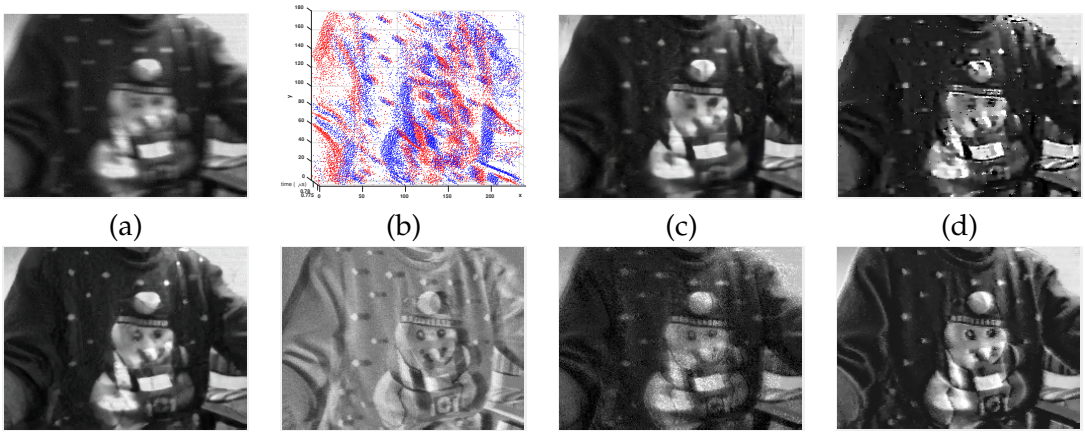


Figure 5.1: *Deblurring and reconstruction results of our method compared with the state-of-the-art methods on our real blurry event dataset. (a) The input blurry image. (b) The corresponding event data. (c) Deblurring result of Tao et al. [2018]. (d) Deblurring result of Pan et al. [2017a]. (e) Deblurring result of Jin et al. [2018]. Jin uses video as training data to train a supervised model to perform deblur, where the video can also be considered as similar information as the event data. (f)-(g) Reconstruction results of Scheerlinck et al. [2018], (f) from only events, (g) from combining events and frames. (h) Our reconstruction result. (Best viewed on screen).*

Event cameras (such as the Dynamic Vision Sensor (DVS) Lichtsteiner et al. [2008] and the Dynamic and Active-pixel Vision Sensor (DAVIS) Brandli et al. [2014a]) are sensors that asynchronously measure the intensity changes at each pixel independently with microsecond temporal resolution[1]. The event stream encodes the motion information by measuring the precise pixel-by-pixel intensity changes. Event cameras are more robust to low lighting and highly dynamic scenes than traditional cameras since they are not affected by under/over exposure or motion blur associated with a synchronous shutter.

Due to the inherent differences between event cameras and standard cameras, existing computer vision algorithms designed for standard cameras can-

[1]If nothing moves in the scene, no events are triggered.

not be applied to event cameras directly. Although the DAVIS Brandli et al. [2014a] can provide the simultaneous output of the intensity frames and the event stream, there still exist major limitations with current event cameras:

- **Low frame-rate intensity images:** In contrast to the high temporal resolution of event data ($\geq 3\mu$s frame rate), the current event cameras only output low frame-rate intensity images ($\geq 5m$s time resolution).
- **Inherent blurry effects:** When recording highly dynamic scenes, motion blur is a common issue due to the relative motion between the camera and the scene. The output of the intensity image from the APS tends to be blurry.

To address these above challenges, various methods have been proposed by reconstructing high frame-rate videos. The existing methods can be in general categorized as **1**) Event data only solutions Bardow et al. [2016]; Reinbacher et al. [2016]; Barua et al. [2016], where the results tend to lack the texture and consistency of natural videos, as they fail to use the complementary information contained in the low frame-rate intensity image; **2**) Low frame-rate intensity-image-only solutions Jin et al. [2018], where an end-to-end learning framework has been proposed to learn regression between a single blurry image and a video sequence, whereas the rich event data are not used; and **3**) Jointly exploiting event data and intensity images Scheerlinck et al. [2018]; Brandli et al. [2014b], building upon the interaction between both sources of information. However, these methods fail to address the blur issue associated with the captured image frame. Therefore, the reconstructed high frame-rate videos can be degraded by blur.

Although blurry frames cause undesired image degradation, they also encode the relative motion between the camera and the observed scene. Taking full advantage of the encoded motion information would benefit the reconstruction of high frame-rate videos.

To this end, we propose an **Event-based Double Integral (EDI)** model to resolve the above problems by reconstructing a high frame-rate video from a single image (even blur) and its event sequence, where the blur effects have been reduced in each reconstructed frame. Our EDI model naturally relates the desired high frame-rate sharp video, the captured intensity frame and event data. Based on the EDI model, high frame-rate video generation is as simple as solving a non-convex optimization problem in a single scalar variable.

Our main contributions are summarized as follows.

1) We propose a simple and effective model, named the Event-based Double Integral (EDI) model, to restore a high frame-rate sharp video from a single image (even blur) and its corresponding event data.

2) Using our formulation of EDI, we propose a stable and general method to generate a sharp video under various types of blur by solving a single variable non-convex optimization problem, especially in low lighting and complex dynamic conditions.

3) The frame rate of our reconstructed video can theoretically be as high as the event rate (200 times greater than the original frame rate in our experiments).

5.3 Related Work

Event cameras such as the DAVIS and DVS Brandli et al. [2014a]; Lichtsteiner et al. [2008] report log intensity changes, inspired by human vision. Although several works try to explore the advantages of the high temporal resolution provided by event cameras Kim et al. [2016]; Rebecq et al. [2017]; Zhu et al. [2017, 2018a]; Gehrig et al. [2018]; Kueng et al. [2016a]; Scheerlinck et al. [2019a], how to make the best use of the event camera has not yet been fully investigated.

Event-based image reconstruction. Kim et al. [2014] reconstruct high-quality images from an event camera under a strong assumption that the only movement is pure camera rotation, and later extend their work to handle 6-degree-of-freedom motion and depth estimation Kim et al. [2016]. Bardow et al. [2016] aim to simultaneously recover optical flow and intensity images. Reinbacher et al. [2016] restore intensity images via manifold regularization. Barua et al. [2016] generate image gradients by dictionary learning and obtain a logarithmic intensity image via Poisson reconstruction. However, the intensity images reconstructed by the previous approaches suffer from obvious artifacts as well as lack of texture due to the spatial sparsity of event data.

To achieve more image detail in the reconstructed images, several methods trying to combine events with intensity images have been proposed. The DAVIS Brandli et al. [2014a] uses a shared photo-sensor array to simultaneously output events (DVS) and intensity images (APS). Scheerlinck et al. [2018] propose an asynchronous event-driven complementary filter to combine APS intensity images with events, and obtain continuous-time image intensities. Brandli et al. [2014b] directly integrate events from a starting APS image frame, and each new frame resets the integration. Shedligeri and Mitra [2019] first exploit two intensity images to estimate depth. Then, they use the event data only to reconstruct a pseudo-intensity sequence (using Reinbacher et al. [2016]) between the two intensity images and use the pseudo-intensity sequence to estimate ego-motion using visual odometry. Using the estimated 6-DOF pose and depth, they directly warp the intensity image to the intermediate location. Liu et al. [2017a] assume a scene should have static

background. Thus, their method needs an extra sharp static foreground image as input and the event data are used to align the foreground with the background.

Image deblurring. Traditional deblurring methods usually make assumptions on the scenes (such as a static scene) or exploit multiple images (such as stereo, or video) to solve the deblurring problem. Significant progress has been made in the field of single image deblurring. Methods using gradient based regularizers, such as Gaussian scale mixture Fergus et al. [2006], $l_1 \backslash l_2$ norm Krishnan et al. [2011], edge-based patch priors Sun et al. [2013]; Yu et al. [2014] and l_0-norm regularizer Xu et al. [2013], have been proposed. Non-gradient-based priors such as the color line based prior Lai et al. [2015], and the extreme channel (dark/bright channel) prior Pan et al. [2017a]; Yan et al. [2017a] have also been explored. Another family of image deblurring methods tends to use multiple images Cho et al. [2012]; Kim and Lee [2015]; Sellent et al. [2016]; Pan et al. [2017b, 2018].

Driven by the success of deep neural networks, Sun et al. [2015] propose a convolutional neural network (CNN) to estimate locally linear blur kernels. Gong et al. [2017b] learn optical flow from a single blurry image through a fully-convolutional deep neural network. The blur kernel is then obtained from the estimated optical flow to restore the sharp image. Nah et al. [2017] propose a multi-scale CNN that restores latent images in an end-to-end learning manner without assuming any restricted blur kernel model. Tao et al. [2018] propose a light and compact network, SRN-DeblurNet, to deblur the image. However, deep deblurring methods generally need a large dataset to train the model and usually require sharp images provided as supervision. In practice, blurry images do not always have corresponding ground-truth sharp images.

Blurry image to sharp video. Recently, two deep learning based methods Jin et al. [2018]; Purohit et al. [2019] propose to restore a video from a single blurry image with a fixed sequence length. However, their reconstructed videos do not obey the 3D geometry of the scene and camera motion. Although deep-learning based methods achieve impressive performance in various scenarios, their success heavily depend on the consistency between the training datasets and the testing datasets, thus hinder the generalization ability for real-world applications.

5.4 Formulation

In this section, we develop an **EDI** model of the relationships between the events, the latent image and the blurry image. Our goal is to reconstruct a high frame-rate, sharp video from a single image and its corresponding

events. This model can tackle various blur types and work stably in highly dynamic contexts and low lighting conditions.

5.4.1 Event Camera Model

Event cameras are bio-inspired sensors that asynchronously report logarithmic intensity changes Brandli et al. [2014a]; Lichtsteiner et al. [2008]. Unlike conventional cameras that produce the full image at a fixed frame-rate, event cameras trigger events whenever the change in intensity at a given pixel exceeds a preset threshold. Event cameras do not suffer from the limited dynamic ranges typical of sensors with synchronous exposure time, and are able to capture high-speed motion with microsecond accuracy.

Inherent in the theory of event cameras is the concept of the latent image $\mathbf{L}_{xy}(t)$, denoting the instantaneous intensity at pixel (x, y) at time t, related to the rate of photon arrival at that pixel. The latent image $\mathbf{L}_{xy}(t)$ is not directly output by the camera. Instead, the camera outputs a sequence of *events*, denoted by (x, y, t, σ), which record changes in the intensity of the latent image. Here, (x, y) are image coordinates, t is the time the event takes place, and polarity $\sigma = \pm 1$ denotes the direction (increase or decrease) of the intensity change at that pixel and time. Polarity is given by,

$$\sigma = \mathcal{T}\left(\log\left(\frac{\mathbf{L}_{xy}(t)}{\mathbf{L}_{xy}(t_{\text{ref}})}\right), c\right), \tag{5.1}$$

where $\mathcal{T}(\cdot, \cdot)$ is a truncation function,

$$\mathcal{T}(d, c) = \begin{cases} +1, & d \geq c, \\ 0, & d \in (-c, c), \\ -1, & d \leq -c. \end{cases}$$

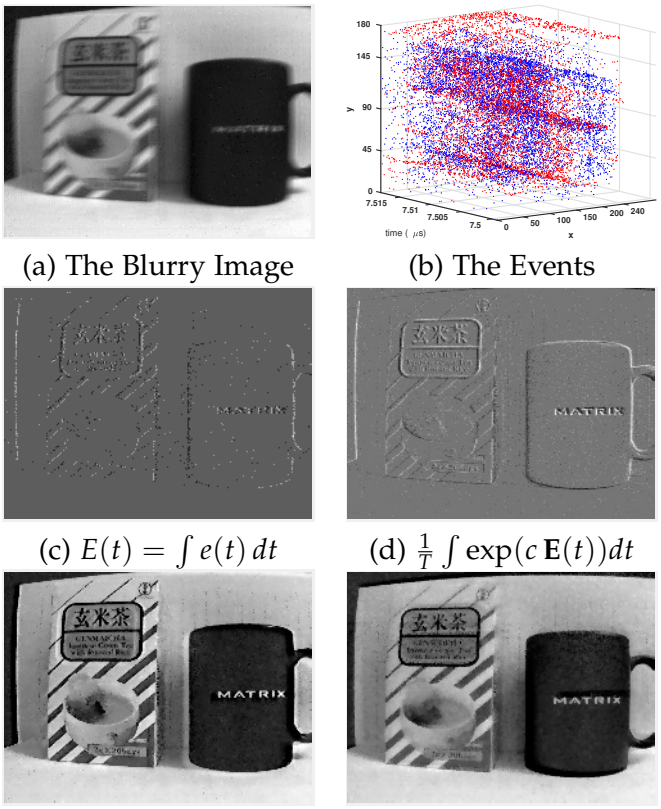
Here, c is a threshold parameter determining whether an event should be recorded or not, t_{ref} denotes the timestamp of the previous event. When an event is triggered, $\mathbf{L}_{xy}(t_{\text{ref}})$ at that pixel is updated to a new intensity level.

5.4.2 Intensity Image Formation

In addition to the event sequence, event cameras can provide a full-frame grey-scale intensity image, at a much slower rate than the event sequence. The grey-scale images may suffer from motion blur due to their long exposure time. A general model of image formation is given by,

$$\mathbf{B} = \frac{1}{T}\int_{f-T/2}^{f+T/2} \mathbf{L}(t)\, dt, \tag{5.2}$$

where $\mathbf{B}$ is a blurry image, equal to the average value of latent images during the exposure time $[f - T/2, f + T/2]$.



(a) The Blurry Image (b) The Events

(c) $E(t) = \int e(t)\, dt$ (d) $\frac{1}{T} \int \exp(c\, \mathbf{E}(t))dt$

(e) Sample Frames of Our Reconstructed Video

Figure 5.2: *The event data and our reconstructed result, where (a) and (b) are the input of our method. (a) The intensity image from the event camera. (b) Events from the event camera plotted in 3D space-time (x, y, t) (blue: positive event; red: negative event). (c) The first integral of several events during a small time interval. (d) The second integral of events during the exposure time. (e) Samples from our reconstructed video from* $\mathbf{L}(0)$ *to* $\mathbf{L}(200)$.

5.4.3 Event-based Double Integral Model

We aim to recover a sequence of latent intensity images by exploiting both the blur model and the event model. We define $e_{xy}(t)$ as a function of continuous time t such that

$$e_{xy}(t) = \sigma\ \delta_{t_0}(t),$$

whenever there is an event (x, y, t_0, σ). Here, $\delta_{t_0}(t)$ is an impulse function, with unit integral, at time t_0, and the sequence of events is turned into a continuous time signal, consisting of a sequence of impulses. There is such a function $e_{xy}(t)$ for every point (x, y) in the image. Since each pixel can be treated separately, we omit the subscripts x, y.

During an exposure period $[f - T/2, f + T/2]$, we define $\mathbf{E}(t)$ as the sum

of events between time f and t at a given pixel,

$$\mathbf{E}(t) = \int_f^t e(s)ds,$$

which represents the proportional change in intensity between time f and t. Except under extreme conditions, such as glare and no-light conditions, the latent image sequence $\mathbf{L}(t)$ is expressed as,

$$\mathbf{L}(t) \ = \mathbf{L}(f)\, \exp(c\, \mathbf{E}(t)) = \mathbf{L}(f)\, \exp(c)^{\mathbf{E}(t)}\ . \tag{5.3}$$

We put a tilde on top of things to denote logarithm, e.g., $\widetilde{\mathbf{L}}(t) = \log(\mathbf{L}(t))$.

$$\widetilde{\mathbf{L}}(t) \ = \widetilde{\mathbf{L}}(f) + c\, \mathbf{E}(t). \tag{5.4}$$

Given a sharp frame, we can reconstruct a high frame-rate video from the sharp starting point $\mathbf{L}(f)$ by using Eq. (5.4). When the input image is blurry, a trivial solution would be to first deblur the image with an existing deblurring method and then to reconstruct the video using Eq. (5.4) (see Fig.5.6 for details). However, in this way, the event data between intensity images is not fully exploited, thus resulting in inferior performance. Instead, we propose to reconstruct the video by exploiting the inherent connection between event and blur, and present the following model.

As for the blurred image,

$$\mathbf{B} = \frac{1}{T}\int_{f-T/2}^{f+T/2} \mathbf{L}(t)dt = \frac{\mathbf{L}(f)}{T}\int_{f-T/2}^{f+T/2} \exp\left(c\int_f^t e(s)ds\right)\, dt\ . \tag{5.5}$$

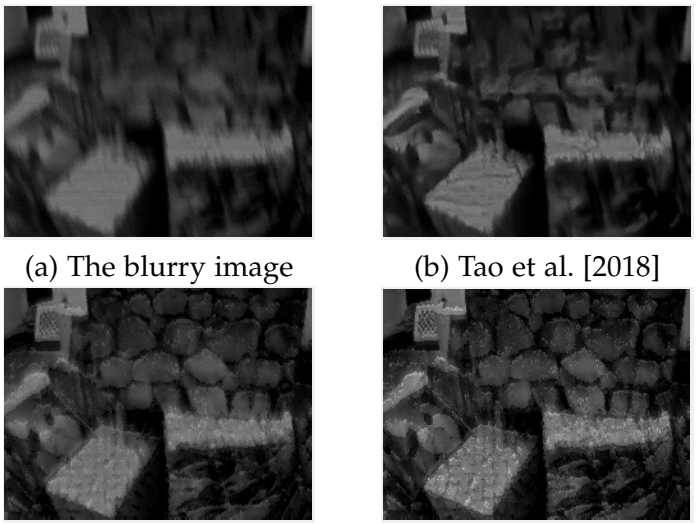
In this manner, we construct the relation between the captured blurry image $\mathbf{B}$ and the latent image $\mathbf{L}(f)$ through the double integral of the event. We name Eq. (5.5) the **Event-based Double Integral (EDI)** model. Taking the logarithm on both sides of Eq. (5.5) and rearranging, yields

$$\widetilde{\mathbf{L}}(f) = \widetilde{\mathbf{B}} - \log\left(\frac{1}{T}\int_{f-T/2}^{f+T/2} \exp(c\, \mathbf{E}(t))dt\right), \tag{5.6}$$

which shows a linear relation between the blurry image, the latent image and the integral of the events in the log space.

5.4.4 High Frame-Rate Video Generation

The right-hand side of Eq. (5.6) is known, apart from perhaps the value of the contrast threshold c, the first term from the grey-scale image, the second term from the event sequence, it is possible to compute $\widetilde{\mathbf{L}}(f)$, and hence $\mathbf{L}(f)$



(a) The blurry image (b) Tao et al. [2018]

(c) By human observation (d) By energy minimization

Figure 5.3: *An example of our reconstruction result using different methods to estimate c, from the real dataset Mueggler et al. [2017]. (a) The blurry image. (b) Deblurring result of Tao et al. [2018] (c) Our result where c is chosen by manual inspection. (d) Our result where c is computed automatically by our proposed energy minimization* (5.9).

by exponentiation. Subsequently, from Eq. (5.4) the latent image $\mathbf{L}(t)$ at any time may be computed.

To avoid accumulated errors of constructing a video from many frames of a blurred video, it is more suitable to construct each frame $\mathbf{L}(t)$ using the closest blurred frame.

Theoretically, we could generate a video with frame-rate as high as the DVS's eps (events per second). However, as each event carries little information and is subject to noise, several events must be processed together to yield a reasonable image. We generate a reconstructed frame every 50-100 events, so for our experiments, the frame-rate of the reconstructed video is usually 200 times greater than the input low frame-rate video. Furthermore, as indicated by Eq. (5.6), the challenging blind motion deblurring problem has been reduced to a single variable optimization problem of how to find the best value of the contrast threshold c. In the following section, we use $\mathbf{L}(c, t)$ to present the latent sharp image $\mathbf{L}(t)$ with different c.

5.5 Optimization

The unknown contrast threshold c represents the minimum change in log intensity required to trigger an event. By choosing an appropriate c in Eq. (5.5), we can generate a sequence of sharper images. To this end, we first need to

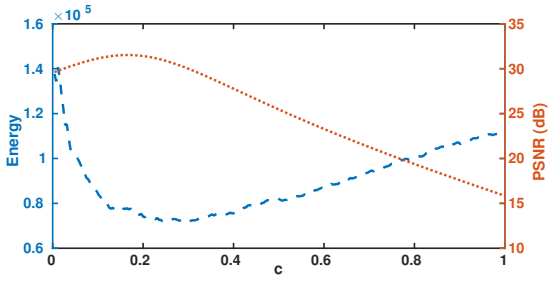


Figure 5.4: *The figure plot deblurring performance against the value of c. The image is clearer with higher PSNR value.*

evaluate the sharpness of the reconstructed images. Here, we propose two different methods to estimate the unknown variable c: manually chosen and automatically optimized.

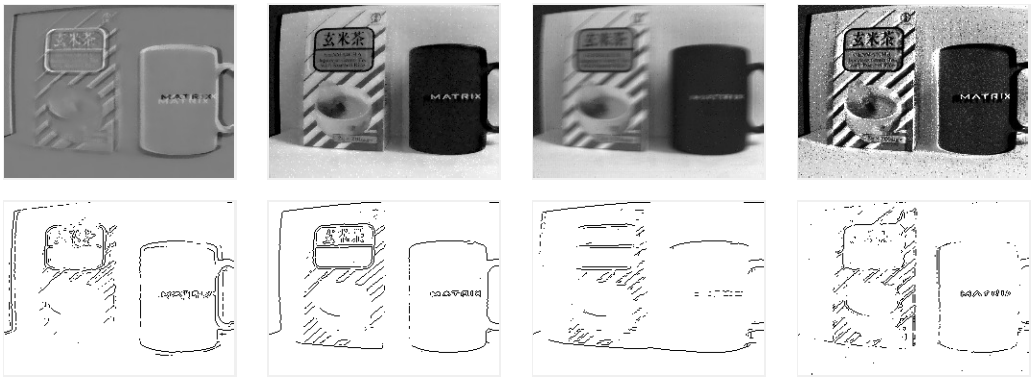
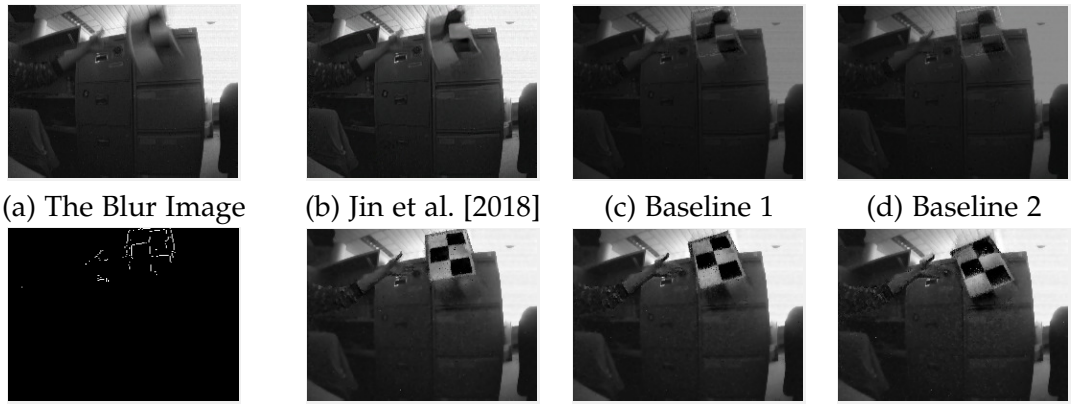


Figure 5.5: *At left, the edge image $M(f)$ and below, its Sobel edge map. To the right are 3 reconstructed latent images using different values of c, low 0.03, middle 0.11 and high 0.55. Above, the reconstructed images, below, their Sobel edge maps. The optimal value of the threshold c is found by computing the cross-correlation of such images with the edge map at the left. (Best viewed on screen).*

5.5.1 Manually Chosen c

According to our **EDI** model in Eq. (5.5), given a value for c, we can obtain a sharp image. Therefore, we develop a method for deblurring by manually inspecting the visual effect of the deblurred image. In this way, we incorporate human perception into the reconstruction loop and the deblurred images should satisfy human observation. In Fig. 5.3, we give an example for manually chosen and automatically optimized results on dataset from Mueggler et al. [2017].



(a) The Blur Image (b) Jin et al. [2018] (c) Baseline 1 (d) Baseline 2

(e) $\mathcal{S}(\mathbf{L}(c,t)) \cdot \mathcal{S}(\mathbf{M}(t))$ (f) Samples of Our Reconstructed Video

Figure 5.6: *Deblurring and reconstruction results on our real blurry event dataset. (a) Input blurry images. (b) Deblurring result of Jin et al. [2018]. (c) Baseline 1 for our method. We first use the state-of-the-art video-based deblurring method Jin et al. [2018] to recover a sharp image. Then use the sharp image as input to a state-of-the-art reconstruction method Scheerlinck et al. [2018] to get the intensity image. (d) Baseline 2 for our method. We first use method Scheerlinck et al. [2018] to reconstruct an intensity image. Then use a deblurring method Jin et al. [2018] to recover a sharp image. (e) The cross-correlation between* $\mathcal{S}(\mathbf{L}(c,t))$ *and* $\mathcal{S}(\mathbf{M}(t))$*. (f) Samples from our reconstructed video from* $\mathbf{L}(0)$ *to* $\mathbf{L}(150)$*. (Best viewed on screen).*

5.5.2 Automatically Chosen c

To automatically find the best c, we need to build an evaluation metric (energy function) that can evaluate the quality of the deblurred image $\mathbf{L}(c,t)$. Specifically, we propose to exploit different prior knowledge for sharp images and the event data.

5.5.2.1 Edge Constraint for Event Data

As mentioned before, when a proper c is given, our reconstructed image $\mathbf{L}(c,t)$ will contain much sharper edges compared with the original input intensity image. Furthermore, event cameras inherently yield responses at moving intensity boundaries, so edges in the latent image may be located where (and when) events occur. This allows us to find latent image edges. An edge at time t corresponds to an event (at the pixel in question) during some time interval around t so we convolve the event sequence with an exponentially decaying window, to obtain a denoised edge boundary,

$$\mathbf{M}(t) = \int_{-T/2}^{T/2} \exp(-(\alpha|t-s|))\, e(s)\, ds,$$

where α is a weight parameter for time attenuation, and is set to 1.0. Then, we use the Sobel filter $\mathcal{S}$ to get a sharper binary edge map, which is also applied to $\mathbf{L}(c,t)$. (See Fig. 5.5 and 5.6 for details).

Here, we use cross-correlation between $\mathcal{S}(\mathbf{L}(c,t))$ and $\mathcal{S}(\mathbf{M}(t))$ to evaluate the sharpness of $\mathbf{L}(c,t)$.

$$\phi_{\text{edge}}(c) = \sum_{x,y} \mathcal{S}(\mathbf{L}(c,t))(x,y) \cdot \mathcal{S}(\mathbf{M}(t))(x,y) \; . \tag{5.7}$$

5.5.2.2 Regularizing the Intensity Image

In our model, total variation is used to suppress noise in the latent image while preserving edges, and penalize the spatial fluctuations Rudin et al. [1992].

$$\phi_{\text{TV}}(c) = |\nabla \mathbf{L}(c,t)|_1, \tag{5.8}$$

where ∇ represents the gradient operators.

5.5.2.3 Energy Minimization

The optimal c can be estimate by solving Eq. (5.9),

$$\min_c \phi_{\text{TV}}(c) + \lambda \phi_{\text{edge}}(c), \tag{5.9}$$

where λ is a trade-off parameter. The response of cross-correlation reflect the matching rate of $\mathbf{L}(c,t)$ and $\mathbf{M}(t)$ which makes $\lambda < 0$. This single-variable minimization problem can be solved by the nonlinear least-squares method Moré [1978], Scatter-search Ugray et al. [2007] or Fibonacci search Dunlap [1997].

In Fig. 5.4, we illustrate the clearness of the reconstructed image against the value of c. Meanwhile, we also provide the PSNR of the corresponding reconstructed image. As demonstrated in the figure, our proposed reconstruction metric could locate/identify the best deblurred image with peak PSNR properly.

5.6 Experiment

5.6.1 Experimental Setup

Synthetic dataset. In order to provide a quantitative comparison, we build a synthetic dataset based on the GoPro blurry dataset Nah et al. [2017]. It supplies ground truth videos which are used to generate the blurry images. Similarly, we employ the ground-truth images to generate event data based on the methodology of *event camera model*.

Table 5.1: *Quantitative comparisons with Pan et al. [2017a]; Sun et al. [2015]; Gong et al. [2017b]; Jin et al. [2018]; Tao et al. [2018]; Zhang et al. [2018]; Nah et al. [2017]; Scheerlinck et al. [2018] on the Synthetic dataset Nah et al. [2017]. This dataset provides videos can be used to generate not only blurry images but also event data. All methods are tested under the same blurry condition, where methods Nah et al. [2017]; Jin et al. [2018]; Tao et al. [2018]; Zhang et al. [2018] use GoPro dataset Nah et al. [2017] to train their models. Jin et al. [2018] achieves their best performance when the image is down-sampled to 45% mentioned in their paper.*

Average result of the deblurred images on dataset Nah et al. [2017]								
	Pan	Sun	Gong	Jin	Tao	Zhang	Nah	Ours
PSNR(dB)	23.50	25.30	26.05	26.98	**30.26**	29.18	29.08	29.06
SSIM	0.8336	0.8511	0.8632	0.8922	0.9342	0.9306	0.9135	**0.9430**
Average result of the reconstructed videos on dataset Nah et al. [2017]								
	Baseline 1		Baseline 2		Scheerlinck		Jin	Ours
PSNR(dB)	25.52		26.34		25.84		25.62	**28.49**
SSIM	0.7685		0.8090		0.7904		0.8556	**0.9199**

Real dataset. We evaluate our method on a public Event-Camera dataset Mueggler et al. [2017], which provides a collection of sequences captured by the event camera for high-speed robotics. Furthermore, we present our real *blurry event dataset* [2], where each real sequence is captured with the DAVIS Brandli et al. [2014a] under different conditions, such as indoor, outdoor scenery, low lighting conditions, and different motion patterns (e.g., camera shake, objects motion) that naturally introduce motion blur into the APS intensity images.

Implementation details. For all our real experiments, we use the DAVIS that shares photosensor array to simultaneously output events (DVS) and intensity images (APS). The framework is implemented by using MATLAB®. It takes around 1.5 second to process one image on a single i7 core running at 3.6 GHz.

5.6.2 Experimental Results

We compare our proposed approach with state-of-the-art blind deblurring methods, including conventional deblurring methods Pan et al. [2017a]; Yan et al. [2017a], deep based dynamic scene deblurring methods Nah et al. [2017]; Jin et al. [2018]; Tao et al. [2018]; Zhang et al. [2018]; Sun et al. [2015], and event-based image reconstruction methods Reinbacher et al. [2016]; Scheerlinck et al. [2018]. Moreover, Jin et al. [2018] can restore a video from a single blurry image based on a deep network, where the middle frame in the restored odd-numbered sequence is the best.

[2]To be released with codes

Figure 5.7: *An example of the reconstructed result on our synthetic event dataset based on the GoPro dataset Nah et al. [2017]. Nah et al. [2017] provides videos to generate the blurry images and event data. (a) The blurry image. The red close-up frame is for (b)-(e), the yellow close-up frame is for (f)-(g). (b) The deblurring result of Jin et al. [2018]. (c) Our deblurring result. (d) The crop of their reconstructed images and the frame number is fixed at 7. Jin et al. [2018] uses the GoPro dataset added with 20 scenes as training data and their model is supervised by 7 consecutive sharp frames. (e) The crop of our reconstructed images. (f) The crop of Reinbacher Reinbacher et al. [2016] reconstructed images from only events. (g) The crop of Scheerlinck Scheerlinck et al. [2018] reconstructed image, they use both events and the intensity image. For (e)-(g), the shown frames are the chosen examples, where the length of the reconstructed video is based on the number of events.*

In order to prove the effectiveness of our **EDI** model, we show some baseline comparisons in Fig. 5.6 and Table 5.1. For baseline 1, we first apply a state-of-the-art deblurring method Tao et al. [2018] to recover a sharp image, and then the recovered image as an input is then fed to a reconstruction method Scheerlinck et al. [2018]. For baseline 2, we first use the video reconstruction method to construct a sequence of intensity images, and then apply the deblurring method to each frame. As seen in Table 5.1, our approach obtains higher PSNR and SSIM in comparison to both baseline 1 and baseline 2. This also implies that our approach better exploits the event data to not only recover sharp images but also reconstruct high frame-rate videos.

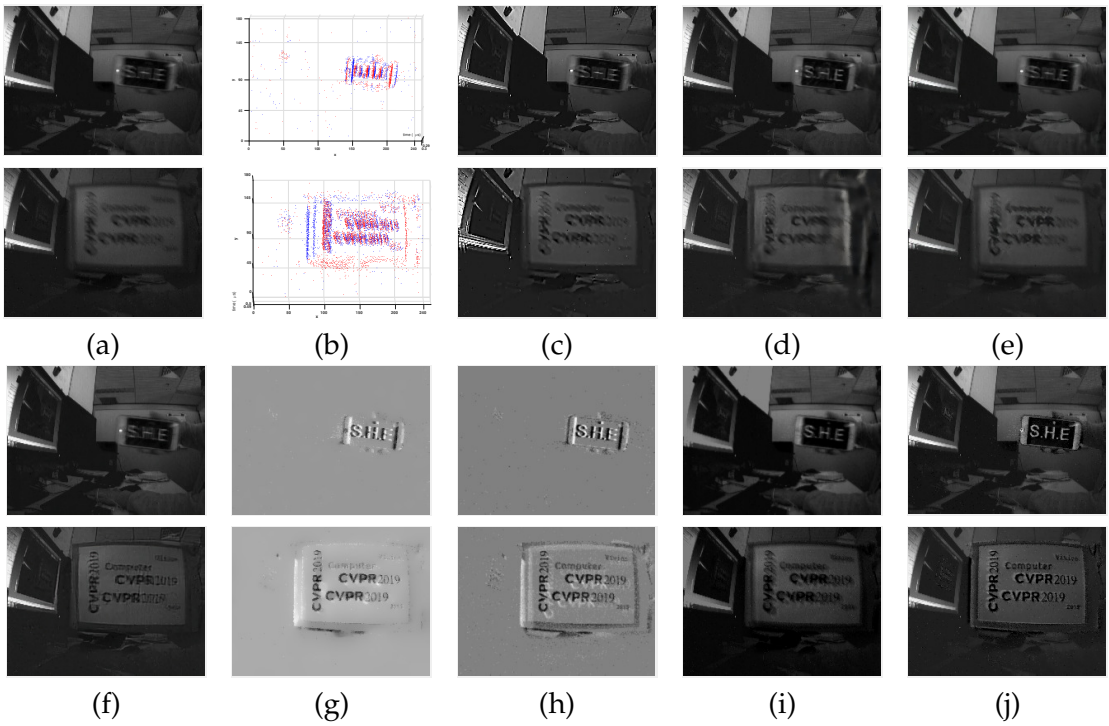


Figure 5.8: *Examples of reconstruction result on our real blurry event dataset in low lighting and complex dynamic conditions (a) Input blurry images. (b) The event information. (c) Deblurring results of Pan et al. [2017a]. (d) Deblurring results of Tao et al. [2018]. (e) Deblurring results of Nah et al. [2017]. (f) Deblurring results of Jin et al. [2018] and they use video as training data. (g) Reconstruction result of Reinbacher et al. [2016] from only events. (h)-(i) Reconstruction results of Scheerlinck et al. [2018], (h) from only events, (i) from combining events and frames. (j) Our reconstruction result. Results in (c)-(f) show that real high dynamic settings and low light condition is still challenging in the deblurring area. Results in (g)-(h) show that while intensity information of a scene is still retained with an event camera recording, color, and delicate texture information cannot be recovered.*

In Table 5.1 and Fig. 5.7, we show the quantitative and qualitative comparisons with the state-of-the-art image deblurring approaches Sun et al. [2015]; Pan et al. [2017a]; Gong et al. [2017b]; Jin et al. [2018]; Tao et al. [2018]; Zhang et al. [2018]; Nah et al. [2017], and the video reconstruction method Scheerlinck et al. [2018] on our synthetic dataset, respectively. As indicated in Table 5.1, our approach achieves the best performance on SSIM and competitive result on PSNR compared to the state-of-the-art methods, and attains significant performance improvements on high-frame video reconstruction.

We report our reconstruction results on the real dataset, including text images and low-lighting images, in Fig. 5.1, Fig. 5.2, Fig. 5.3 and Fig. 5.8. In comparison to existing event-based image reconstructed methods Reinbacher et al. [2016]; Scheerlinck et al. [2018], our reconstructed images are not only

more realistic but also contain richer details. More deblurring results are shown in Fig. 5.9

5.7 Conclusion

In this chapter, we propose an **Event-based Double Integral (EDI)** model to naturally connect intensity images and event data captured by the event camera, which also takes the blur generation process into account. In this way, our model can be used to not only recover latent sharp images but also reconstruct intermediate frames at high frame-rate. We also propose a simple yet effective method to solve our EDI model. Due to the simplicity of our optimization process, our method is efficient as well. Extensive experiments show that our method can generate high-quality high frame-rate videos efficiently under different conditions, such as low lighting and complex dynamic scenes.

(a) Sharp images (b) Blurry images (c) Jin et al. [2018] (d) Pan et al. [2017a]

(e) Yan et al. [2017a] (f) Tao et al. [2018] (g) Nah et al. [2017] (h) Ours

Figure 5.9: Examples of deblurring results on our synthetic event dataset. (a) Sharp images. (b) Generated blurry images. (c) Deblurring results of Jin et al. [2018]. (d) Deblurring results of Pan et al. [2017a]. (e) Deblurring results of Yan et al. [2017a]. (f) Deblurring results of Tao et al. [2018]. (g) Deblurring results of Nah et al. [2017]. (h) Our deblurring results. (Best view in color on screen).

Chapter 6

High Frame Rate Video Reconstruction based on an Event Camera

In this chapter, we improved the EDI model (in chapter 5) to the multiple Event-based Double Integral model by using multiple images and their events to handle flickering effects and noise in the generated video. Also, we provide a more efficient solver to minimize the proposed energy model.

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 (TPAMI), 2020.

6.1 Abstract

Event-based cameras measure intensity changes (called '*events*') with microsecond accuracy under high-speed motion and challenging lighting conditions. With the 'active pixel sensor' (APS), the 'Dynamic and Active-pixel Vision Sensor' (DAVIS) allows the simultaneous output of intensity frames and events. However, the output images are captured at a relatively low frame rate and often suffer from motion blur. A blurred image can be regarded as the integral of a sequence of latent images, while events indicate changes between the latent images. Thus, we are able to model the blur-generation process by associating event data to a latent sharp image. Based on the abundant event data alongside a low frame rate, easily blurred images, we propose a simple yet effective approach to reconstruct high-quality and high frame rate sharp videos. Starting with a single blurred frame and its event data from DAVIS, we propose the **Event-based Double Integral (EDI)** model and solve it by adding regularization terms. Then, we extend it to **multiple Event-based Double Integral (mEDI)** model to get more smooth results based on

multiple images and their events. Furthermore, we provide a new and more efficient solver to minimize the proposed energy model. By optimizing the energy function, we achieve significant improvements in removing blur and the reconstruction of a high temporal resolution video. The video generation is based on solving a simple non-convex optimization problem in a single scalar variable. Experimental results on both synthetic and real datasets demonstrate the superiority of our **mEDI** model and optimization method compared to the state-of-the-art.

6.2 Introduction

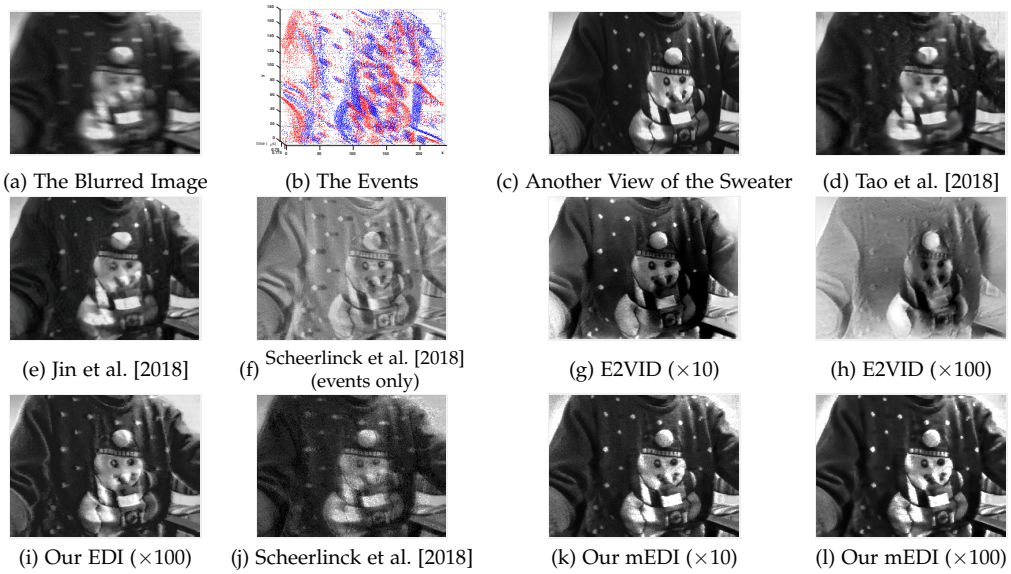
Event cameras (such as the Dynamic Vision Sensor (DVS) Lichtsteiner et al. [2008], and the DAVIS Brandli et al. [2014a]) are sensors that asynchronously measure intensity changes at each pixel independently with microsecond temporal resolution (if nothing moves in the scene, no events are triggered). The event stream encodes the motion information by measuring the precise pixel-by-pixel intensity changes. Event cameras are more robust to low lighting and highly dynamic scenes than traditional cameras since they are not affected by under/overexposure associated with a synchronous shutter.

Due to the inherent differences between event cameras and standard cameras, existing computer vision algorithms designed for standard cameras cannot be applied to event cameras directly. Although the DAVIS Brandli et al. [2014a] can provide simultaneous output of intensity frames and events, there still exist major limitations with current DAVIS cameras:

- **Low frame rate intensity images:** In contrast to the high temporal resolution of event data ($\geq 3\mu$s frame rate), the current DAVIS only output low frame rate intensity images ($\geq 20ms$ temporal resolution).
- **Inherent blur effects:** When recording highly dynamic scenes, motion blur is a common issue due to the relative motion between the camera and the scene. The output of the intensity image from the APS tends to be blurry.

To address these challenges, various methods have been proposed by reconstructing high frame rate videos. Existing methods can be, in general, categorized as:

1) Event-only solutions, Bardow et al. [2016]; Rebecq et al. [2019]; Wang et al. [2019]; Scheerlinck et al. [2019a], where the results tend to lack the texture and consistency of natural videos (especially for scenes with a static background or a slowly moving background/foreground), as they fail to use the complementary information contained in low frame rate intensity images;



(a) The Blurred Image (b) The Events (c) Another View of the Sweater (d) Tao et al. [2018]

(e) Jin et al. [2018] (f) Scheerlinck et al. [2018] (events only) (g) E2VID (×10) (h) E2VID (×100)

(i) Our EDI (×100) (j) Scheerlinck et al. [2018] (k) Our mEDI (×10) (l) Our mEDI (×100)

Figure 6.1: *Deblurring and reconstruction results of our method compared with the state-of-the-art methods on our real blur event dataset. (a) The input blurred image. (b) The corresponding event data. (c) A sharp image for the sweater captured as a reference for colour and shape (a real blurred image can hardly have its ground truth sharp image). (d) Deblurring result of Tao et al. [2018]. (e) Deblurring result of Jin et al. [2018]. Jin uses video as training data to train a supervised model to perform deblur, where the video can also be considered as similar information as the event data. (f) Reconstruction results of Scheerlinck et al. [2018] from only events. (g) Reconstruction results of Rebecq et al. [2019] from only events. Based on their default settings, the time resolution of the reconstructed video is around* $\times 10$ *times higher than the time resolution of the original video. (h) Reconstruction results of Rebecq et al. [2019] from only events. The time resolution here is around* $\times 100$*. (i) Reconstruction result of Pan et al. [2019c] from combining events and a single blurred frame. (j)Reconstruction results of Scheerlinck et al. [2018] from events and images. (k)-(l) Our reconstruction result from combining events and multiple blurred frames at different time resolution. Our result preserves more abundant and faithful texture and the consistency of the natural image. (Best viewed on screen).*

2) Events and intensity images combined solutions, Scheerlinck et al. [2018]; Brandli et al. [2014b], which build upon the interaction between both sources of information. However, these methods fail to address the blur issue associated with the captured image frame. Therefore, the reconstructed high frame rate videos can be degraded by blur.

Contrary to existing 'image + event' based methods that ignore the blur effect in the image, or discard it entirely, we give an alternative insight into the problem. While blurred frames cause undesired image degradation, they inherently encode the relative motion between the camera and the observed scene, and the integration of multiple images during the exposure time. Tak-

ing full advantage of the encoded information in the blurred image would benefit the reconstruction of high frame rate videos.

To tackle above problems, in our previous work Pan et al. [2019c], we propose an **Event-based Double Integral (EDI)** model to fuse an image (even with blur) with its event sequence to reconstruct a high frame rate, blur-free video. Our **EDI** model naturally relates the desired high frame rate sharp video, the captured intensity frame and event data. Based on the **EDI** model, high frame rate video generation is as simple as solving a non-convex optimization problem in a single scalar variable.

As the **EDI** model is based on a single image, noise from the event data can easily degrade the quality of reconstructed videos, especially at transitions between images. To mitigate accumulated noise from events, we limit the integration to a small time interval around the centre of the exposure time, allowing us to reconstruct a small video segment associated with one image. The final video is obtained by stitching all the video segments together. However, this still results in flickering, especially when the camera and objects have larger relative motion. In addition, the regularization terms (with extra weight parameters) are included in the energy function when solving the contrast threshold for our EDI model. Thus, we extended our **EDI** model to a **multiple Event-based Double Integral (mEDI)** one to handle discontinuities at the boundaries of reconstructed video segments and develop a simple yet effective optimization solution. Later in our experiments, it shows the significant improvement in the smoothness and quality of the reconstructed videos.

In this chapter, we first introduce our previous approach (the EDI model) in Sec. 6.4. Then, we build an extension framework based on multiple images and describe the approach in Sec. 6.5. Jointly optimizing *multi-frames for generating long video sequences* significantly alleviates the flickering problem for the generated videos, whereas EDI treats each image individually and may suffer flicking artefacts.

The extensions are as follows:

1) We propose a **multiple Event-based Double Integral (mEDI)** model to restore better high frame rate sharp videos. The model is based on multiple images (even blurred) and their corresponding events.

2) Our **mEDI** is able to generate a sharp video under various types of blur by solving a single variable non-convex optimization problem, especially in low lighting condition and complex dynamic scene.

3) We develop a simple yet effective optimization solution. In doing so, we significantly reduce the computational complexity with the Fibonacci sequence.

4) The frame rate of our reconstructed video can theoretically be as high as the event rate (200 times greater than the original frame rate in our experiment). With multiple images, the reconstructed videos preserve more abundant texture and the consistency of natural images.

6.3 Related Work

Event cameras such as the DAVIS and DVS Brandli et al. [2014a]; Lichtsteiner et al. [2008] report log intensity changes, inspired by human vision. The result is a continuous, asynchronous stream of events that encodes non-redundant information about local brightness change. Estimating intensity images from events is important. The reconstructed images grant computer vision researchers a readily available high temporal resolution, high-dynamic-range imaging platform that can be used for tasks such as face-detection Barua et al. [2016], moving object segmentation Stoffregen et al. [2019], SLAM Cook et al. [2011]; Kim et al. [2014, 2016]; Rebecq et al. [2017]; Vidal et al. [2018], localization Liu et al. [2017b, 2019a] and optical flow estimation Zhu et al. [2018a]; Gehrig et al. [2019a]; Stoffregen et al. [2020]; Pan et al. [2020b]. Although several works try to explore the advantages of the high temporal resolution provided by event cameras Zhu et al. [2017]; Gehrig et al. [2018]; Kueng et al. [2016a]; Gallego et al. [2019]; Brandli et al. [2014c], how to make the best use of the event camera has not yet been fully investigated. In this section, we review image reconstruction from event-based methods, and images and event combined methods. We further discuss works on image deblurring.

Event-based image reconstruction. A typical way is done by processing a spatio-temporal window of events. Taking a spatio-temporal window of events imposes a latency cost at minimum equal to the length of the time window, and choosing a time-interval (or event batch size) that works robustly for all types of scenes is not trivial. Barua et al. [2016] generate image gradients by dictionary learning and obtain a logarithmic intensity image via Poisson reconstruction. Bardow et al. [2016] simultaneously optimise optical flow and intensity estimates within a fixed-length, sliding spatio-temporal window using the primal-dual algorithm Posch et al. [2010]. Cook et al. [2011] integrate events into interacting maps to recover intensity, gradient, and optical flow while estimating global rotating camera motion. Kim et al. [2014] reconstruct high-quality images from an event camera under a strong assumption that the only movement is pure camera rotation, and later extend their work to handle 6-degree-of-freedom motion and depth estimation Kim et al. [2016]. Reinbacher et al. [2016] integrate events over time while periodically regularising the estimate on a manifold defined by the timestamps of the latest events at each pixel. Optimisation based event-only methods (i.e., without the process of learning from training data) will generate artefacts and lack of texture

when event data is sparse, because they cannot integrate sufficient information from the available sparse events. Recently, learning-based approaches have improved the image reconstruction quality significantly with powerful event data representations via deep learning Rebecq et al. [2019, 2020]; Wang et al. [2019]; Scheerlinck et al. [2020]. Rebecq et al. [2019] propose E2VID, a fully convolutional, recurrent UNet architecture to encode events in a spatio-temporal voxel grid. In Rebecq et al. [2020], they propose a recurrent network to reconstruct videos from a stream of events and incorporate stacked ConvLSTM gates, which prevent vanishing gradients during backpropagation for long sequences. Wang et al. [2019] form a 3D event volume by stacking event frame in a time interval. A reconstructed intensity frame is generated by summing events at each pixel in a smaller time interval.

Several methods trying to combine events with intensities have been proposed to achieve more image details in the reconstructed images. The DAVIS Brandli et al. [2014a] uses a shared photo-sensor array to simultaneously output events (DVS) and intensity images (APS). Brandli et al. [2014b] combine images and event streams from the DAVIS camera to create inter-frame intensity estimates by dynamically estimating the contrast threshold (temporal contrast) of each event. Each new image frame resets the intensity estimate, preventing excessive growth of integration error. However, it also discards important accumulated event information. Scheerlinck et al. [2018] propose an asynchronous event-driven complementary filter to combine APS intensity images with events, and obtain continuous-time image intensities. Shedligeri and Mitra [2019] first exploit two intensity images to estimate depth. Second, they only use events to reconstruct a pseudo-intensity sequence (using method Reinbacher et al. [2016]) between the two intensity images. They, taking the pseudo-intensity sequence, they estimate the ego-motion using visual odometry. With the estimated 6-DOF pose and depth, they directly warp the intensity image to the intermediate location. Liu et al. [2017a] assume a scene should have a static background. Thus, their method needs an extra sharp static foreground image as input, and the event data are used to align the foreground with the background.

Image deblurring. Recently, significant progress has been made in blind image deblurring. Traditional deblurring methods usually make assumptions on the scenes (such as a static scene) or exploit multiple images (such as stereo, or video) to solve the deblurring problem. Significant progress has been made in the field of single image deblurring. Methods using gradient based regularizers, such as Gaussian scale mixture Fergus et al. [2006], $l_1 \backslash l_2$ norm Krishnan et al. [2011], edge-based patch priors Sun et al. [2013]; Yu et al. [2014] and l_0-norm regularizer Xu et al. [2013]; Pan et al. [2019b], have been proposed. Non-gradient-based priors such as the color line based prior Lai et al. [2015], and the extreme channel (dark/bright channel) prior Pan et al.

[2017a]; Yan et al. [2017a] have also been explored. Since blur parameters and the latent image are difficult to be estimated from a single image, the single-image-based approaches are extended to use multiple images Kim and Lee [2015]; Sellent et al. [2016]; Pan et al. [2017b, 2018]; Pan et al. [2020].

Driven by the success of deep neural networks, Sun et al. [2015] propose a convolutional neural network (CNN) to estimate locally linear blur kernels. Gong et al. [2017b] learn optical flow from a single blurred image through a fully-convolutional deep neural network. The blur kernel is then obtained from the estimated optical flow to restore the sharp image. Nah et al. [2017] propose a multi-scale CNN that restores latent images in an end-to-end learning manner without assuming any restricted blur kernel model. Tao et al. [2018] propose a light and compact network, SRN-DeblurNet, to deblur the image. However, deep deblurring methods generally need a large dataset to train the model and usually require sharp images provided as supervision. In practice, blurred images do not always have corresponding ground-truth sharp images.

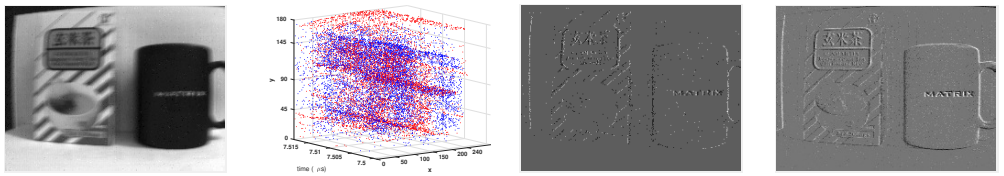
Blurred image to sharp video. Recently, two deep learning-based methods Jin et al. [2018]; Purohit et al. [2019] propose to restore a video from a single blurred image with a fixed sequence length. However, their reconstructed videos do not obey the 3D geometry of the scene and camera motion, limiting the further application of the reconstructed video, such as optical flow estimation. Although deep learning-based methods achieve impressive performance in various scenarios, their success heavily depends on the consistency between the training datasets and the testing datasets, thus hinder the generalisation ability for real-world applications.

6.4 Formulation

Our goal is to reconstruct a high frame rate, sharp video from a single or multiple (blurred) images and their corresponding events. In this section, we first introduce our **EDI** model. Then, we extend it to the **mEDI** model that includes multiple blurred images. Our models, both **EDI** and **mEDI**, can tackle various blur types and work stably in highly dynamic scenarios and low lighting conditions.

6.4.1 Event Camera Model

Event cameras are bio-inspired sensors that asynchronously report logarithmic intensity changes Brandli et al. [2014a]; Lichtsteiner et al. [2008]. Unlike conventional cameras that produce full images at a fixed frame rate, event cameras trigger events whenever the change in intensity at a given pixel exceeds a preset threshold. Event cameras do not suffer from limited dynamic



(a) The Blurred Image (b) The Events (c) $E(t) = \int e(t)\,dt$ (d) $\frac{1}{T}\int \exp(c\,\mathbf{E}(t))dt$

(e) c = 0.10 (f) c = 0.22 (g) c = 0.23 (h) c = 0.60

Figure 6.2: *The event data and our reconstructed result, where (a) and (b) are the input of our method. (a) The intensity image from the DAVIS. (b) Events from the event camera plotted in 3D space-time (x, y, t) (blue: positive event; red: negative event). (c) The first integral of several events during a small time interval. (d) The second integral of events during the exposure time. (e)-(h) Samples of reconstructed image with different c. The value is from low (0.10), to proper (around 0.23) and high (0.60). Note, c = 0.23 in (g) is the chosen automatically by our optimization process.*

ranges typical of sensors with the synchronous exposure time, and capture the high-speed motion with microsecond accuracy.

Inherent in the theory of event cameras is the concept of the latent image $\mathbf{L}_{xy}(t)$, denoting the instantaneous intensity at pixel (x, y) at time t, related to the rate of photon arrival at that pixel. The latent image $\mathbf{L}_{xy}(t)$ is not directly output by the camera. Instead, the camera outputs a sequence of *events*, denoted by (x, y, t, σ). Here, (x, y) denote image coordinates, t denotes the time the event takes place, and polarity $\sigma = \pm 1$ denotes the direction (increase or decrease) of the intensity change at that pixel and time. Polarity is given by,

$$\sigma = \mathcal{T}\left(\log\left(\frac{\mathbf{L}_{xy}(t)}{\mathbf{L}_{xy}(t_{\text{ref}})}\right), c\right), \tag{6.1}$$

where $\mathcal{T}(\cdot, \cdot)$ is a truncation function,

$$\mathcal{T}(d, c) = \begin{cases} +1, & d \geq c, \\ -1, & d \leq -c. \end{cases}$$

Here, c is a threshold parameter determining whether an event should be recorded or not, $\mathbf{L}_{xy}(t)$ and $\mathbf{L}_{xy}(t_{\text{ref}})$ denote the intensity of the pixel (x, y) at time instances t and t_{ref}, respectively. When an event is triggered, $\mathbf{L}_{xy}(t_{\text{ref}})$ at

that pixel is updated to a new intensity level. As described by Lichtsteiner et al. [2008], the DVS only uses a global threshold c. However, the contrast threshold of an event camera is not constant, but normally distributed. Several methods Delbruck et al. [2020]; Gallego et al. [2017] assume that the positive and negative contrast thresholds (i.e., , c_+ and c_-) exhibit different distribution noise. We observed using a global threshold c, (i.e., , $c_+ = c_-$) also yields satisfying video deblurring and high-frame rate reconstruction results while significantly simplifying the optimization procedure. Thus, we adopt a global c in the following section.

6.4.2 Intensity Image Formation

In addition to event streams, event cameras can provide full-frame grey-scale intensity images, at a much lower rate than the event sequence. Grey-scale images may suffer from motion blur due to their long exposure time. A general model of the blurred image formation is given by,

$$\mathbf{B} = \frac{1}{T}\int_{f-T/2}^{f+T/2} \mathbf{L}(t)\, dt, \tag{6.2}$$

where $\mathbf{B}$ is the blurred image, equal to the average of latent images during the exposure time $[f - T/2, f + T/2]$. Let $\mathbf{L}(f)$ be the snapshot of the image intensity at time f, the latent sharp image at the centre of the exposure period.

6.4.3 Event-based Double Integral Model

We aim to recover the latent sharp intensity video by exploiting both the blur model and the event model. We define $e_{xy}(t)$ as a function of continuous time t such that,

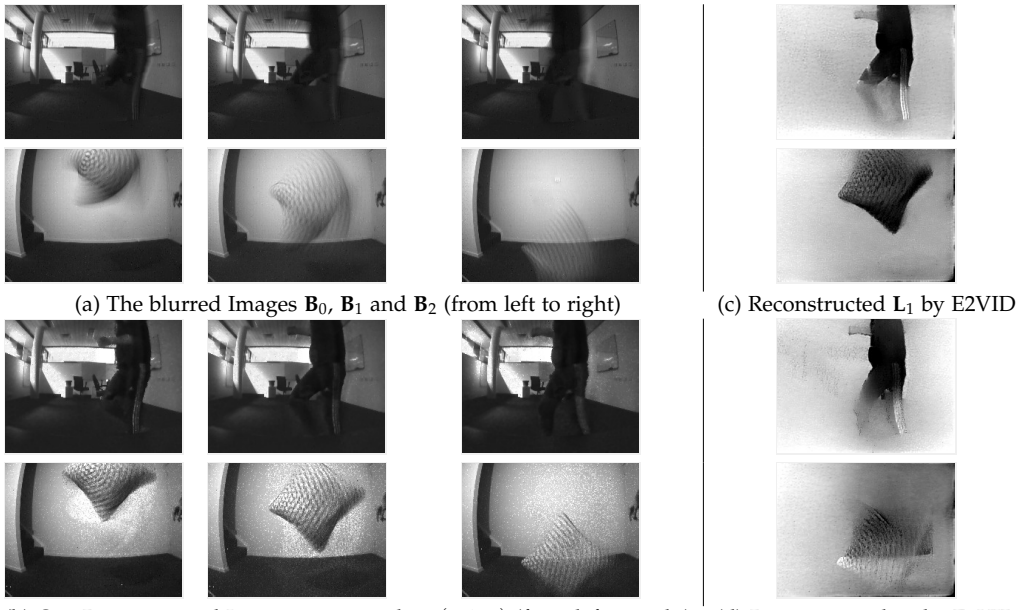
$$e_{xy}(t) = \sigma\ \delta_{t_0}(t),$$

whenever there is an event (x, y, t_0, σ). Here, $\delta_{t_0}(t)$ is an impulse function, with unit integral, at time t_0, and the sequence of events is turned into a continuous time signal, consisting of a sequence of impulses. There is such a function $e_{xy}(t)$ for every point (x, y) in the image. Since each pixel can be treated separately, we omit the subscripts x, y.

Given a reference timestamp f, we define $\mathbf{E}(t)$ as the sum of events between time f and t,

$$\mathbf{E}(t) = \int_f^t e(s)ds,$$

which represents the proportional change in intensity between time f and t. Except under extreme conditions, such as glare and no-light conditions, the



(a) The blurred Images $\mathbf{B}_0$, $\mathbf{B}_1$ and $\mathbf{B}_2$ (from left to right)

(c) Reconstructed $\mathbf{L}_1$ by E2VID

(b) Our Reconstructed Images $\mathbf{L}_0$, $\mathbf{L}_1$ and $\mathbf{L}_2$ ($\times 100$) (from left to right)

(d) Reconstructed $\mathbf{L}_2$ by E2VID

Figure 6.3: *The examples of our reconstructed results are based on our real event dataset. The threshold c is estimated automatically from three blurred images and their events based on our mEDI model. (a), (b) Blur image and our reconstructed Images* $\mathbf{L}_0$, $\mathbf{L}_1$ *and* $\mathbf{L}_2$ *(c), (d) Reconstruction results of* $\mathbf{L}_1$ *and* $\mathbf{L}_2$ *by Rebecq et al. [2019] from only events. The time resolution here is around* $\times 6$ *based on their default settings. The time resolution of the reconstructed video by E2VID Rebecq et al. [2019] is around* $\times 8$ *to* 15 *times higher than the time resolution of the original video. (Best viewed on screen).*

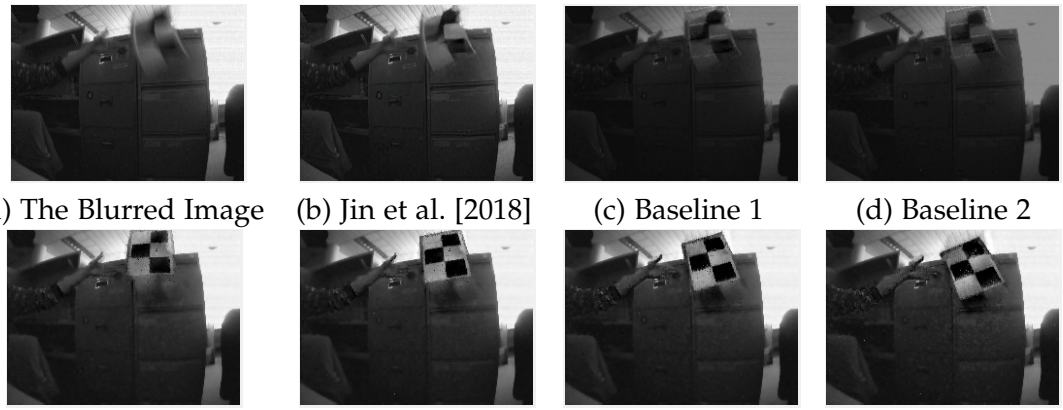
latent image sequence $\mathbf{L}(t)$ is expressed as,

$$\mathbf{L}(t) \ = \mathbf{L}(f) \, \exp(c \, \mathbf{E}(t)) \ .$$

In particular, an event (x, y, t, σ) is triggered when the intensity of a pixel (x, y) increases or decreases by an amount c at time t. With a high enough temporal resolution, the intensity changes of each pixel can be segmented to consecutive event streams with different amounts of events. We put a tilde on top of things to denote logarithm, e.g., $\widetilde{\mathbf{L}}(t) = \log(\mathbf{L}(t))$. Thus, we have,

$$\widetilde{\mathbf{L}}(t) \ = \widetilde{\mathbf{L}}(f) + c \, \mathbf{E}(t). \tag{6.3}$$

Given a sharp frame, we can reconstruct a high frame rate video from the sharp starting point $\mathbf{L}(f)$ by using Eq. (6.3). When an input image is blurred, a trivial solution would be to first deblur the image with an existing deblurring method and then to reconstruct a video using Eq. (6.3) (see Fig. 6.4 for details). However, in this way, the event data between intensity images are



(a) The Blurred Image (b) Jin et al. [2018] (c) Baseline 1 (d) Baseline 2

(f) Samples of Our Reconstructed Video

Figure 6.4: *Deblurring and reconstruction results on our real blur event dataset. (a) Input blurred images. (b) Deblurring result of Jin et al. [2018]. (c) Baseline 1 for our method. We first use the state-of-the-art video-based deblurring method Jin et al. [2018] to recover a sharp image. Then use the sharp image as input to a state-of-the-art reconstruction method Scheerlinck et al. [2018] to get the intensity image. (d) Baseline 2 for our method. We first use method Scheerlinck et al. [2018] to reconstruct an intensity image. Then use a deblurring method Jin et al. [2018] to recover a sharp image. (e) Samples from our reconstructed video from* $\mathbf{L}(0)$ *to* $\mathbf{L}(150)$.

not fully exploited, thus resulting in inferior performance. Moreover, none of existing deblurring methods can be guaranteed to work stably in a complex dynamic scenery. Instead, we propose to reconstruct the video by exploiting the inherent connection between events and blur, and present the following model.

As for the blurred image,

$$\begin{aligned}\mathbf{B} &= \frac{1}{T}\int_{f-T/2}^{f+T/2} \mathbf{L}(f)\, \exp\Big(c\,\mathbf{E}(t)\Big)\, dt \\ &= \frac{\mathbf{L}(f)}{T}\int_{f-T/2}^{f+T/2} \exp\Big(c\int_{f}^{t} e(s)ds\Big)\, dt\ .\end{aligned} \tag{6.4}$$

In this manner, we build the relation between the captured blurred image $\mathbf{B}$ and the latent image $\mathbf{L}(f)$ through the double integral of the event. We name Eq. (6.4) the **Event-based Double Integral (EDI)** model.

We denote

$$\mathbf{J}(c) = \frac{1}{T}\int_{f-T/2}^{f+T/2} \exp(c\,\mathbf{E}(t))dt.$$

(a) The Blurred Image (b) E2VID ($\times 8$) (c) Our EDI ($\times 100$) (d) Our mEDI ($\times 100$)

Figure 6.5: *Examples of reconstruction results on real event dataset. (a) The intensity image from the event camera. (b) Reconstruction result of our E2VID* et al.*Rebecq et al. [2019] from only events. The temporal resolution is around* $\times 8$ *based on their default settings, while ours are* $\times 100$ *times higher than the original videos'. (c) Reconstruction result of our EDI model* et al.*Pan et al. [2019c] from combining events and a single blurred frame. (d) Reconstruction result of our mEDI model from combining events and multiple blurred frames. Our method based on multiple images gets better results than our previous one based only on one single image, especially on large motion scenery and extreme light conditions. (Best viewed on screen).*

Taking the logarithm on both sides of Eq. (6.4) and rearranging it yields

$$\widetilde{\mathbf{L}}(f) = \widetilde{\mathbf{B}} - \widetilde{\mathbf{J}}(c), \tag{6.5}$$

which shows a linear relationship between the blurred image, the latent image and integrated events in the log space.

6.4.4 High Frame Rate Video Generation

The right-hand side of Eq. (6.5) is known, apart from perhaps the value of the contrast threshold c, the first term from the grey-scale image, the second term from the event sequence, so it is possible to compute $\widetilde{\mathbf{L}}$, and hence $\mathbf{L}$ by exponentiation. Subsequently, from Eq. (6.3) the latent image $\mathbf{L}(t)$ at any time may be computed.

To avoid accumulated errors of constructing a video from many frames of a blurred video, it is more suitable to construct each frame $\mathbf{L}(t)$ using the closest blurred frame.

Theoretically, we could generate a video with a frame rate as high as the DVS's event rate. However, since each event carries little information and is subject to noise, several events must be processed together to yield a reasonable image. We generate a reconstructed frame every $50 - 100$ events, so

for our experiment, the frame rate of the reconstructed video is usually 200 times greater than the input low frame rate video. Furthermore, as indicated by Eq. (6.5), the challenging blind motion deblurring problem has been reduced to a single variable optimization problem of finding the best value of the contrast threshold c.

6.4.5 Finding c with Regularization Terms

As indicated by Eq. (6.5), the blind motion deblurring problem has been reduced to a single variable optimization problem of how to find the best value of the threshold c. To this end, we need to build an evaluation metric (energy function) that can evaluate the quality of the deblurred image $\mathbf{L}(t)$. Specifically, we propose to exploit different prior knowledge for sharp images and the event data.

Edge constraint for event data. As mentioned before, when a proper c is given, our reconstructed image $\mathbf{L}(c,t)$ will contain much sharper edges compared with the original input intensity image. Furthermore, event cameras inherently yield responses at moving intensity boundaries, so edges in the latent image may be located where (and when) events occur. We convolve the event sequence with an exponentially decaying window, to obtain a denoised yet wide edge boundary,

$$\mathbf{M}(t) = \int_{-T/2}^{T/2} \exp(-(|t-s|))\, e(s)\, ds,$$

Then, we use the Sobel filter $\mathcal{S}$ to get a sharper binary edge map, which is also applied to $\mathbf{L}(c,t)$. Here, we use $\mathbf{L}(c,t)$ to present the latent sharp image $\mathbf{L}(t)$ with different c.

Here, we use cross-correlation between $\mathcal{S}(\mathbf{L}(c,t))$ and $\mathcal{S}(\mathbf{M}(t))$ to evaluate the sharpness of $\mathbf{L}(c,t)$.

$$\phi_{\text{edge}}(c) = \sum_{x,y} \mathcal{S}(\mathbf{L}(c,t))(x,y) \cdot \mathcal{S}(\mathbf{M}(t))(x,y)\ . \tag{6.6}$$

Intensity Image Constraint. Total variation is used to suppress noise in the latent image while preserving edges, and to penalize spatial fluctuations Rudin et al. [1992].

$$\phi_{\text{TV}}(c) = |\nabla \mathbf{L}(c,t)|_1, \tag{6.7}$$

where ∇ represents the gradient operators.

Energy Minimization. The optimal c can be estimate by solving Eq. (6.8),

$$\min_c \phi_{\text{TV}}(c) + \lambda \phi_{\text{edge}}(c), \tag{6.8}$$

where λ is a trade-off parameter. The response of cross-correlation reflects the matching rate of $\mathbf{L}(c,t)$ and $\mathbf{M}(t)$ which makes $\lambda < 0$. This single-variable minimization problem can be solved by Golden Section Search.

6.5 Using More Than One Frame

Though our EDI can reconstruct high frame rate videos efficiently, noise from events can easily degrade the quality of reconstructed videos with low temporal consistency. In addition, regularization terms in the energy function introduce unexpected weight parameters. Therefore, we propose a multiple images based approach to tackle the above problems with a simple yet effective optimization solution.

6.5.1 Multiple Event-based Double Integral Model

Suppose an event camera captures a continuing sequence of events, and also blurred images, $\mathbf{B}_i$ for $i = 0, \cdots, n$. Assume that the exposure time is T and the reference frame $\mathbf{B}_i$ is at time f_i. Each $\mathbf{B}_i$ is associated with a latent image $\mathbf{L}_i(f_i)$ and is generated as an integral of $\mathbf{L}_i(t)$ over the exposure interval $[f_i - T/2, f_i + T/2]$. In addition, we rewrite $\mathbf{E}(t)$, $\mathbf{L}(t)$ and $\mathbf{J}(c)$ for the i^{th} frame as

$$\begin{aligned}
\mathbf{E}_i(t) &= \int_{f_i}^{t} e(s)ds \\
\mathbf{L}_i(t) &= \mathbf{L}_i(f_i)\ \exp(c\,\mathbf{E}_i(t)) \\
\mathbf{J}_i(c) &= \frac{1}{T}\int_{f_i-T/2}^{f_i+T/2} \exp(c\,\mathbf{E}_i(t))dt.
\end{aligned}$$

The **EDI** model in Eq. (6.5) in section 6.4 gives

$$\widetilde{\mathbf{B}}_i = \widetilde{\mathbf{L}}_i(f_i) + \widetilde{\mathbf{J}}_i(c), \tag{6.9}$$

for each blurred image in the sequence. We use $\mathbf{L}_i$ to represent $\mathbf{L}_i(f_i)$ in the following section. Then, Eq. (6.9) is written as

$$\widetilde{\mathbf{B}}_i = \widetilde{\mathbf{L}}_i + \widetilde{\mathbf{J}}_i(c) = \widetilde{\mathbf{L}}_i + a_i\ . \tag{6.10}$$

The latent image $\mathbf{L}_{i+1}$ is formed from latent image $\mathbf{L}_i$ by integrating events over the period $[f_i, f_{i+1}]$, which gives

$$\widetilde{\mathbf{L}}_{i+1} = \widetilde{\mathbf{L}}_i + c\int_{f_i}^{f_{i+1}} e(s)ds = \widetilde{\mathbf{L}}_i + b_i. \tag{6.11}$$

This describes the **mEDI** model based on multiple images and their events.

$$\begin{aligned} \widetilde{\mathbf{L}}_i &= \widetilde{\mathbf{B}}_i - a_i \\ \widetilde{\mathbf{L}}_{i+1} - \widetilde{\mathbf{L}}_i &= b_i. \end{aligned} \tag{6.12}$$

The known values are $\widetilde{\mathbf{B}}_i$, whereas the unknowns are $\widetilde{\mathbf{L}}_i$, a_i and b_i. These quantities are associated with a single pixel and we solve for each pixel independently. We therefore obtain a set of linear equations based on Eq. (6.12) as

$$\left[\begin{array}{ccccccc} 1 & -1 & & & & & \\ & 1 & -1 & & & & \\ & & \ddots & \ddots & & & \\ & & & & 1 & -1 & \\ \hline 1 & & & & & & \\ & 1 & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & & & & & 1 \end{array}\right] \begin{bmatrix} \widetilde{\mathbf{L}}_1 \\ \widetilde{\mathbf{L}}_2 \\ \vdots \\ \widetilde{\mathbf{L}}_n \end{bmatrix} = \begin{bmatrix} -b_1 \\ \vdots \\ -b_{n-1} \\ \widetilde{\mathbf{B}}_1 - a_1 \\ \vdots \\ \widetilde{\mathbf{B}}_n - a_n \end{bmatrix}, \tag{6.13}$$

where a_i and b_i depend on c, but particularly a_i depends on c in a non-linear way. Writing Eq. (6.13) as $\mathbf{Ax} = \mathbf{w}$, the least-squares solution is given by solving $\mathbf{A}^T\mathbf{Ax} = \mathbf{A}^T\mathbf{w}$.

6.5.2 LU Decomposition

Because of their particular form, these equations can be solved very efficiently as will now be shown. Expanding the equations

$$\mathbf{A}^T\mathbf{Ax} = \mathbf{A}^T\mathbf{w}$$

gives

$$\begin{bmatrix} 2 & -1 & & & & & \\ -1 & 3 & -1 & & & & \\ & -1 & 3 & -1 & & & \\ & & & \ddots & & & \\ & & & & -1 & 3 & -1 \\ & & & & & -1 & 2 \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{L}}_1 \\ \widetilde{\mathbf{L}}_2 \\ \vdots \\ \widetilde{\mathbf{L}}_n \end{bmatrix} = \begin{bmatrix} \widetilde{\mathbf{B}}_1 - a_1 - b_1 \\ \widetilde{\mathbf{B}}_2 - a_2 - b_2 + b_1 \\ \vdots \\ \widetilde{\mathbf{B}}_{n-1} - a_{n-1} - b_{n-1} + b_{n-2} \\ \widetilde{\mathbf{B}}_n - a_n + b_{n-1} \end{bmatrix}. \tag{6.14}$$

This is a particularly easy set of equations to solve. Since it has to be

solved for each pixel, it is important to do it efficiently. The best way to solve Eq. (6.14) is to take the LU decomposition of the left-hand-side matrix, which has a particularly simple form.

Let $\mathbf{A}^T\mathbf{w} = \mathbf{r}$, we writing Eq. (6.14) as $\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{r}$. The LU decomposition of $\mathbf{A}^T\mathbf{A}$ (with appropriate reordering of rows) is given by

$$\mathtt{LU} = \begin{bmatrix} -2 & -5 & -13 & \cdots & 1 \\ 1 & & & & 0 \\ & 1 & & & 0 \\ & & \ddots & & \vdots \\ & & & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 & 3 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 3 & -1 \\ & & & -1 & 2 \\ & & & & \phi_{2n-1} \end{bmatrix}.$$

More precisely, if the Fibonacci sequence is $1, 2, 3, 5, 8, \cdots$ and ϕ_k denotes the $k-$th entry of this sequence (thus $\phi(0) = 1$, $\phi(2) = 2$), then the top line of the left-hand matrix is

$$\begin{bmatrix} \phi_2 & \phi_4 & \cdots & \phi_{2(n-1)} & 1 \end{bmatrix},$$

consisting of the even numbered entries of the Fibonacci sequence. The entry at the bottom right of the right-hand matrix is ϕ_{2n-1}, the next odd-numbered Fibonacci number, which is also the determinant of the original matrix. Solving equations by LU decomposition and back-substitution is particularly simple in this case. The procedure in solving equations $\mathtt{LU}\mathbf{x} = \mathbf{r}$ is done by solving

$$\begin{aligned} \mathtt{L}\mathbf{y} &= \mathbf{r} \\ \mathtt{U}\mathbf{x} &= \mathbf{y}. \end{aligned}$$

The solution of $\mathtt{L}\mathbf{y} = \mathbf{r} = (\mathtt{r}_1, \mathtt{r}_2, \cdots, \mathtt{r}_\mathtt{n})^\mathtt{T}$ is simply

$$\mathbf{y} = (r_2, r_3, \cdots, r_n, \sum_{i=1}^{n-1} r_i \phi_{2i})^T.$$

The solution of $\mathtt{U}\mathbf{x} = \mathbf{y}$ is given by back-substitution from the bottom:

$$\begin{aligned} x_n &= y_n / \phi_{2n-1} = \sum_{i=1}^{n-1} r_i \phi_{2i} / \phi_{2n-1} \\ x_{n-1} &= 2x_n - r_n \\ x_{n-2} &= 3x_{n-1} - x_n - r_{n-1} \\ x_{n-3} &= 3x_{n-2} - x_{n-1} - r_{n-2} \\ &\cdots \\ x_1 &= 3x_2 - x_3 - r_2 \end{aligned} \tag{6.15}$$

The values x_i is the pixel value for latent image $\mathbf{L}_i$. If c is known, then the values on the right of are dependent on c, and the sequence of $\mathbf{L}_n$ can be computed.

$$\begin{aligned}
\mathbf{L}_n &= \sum_{i=1}^{n-1} r_i \phi_{2i} / \phi_{2n-1} \\
\mathbf{L}_{n-1} &= 2\mathbf{L}_n - \widetilde{\mathbf{B}}_n - a_n + b_{n-1} \\
\mathbf{L}_{n-2} &= 3\mathbf{L}_{n-1} - \mathbf{L}_n - \widetilde{\mathbf{B}}_{n-1} - a_{n-1} - b_{n-1} + b_{n-2} \\
\mathbf{L}_{n-3} &= 3\mathbf{L}_{n-2} - \mathbf{L}_{n-1} - \widetilde{\mathbf{B}}_{n-2} - a_{n-2} - b_{n-2} + b_{n-3} \\
&\cdots \\
\mathbf{L}_1 &= 3\mathbf{L}_2 - \mathbf{L}_3 - \widetilde{\mathbf{B}}_2 - a_2 - b_2 + b_1
\end{aligned} \tag{6.16}$$

Furthermore, the problem has been reduced to a single variable optimization problem of how to find the best value of the contrast threshold c.

6.6 Optimization

The unknown contrast threshold c represents the minimum change in log intensity required to trigger an event. With an appropriate c in Eq. (6.12), we can generate a sequence of sharper images. Here, we propose two different methods to estimate the unknown variable c, which are manually chosen and automatically optimized by our approach.

6.6.1 Manually Chosen c

According to our **mEDI** model in Eq. (6.12), given a value for c, we obtain sharp images. Therefore, we develop a method for deblurring by manually inspecting the visual effect of the deblurred image. In this way, we incorporate human perception into the reconstruction loop and the deblurred images should satisfy human observation. In Fig. 6.2 and 6.6, we give examples for manually chosen results on our dataset, and the *Event-Camera Dataset* Mueggler et al. [2017].

6.6.2 Automatically Chosen c

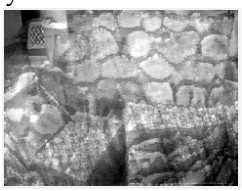
To automatically find the best c, we need to build an evaluation metric (energy function) that can evaluate the quality of the deblurred image $\mathbf{L}_i(t)$. Different from our EDI that including regularization terms (with extra weight parameters) in the energy function, we develop a simple yet effective optimization

(a) The Blurred image

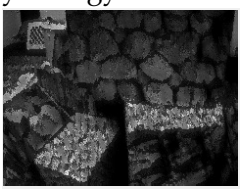
(b) Tao et al. [2018]

(c) By Human Observation

(d) By Energy Minimization



(e) E2VID



(f) mEDI

Figure 6.6: *An example of our reconstruction result using different methods to estimate c, on a real sequence from the* Event-Camera Dataset *Mueggler et al. [2017]. (a) The blurred image. (b) Deblurring result of Tao et al. [2018]. (c) Our result where c is chosen by manual inspection. (d) Our result where c is computed automatically by our proposed energy minimization Eq.* (6.19)*. (e) Reconstruction results of Rebecq et al. [2019] from only events. The temporal resolution of the reconstructed video is around* $\times 8$ *times higher than the original videos' based on their default settings. (f) Our mEDI result where the temporal resolution is the same as (e).*

solution. More specifically, we adopt the Fibonacci sequence search to solve the optimization which significantly reduces the computational complexity.

6.6.2.1 Energy function

The values on the right-hand side of Eq. (6.12) depend on an unknown parameter c. In particular, we write

$$\begin{aligned} b_i &= c \int_{f_i}^{t} e(s)ds \\ a_i &= \log\left(\frac{1}{T}\int_{f_i-T/2}^{f_i+T/2} \exp(c\,\mathbf{E}(t))dt\right). \end{aligned} \tag{6.17}$$

Given c, x_i can be solved by `LU` decomposition in Sec. 6.5.2. Subsequently,

Table 6.1: *Quantitative comparisons on the Synthetic dataset Nah et al. [2017]. The provided videos are able to generate not only blurred images but also event data. All methods Pan et al. [2017b]; Sun et al. [2015]; Gong et al. [2017b]; Scheerlinck et al. [2018] are tested under the same blur condition, where methods Nah et al. [2017]; Jin et al. [2018]; Tao et al. [2018]; Zhang et al. [2018] use GoPro dataset Nah et al. [2017] to train their models. Note, Baseline 1 is based on Tao et al. [2018] + Scheerlinck et al. [2018], and Baseline 2 is based on Scheerlinck et al. [2018] + Tao et al. [2018]. Jin Jin et al. [2018] achieves their best performance when the image is down-sampled to 45% mentioned in their paper. In this dataset, blurry images are generated by averaging every 11 frames, and they treat the clean middle one (the 6^{th} frame) as the ground truth. The top part in this figure aims to compare with deblurring methods, and only the blurry image (the 6^{th} frame) is evaluated. The bottom part shows the measures of whole reconstructed videos.*

Average result of the deblurred **images** on dataset Nah et al. [2017]									
	Pan	Sun	Gong	Jin	Tao	Zhang	Nah	EDI	mEDI
PSNR(dB)	23.50	25.30	26.05	26.98	30.26	29.18	29.08	29.06	**30.29**
SSIM	0.8336	0.8511	0.8632	0.8922	0.9342	0.9306	0.9135	**0.9430**	0.9194
Average result of the reconstructed **videos** on dataset Nah et al. [2017]									
	Baseline 1		Baseline 2		Scheerlinck		Jin	EDI	mEDI
PSNR(dB)	25.52		26.34		25.84		25.62	28.49	**28.83**
SSIM	0.7685		0.8090		0.7904		0.8556	**0.9199**	0.9098

from Eq. (6.12) the blur image $\mathbf{B}_i$ can be computed.

$$\widetilde{\mathbf{B}}_i(c) = x_i + a_i \tag{6.18}$$

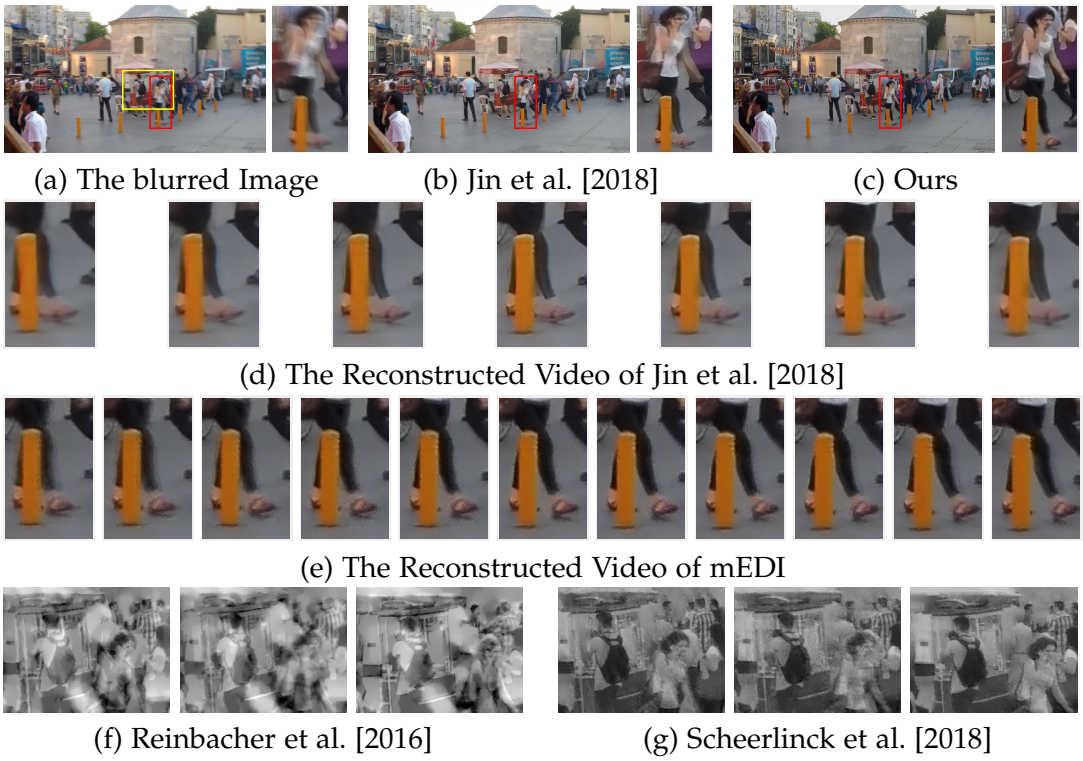
Here, we use $\mathbf{B}_i(c)$ to present the blurred image $\mathbf{B}_i$ with different c. In this case, the optimal c can be estimated by solving Eq. (6.19),

$$\min_c ||\mathbf{B}_i(c) - \mathbf{B}||_2^2. \tag{6.19}$$

Examples show that as a function of c, the residual error in solving the equations is not convex. However, in most cases (empirically) it seems to be convex, or at least it has a single minimum (See Fig. 6.8 for an example).

6.6.2.2 Fibonacci search

Finding the minimum of a function along a single line is easy if that function has a single minimum. In the case of least-squares minimisation problems, various strategies for determining the line-search direction are currently used, such as conjugate gradient methods, gradient descent, and the Levenberg-Marquardt method. When the function has only one stationary point, the maximum/minimum, and when it depends on a single variable in a finite in-



(a) The blurred Image (b) Jin et al. [2018] (c) Ours

(d) The Reconstructed Video of Jin et al. [2018]

(e) The Reconstructed Video of mEDI

(f) Reinbacher et al. [2016] (g) Scheerlinck et al. [2018]

Figure 6.7: *An example of the reconstructed result on our synthetic event dataset based on the GoPro dataset Nah et al. [2017]. Nah et al. [2017] provides videos to generate blurred images and event data. (a) The blurred image. The red close-up frame is for (b)-(e), the yellow close-up frame is for (f)-(g). (b) The deblurring result of Jin et al. [2018]. (c) Our deblurring result. (d) The crop of their reconstructed images and the frame number is fixed at 7. Jin et al. [2018] uses the GoPro dataset added with 20 scenes as training data and their model is supervised by 7 consecutive sharp frames. (e) The crop of our reconstructed images. (f) The crop of Reinbacher et al. [2016] reconstructed images from only events. (g) The crop of Scheerlinck et al. [2018] reconstructed image, they use both events and the intensity image. For (e)-(g), the shown frames are the chosen examples, where the length of the reconstructed video is based on the number of events. (Best viewed on screen).*

terval, the most efficient way to find the maximum is based on the Fibonacci numbers. The procedure, now known widely as 'Fibonacci search', was discovered and shown optimal in a minimax sense by Kiefer Kiefer [1953]; Press et al. [1988].

In this work, we use Fibonacci search for the value of c that gives the least error. In Fig. 6.8, we illustrate the clearness of the reconstructed image (in PSNR value) as a function of the value of c. As demonstrated in the figure, our proposed reconstruction metric could properly locate/identify the best-deblurred image with peak PSNR.

In our proposed method, we assume that $c_+ = c_-$ and use a global c based

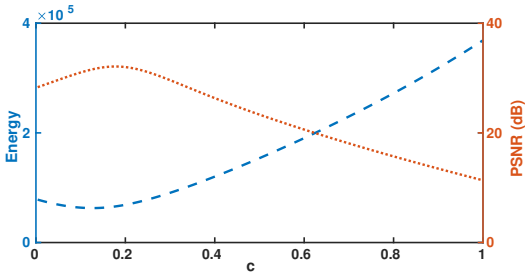


Figure 6.8: *Deblurring performance plotted against the value of c. The image is clearer with higher PSNR value.*

on the following reasons:

1) As illustrated in Fig. 6.8 our deblurring performance has a relatively flat crest against different values of c. Experimental results demonstrated that the quality of our reconstructed videos is robust to the estimation of c within a certain range.

2) We have conducted the experiments with $c_+ \neq c_-$, namely, optimising two variables in our formulation. We observed that the improvement on PSNR is less than 0.1dB in comparison to the results of optimising a global c. However, the computational complexity increases from $\mathcal{O}(n)$ to $\mathcal{O}(n^2)$.

Therefore, we believe it is worthy of trading off between computational simplicity and performance accuracy.

6.7 Experiment

In our experiments, unless otherwise specified, the parameter c for reconstructing images is chosen automatically by our optimization process.

6.7.1 Experimental Setup

Synthetic dataset. To provide a quantitative comparison, we build a synthetic dataset based on the GoPro blur dataset Nah et al. [2017]. It supplies ground truth videos which are used to generate the blurred images. Similarly, we employ the ground-truth images to generate event data based on the methodology of *event camera model*. In this GoPro dataset, we did not notice obvious rolling shutter artefacts because images in this dataset were requested to be captured with low speed of camera motions for providing ground-truth latent sharp images.

Real dataset. We evaluate our method on a public Event-Camera dataset Mueggler et al. [2017], which provides a collection of sequences captured by

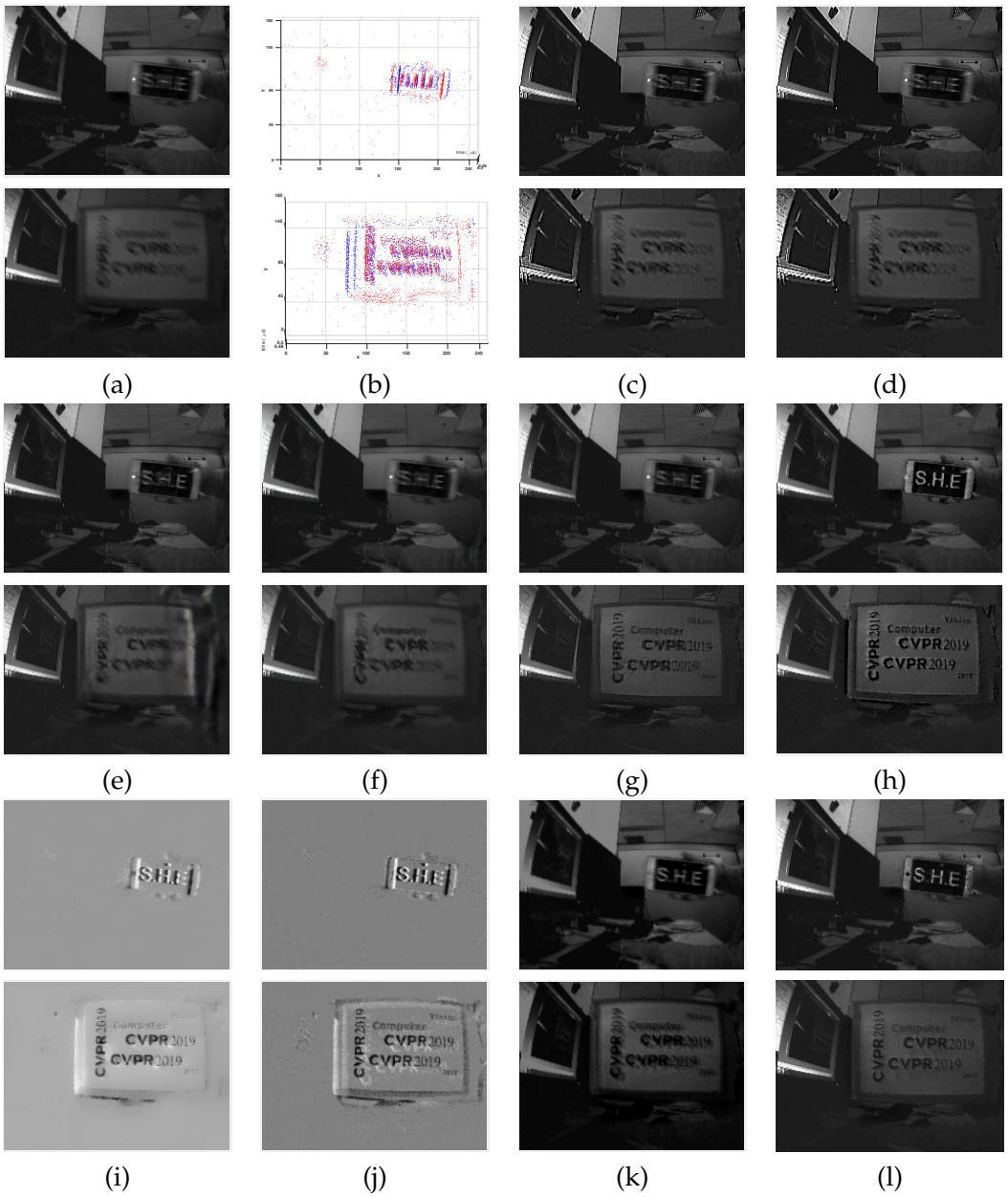


Figure 6.9: *Examples of reconstruction result on our real blur event dataset in low lighting and complex dynamic conditions (a) Input blurred images. (b) The event information. (c) Deblurring results of Pan et al. [2017a]. (d) Deblurring results of Yan et al. [2017a]. (e) Deblurring results of Tao et al. [2018]. (f) Deblurring results of Nah et al. [2017]. (g) Deblurring results of Jin et al. [2018] and they use video as training data. (h) Reconstruction result of Pan et al. [2019c] from combining events and frames. (i) Reconstruction result of Reinbacher et al. [2016] from only events. (j)-(k) Reconstruction results of Scheerlinck et al. [2018], (j) from only events, (k) from combining events and frames. (l) Our reconstruction result. Results in (c)-(g) show that real high dynamic settings and low light conditions are still challenging in the deblurring area. Results in (i)-(j) show that while intensity information of a scene is still retained with an event camera recording, color, and delicate texture information cannot be recovered. (Best viewed on screen).*

the event camera for high-speed robotics. Furthermore, we present our real *blur event dataset*, where each real sequence is captured with the DAVIS240 Brandli et al. [2014a] under different conditions, such as indoor, outdoor scenery, low lighting conditions, and different motion patterns (e.g., camera shake, objects motion) that naturally introduce motion blur into the APS intensity images. We also evaluate our method on a newly published Color Event Camera Dataset (CED) Scheerlinck et al. [2019b] built with DAVIS346 Red Color sensor. They present an extension of the event simulator ESIM Rebecq et al. [2018] that enables simulation of colour events. In contrast to GoPro cameras, event cameras, such as DAVIS, employ global shutters, where an entire scene is captured at the same instant. Therefore, global shutter cameras, e.g., our event camera, do not have rolling shutter effects.

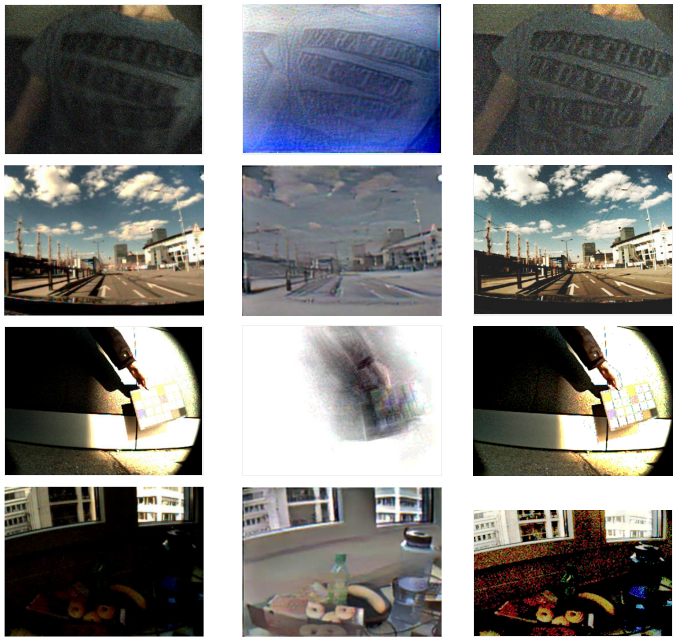
Implementation details. For all our real experiments, we use the DAVIS Brandli et al. [2014a] that shares photosensor array to simultaneously output events (DVS) and intensity images (APS). The framework is implemented using MATLAB®. It takes around 1.5 seconds to process one image on a single i7 core running at 3.6 GHz.

6.7.2 Experimental Results

We compare our proposed approach with state-of-the-art blind deblurring methods, including conventional deblurring methods Pan et al. [2017a]; Yan et al. [2017a], deep based dynamic scene deblurring methods Nah et al. [2017]; Jin et al. [2018]; Tao et al. [2018]; Zhang et al. [2018]; Sun et al. [2015], and event-based image reconstruction methods Rebecq et al. [2019]; Reinbacher et al. [2016]; Scheerlinck et al. [2018]. Moreover, Jin et al. [2018] can restore a video from a single blurred image based on a deep network, where the middle frame in the restored odd-numbered sequence is the best.

To prove the effectiveness of our model, we show some baseline comparisons in Fig. 6.4 and Table 6.1. For baseline 1, we first apply a state-of-the-art deblurring method Tao et al. [2018] to recover a sharp image, and then feed the recovered image as input to a reconstruction method Scheerlinck et al. [2018]. For baseline 2, we first use the video reconstruction method Scheerlinck et al. [2018] to reconstruct a sequence of intensity images, then apply the deblurring method Tao et al. [2018] to each frame. As seen in Table 6.1, our approach obtains higher PSNR and SSIM compared to baseline 1 and baseline 2. This also implies that our approach better exploits the event data to recover sharp images and reconstruct high frame rate videos.

In Table 6.1 and Fig. 6.7, we show quantitative and qualitative comparison on our synthetic dataset, respectively. As indicated in Table 6.1, our approach achieves the best performance on PSNR, and competitive results on SSIM compared to state-of-the-art methods and attains significant performance im-



(a) DAVIS frame (b) E2VID (c) Ours

Figure 6.10: *An example of our reconstruction result on the color event camera dataset CED Scheerlinck et al. [2019b]. (a) The input image. (b) Reconstruction results of Rebecq et al. [2019] from only events. The temporal resolution of the reconstructed video is around* $\times 12$ *times higher than the original videos' based on their default settings. (c) Our mEDI result where the temporal resolution is the same as (b). From top to bottom, a scene with a low lighting condition, an outdoor scene, a scene with slow-moving objects (static background), and an HDR scene. Our mEDI model performs well in the top two rows, while E2VID is able to provide vivid colour textures in the HDR scene. Note that our method focuses on reconstructing high-frame rate videos rather than changing the dynamic range of input videos. In order to illustrate our detailed textures in the HDR scene, we employ an HDR enhancement method Eilertsen et al. [2017].*

provements on high-frame video reconstruction.

In Fig. 6.3, Fig. 6.5 and Fig. 6.10, we qualitatively compare our generated videos with state-of-the-art event-based image reconstruction methods Rebecq et al. [2019]; Scheerlinck et al. [2018]; Pan et al. [2019c]. Experimental results indicate that event-only methods work well on scenes of fast camera motions since the distribution of events has a wide coverage of scene content. Also, E2VID Rebecq et al. [2019] is enabled to provide more vivid colour textures in the HDR scene. However, for scenes with a static background or a slowly moving background/foreground, the reconstructed images by event-only methods will lose texture details in the areas without events. On the contrary, our 'image and event' combined method achieves superior performance on scenes with high dynamic motions and works robustly even with

static backgrounds and sparse events.

We also report our reconstruction (and deblurring) results on real datasets, including text images and low-lighting images, in Fig. 6.1, Fig. 6.6, and Fig. 6.9.

Compared with state-of-the-art deblurring methods, our method achieves superior results. In comparison to existing event-based image reconstruction methods Reinbacher et al. [2016]; Scheerlinck et al. [2018]; Pan et al. [2019c]; Rebecq et al. [2019], our reconstructed images are not only more realistic but also contain richer details. For more deblurring results and **high-temporal resolution videos**, please visit our home page.

6.8 Limitation

Though event cameras record continuous, asynchronous streams of events that encode non-redundant information for our **mEDI** model, there are still some limitations when doing reconstruction.

1) Extreme lighting changes, such as suddenly turning on/off the light, moving from dark indoor scenes to outdoor scenes. The relatively low dynamic range of the intensity image might degrade the performance of our method in high dynamic scenes;

2) Event error accumulation, such as noisy event data, small object motions with fewer events. Though we integrate over short time intervals from the centre of the exposure time to mitigate this error, accumulated noise can reduce the quality of reconstructed images.

6.9 Conclusion

In this chapter, we have proposed a **multiple Event-based Double Integral (mEDI)** model to naturally connect intensity images and events recorded by an event camera (DAVIS), which also takes the blur generation process into account. In this way, our model can be used to recover the latent sharp images and reconstruct intermediate frames at a high frame rate. We also propose a simple yet effective method to solve our **mEDI** model. Due to the simplicity of our optimization process, our method is efficient as well. Extensive experiments show that our method can generate high-quality, high frame-rate videos efficiently under different conditions, such as low lighting and complex dynamic scenes.

Chapter 7

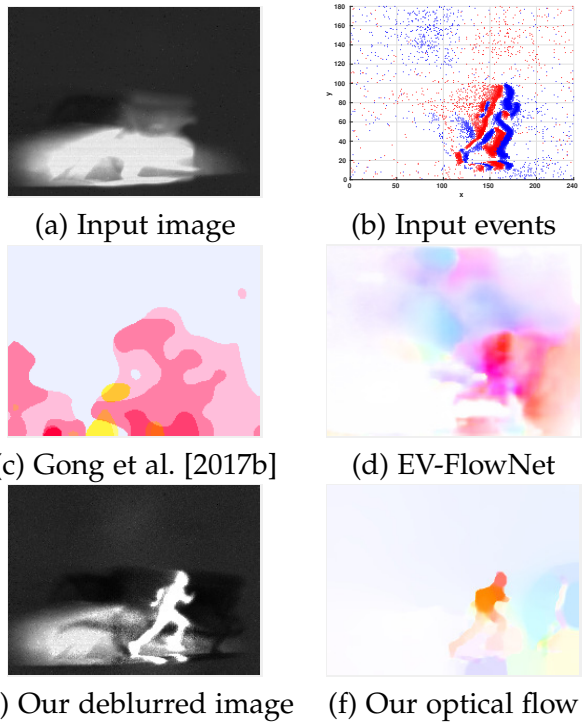
Single Image Optical Flow Estimation with an Event Camera

This chapter proposes a single image (potentially blurred) and event-based optical flow estimation approach to unlock their potential applications. In doing so, we introduce an *event-based brightness constancy* constraint on absolute intensity to encode the relation between optical flow and the event data. Also, we use the blur formation model in our objective function to handle optical flow estimation on the blurred image.

Liyuan Pan, Miaomiao Liu, and Richard Hartley. Single Image Optical Flow Estimation with an Event Camera. Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

7.1 Abstract

Event cameras are bio-inspired sensors that asynchronously report intensity changes in microsecond resolution. DAVIS can capture high dynamics of a scene and simultaneously output high temporal resolution events and low frame-rate intensity images. In this chapter, we propose a single image (potentially blurred) and event-based optical flow estimation approach. First, we demonstrate how events can be used to improve flow estimates. To this end, we encode the relation between flow and events effectively by presenting an event-based photometric consistency formulation. Then, we consider the special case of image blur caused by high dynamics in the visual environments and show that including the blur formation in our model further constrains flow estimation. This is in sharp contrast to existing works that ignore the blurred images while our formulation can naturally handle either blurred or sharp images to achieve accurate flow estimation. Finally, we reduce flow estimation, as well as image deblurring, to an alternative optimization problem of an objective function using the primal-dual algorithm. Experimental results on both synthetic and real data (with blurred and non-blurred images) show the superiority of our model in comparison to state-of-the-art approaches.



(a) Input image (b) Input events

(c) Gong et al. [2017b] (d) EV-FlowNet

(e) Our deblurred image (f) Our optical flow

Figure 7.1: **Optical flow estimation.** *(a) and (b) are the input to our method, where (a) shows the intensity image from DAVIS, and (b) visualises the integrated events over a temporal window (blue: positive event; red: negative event). (c) Flow result of Gong et al. [2017b] by using a single blurred image. (d) Flow result of Zhu et al. [2018a], by using events. (e) and (f) are our results. Our methods is able to handle large motion scenery. (Best viewed on screen).*

7.2 Introduction

Event cameras (such as DVS Lichtsteiner et al. [2008] and DAVIS Brandli et al. [2014a]) measure intensity changes at each pixel independently with microsecond accuracy. Unlike conventional cameras recording images at a fixed frame rate, event cameras trigger the event whenever the change in intensity at a given pixel exceeds a preset threshold. Event cameras are gaining attention for their high temporal resolution, robustness to low lighting and highly dynamic scenes which can be used for tasks such as tracking Rebecq et al. [2016]; Gehrig et al. [2018], deblurring Pan et al. [2019c], and SLAM Kim et al. [2016]; Kueng et al. [2016b]; Vidal et al. [2018]. However, standard vision algorithms cannot be applied to event cameras directly. Hence, new methods are required to be tailored to event cameras and unlock their potential. In this chapter, we aim to show how events can improve flow estimates, even with a blurred image.

Optical flow estimation is an active topic in the computer vision commu-

nity and serves as the backbone for event-based moving object segmentation Stoffregen et al. [2019], human pose estimation Calabrese et al. [2019], and action recognition Amir et al. [2017]. Traditional flow estimation approaches Horn and Schunck [1981]; Jason et al. [2016]; Yin and Shi [2018] are proposed based on the brightness consistency assumption for corresponding pixels across the image pair, and cannot handle the asynchronous event data Gallego et al. [2019]. A common trend Bardow et al. [2016]; Gallego et al. [2018]; Zhu et al. [2019]; Gehrig et al. [2019b] to estimate flow is from events only. However, events are sparse spatially, flow computed at regions with no events are less reliable than those computed at regions with events (i.e., ,at edges) Liu and Delbruck [2018]. Hence, several methods tends to fuse the intensity information and events Bardow et al. [2016]; Barranco et al. [2014] to estimate flow.

To this end, we aim to use the output of DAVIS, which is events and intensity images, to improve optical flow estimates. A straightforward idea is to reconstruct images from events Pan et al. [2019c]; Rebecq et al. [2019], and then compute flow directly from the reconstructed image. While the generated flow is noisy inherently, it shows the potential to estimate flow by using the image and its event streams (seeing Fig. 7.3). Unfortunately, this approach neglects the inherent connection between flow and events. Thus, we introduce an event-based photometric consistency in our model to encode the relation between flow and event data. Different from Zhu et al. [2018a] that exploit images as the supervision signal for a self-supervised learning framework only, we fully explore the relation between events and flow to formulate our model.

On the other hand, while intensity images are effective for flow estimation, output images of event cameras tend to contain blur artefacts due to dynamic visual environment. It makes flow estimation even more challenging as brightness constancy may not hold for blurred images (seeing Fig. 7.1). Unlike existing methods, we explore the relationship between flow and blurred image formation which provides more constraints to flow estimation. In a nutshell, our model shows the potential of event cameras for single image flow estimation, and can also work under blurred condition by joint sharp image and optical flow estimation.

In summary, our main contributions are

1) We propose a method for optical flow estimation from a single image (blurred potentially) and its event data for the event camera (DAVIS).

2) We introduce an *event-based brightness constancy* constraint on absolute intensity to encode the relation between optical flow and the event data. Besides, we use the blur formation model in our objective function to handle optical flow estimation on the blurred image.

3) Experimental results in both real and synthetic datasets show our method can successfully handle complex real-world flow estimation, depicting fast-moving objects, camera motions, and uncontrolled lighting conditions.

7.3 Related Work

In this section, we review works for flow estimation from event cameras, images, and event-based image reconstruction which could be used for flow estimation. We further discuss a few works for image deblurring related to flow.

Event camera based flow estimation. Benosman et al. [2012] propose an adaptation of the gradient-based Lucas-Kanade algorithm based on DVS. In Benosman et al. [2013], they assume that the flow orientation and amplitude can be estimated using a local differential approach on the surface defined by coactive events. They work well for sharp edges and monochromatic blocks but fail with dense textures, thin lines, and more complicated scenes. Barranco et al. [2015] propose a more expensive phase-based method for high-frequency texture regions and trying to reconstruct the intensity signals to avoid the problem with textured edges. Bardow et al. [2016] jointly reconstruct intensity image and estimate flow based on events by minimizing their objective function. However, accuracy relies on the quality of the reconstructed image. Gallego et al. [2018] present a unifying framework to estimate flow by finding the point trajectories on each image plane that are best aligned with events. Zhu propose EV-FlowNet Zhu et al. [2018a], an event-based flow estimation approach using a self-supervised deep learning pipeline. The event data are represented as 2D frames to feed the network. While images from the sensor are used as a supervision signal, the blur effect is ignored which is shown to be useful for flow estimation in our framework. In Zhu et al. [2019], they further use another event format to train two networks to predict flow, camera ego-motion, and depth for static scenery. Then, they use predictions to remove motion blur from event streams which shows the potential of blurring to improve the flow estimate accuracy. However, flow computed at those constant brightness regions is still less reliable.

Image-based flow estimation. One promising direction is to learn optical flow with CNNs Dosovitskiy et al. [2015]; Jason et al. [2016]; Yin and Shi [2018] by video. FlowNet 2.0 Ilg et al. [2017] develops a stacked architecture that includes warping of the second image with the intermediate flow. PWC-Net Sun et al. [2018] uses the current flow estimate to warp the CNN features of the second image. It then uses the warped features and features of the first image to construct a cost volume to estimate flow. SelFlow Liu et al. [2019b] is based on distilling reliable flow estimations from non-occluded pixels, and

using these predictions to guide optical flow learning for hallucinated occlusions. Several deep learning-driven works attempt to use a single image to estimate flow Walker et al. [2015]; Rosello [2016]; Endo et al. [2019]. Walker et al. [2015] use CNN to predict dense flow, while they assume the image is static.

Event-based image reconstruction. Image reconstruction Rebecq et al. [2019]; Wang et al. [2019]; Pan et al. [2020a] from events can be treated as the data preparation step for traditional image-based flow estimation methods. However, this ignores that the event can contribute to flow estimation. To reconstruct the image with more details, several methods attempt to combine events with intensity images Brandli et al. [2014b]; Scheerlinck et al. [2018]; Pan et al. [2019c]. Pan et al. [2019c] propose an Event-based Double Integral (EDI) model to fuse an image with its events to reconstruct a high frame rate video. In our paper, we combine the EDI model and state-of-the-art optical flow estimation methods to serve as baselines of our approach.

Image deblurring. As the flow accuracy highly depends on the quality of the image, a better-restored image also relies on the quality of the estimated flow. Researchers attempt to use flow to estimate the spatial-varying blur kernel and then restore images Xu et al. [2015]; Kim and Lee [2014, 2015]; Sellent et al. [2016]; Pan et al. [2017b]; Pan et al. [2020]; Pan et al. [2019a]. Recently, learning-based methods have brought significant improvements in image deblurring Gong et al. [2017b]; Nah et al. [2019b]; Zhou et al. [2019b]. Gong et al. [2017b] directly estimate flow from a blurred image by a fully-convolutional neural network (FCN) and recover the sharp image from the estimated flow. It is still a challenging problem for dynamic scene deblurring. Our estimated flow from a single image and events are more robust and the model generalizes well to handle blurred images from complex scenery.

7.4 Variational Approach

We start with reviewing variational approaches for optical flow estimation from a pair of images. Define as $\mathbf{u} = (u, v)$ to be an optical flow field, and $\mathbf{u}(\mathbf{x}) = (u_{\mathbf{x}}, v_{\mathbf{x}})^{\mathrm{T}}$ its value at a given pixel $\mathbf{x}$. From a reference time f to t, the brightness constancy can be written as

$$\mathbf{L}(\mathbf{x}, f) = \mathbf{L}(\mathbf{x} + \mathbf{u}(\mathbf{x}), t) \,, \tag{7.1}$$

where $\mathbf{u} \in \mathbb{R}^{H \times W \times 2}$, and $\mathbf{L} \in \mathbb{R}^{H \times W}$ is the latent image. Here, H, W are the image size. Let the intensity of pixel $\mathbf{x} = (x, y)^{\mathrm{T}}$ at time f be denoted by $\mathbf{L}(\mathbf{x}, f)$. As equation (7.1) is under-determined, regularization terms are introduced to solve optical flow. Horn and Schunck Horn and Schunck [1981]

studied a variational formulation of the problem,

$$\min_{\mathbf{u}} \int_{\Omega} \|\nabla \mathbf{u}(\mathbf{x})\|^2 \, d\mathbf{x} + \int_{\Omega} (\mathbf{L}(\mathbf{x}, f) - \mathbf{L}(\mathbf{x} + \mathbf{u}(\mathbf{x}), t))^2 \, d\mathbf{x} \,, \tag{7.2}$$

where $\|\cdot\|$ is the standard l^2 norm, Ω denotes the image domain, and $\nabla \mathbf{u} \in \mathbb{R}^{H \times W \times 4}$. The first term penalizes high variations in $\mathbf{u}$ to obtain smooth optical flow fields. The second term enforces the brightness constancy constraint (BCC). Here, we denote $\nabla \mathbf{u}(\mathbf{x})$ as

$$\nabla \mathbf{u}(\mathbf{x}) = \left(\frac{\partial u(\mathbf{x})}{\partial x}, \frac{\partial u(\mathbf{x})}{\partial y}, \frac{\partial v(\mathbf{x})}{\partial x}, \frac{\partial v(\mathbf{x})}{\partial y} \right)^{\mathrm{T}} ,$$

where we denote $\nabla \mathbf{u}(\mathbf{x}) = (u_{\mathbf{x}}^{(x)}, u_{\mathbf{x}}^{(y)}, v_{\mathbf{x}}^{(x)}, v_{\mathbf{x}}^{(y)})^{\mathrm{T}}$ for short. Note that (here and elsewhere) superscripts in brackets represent differentiation with respect to x or y.

7.5 Event-based approach

We aim to estimate flow from a set of events (from time f to t) and a single corresponding gray-scale image (blurred potentially) taken by DAVIS. It is noteworthy that flow is defined as a continuously varying motion field at a flexible time slice of event data, which is different from the traditional flow defined based on the image frame rate.

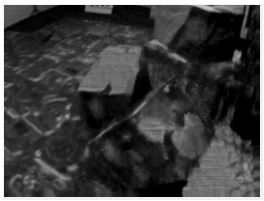
To compute flow from events, a potential solution is to estimate flow from the reconstructed images based on event cameras Pan et al. [2019c]. However, it ignores that events can contribute to flow estimation. In contrast, we observe that events provide correspondences of pixels across time, which implicitly defines flows for pixels with events. It suggests that we should model events directly in our flow estimation framework. Meanwhile, the intensity image is another output of DAVIS. However, it is likely blurred due to high dynamics in the scene. As shown in Gong et al. [2017b], the blur artifacts in the image provides useful information for flow estimation.

We therefore propose to jointly estimate flow $\mathbf{u}$ and the latent image $\mathbf{L}$ by enforcing the brightness constancy by events and the blurred image formation model. In particular, our energy minimization model is formulated as:

$$\min_{\mathbf{L},\mathbf{u}} \mu_1 \phi_{\text{eve}}(\mathbf{L}, \mathbf{u}) + \mu_2 \phi_{\text{blur}}(\mathbf{L}, \mathbf{u}) + \phi_{\text{flow}}(\nabla \mathbf{u}) + \phi_{\text{im}}(\nabla \mathbf{L}) \,, \tag{7.3}$$

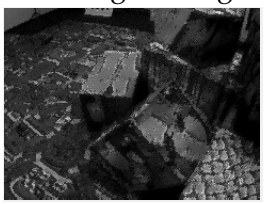
where μ_1 and μ_2 are weight parameters, ϕ_{eve} enforces the BCC by event, ϕ_{blur} enforces the blurred image formation process, ϕ_{flow} and ϕ_{im} enforces the smoothness of the estimated flow and latent image. In following sections, we include details for the objective function in Eq. (7.3).

(a) Blurred image



(b) Deblurred image Zhang et al. [2019]

(c) Deblurred image Pan et al. [2019c]



(d) Ours

Figure 7.2: *An example of our deblurring result on the real dataset Mueggler et al. [2017]. (a) The blurred image. (b) Deblurred by Zhang et al. [2019]. (c) Deblurred by EDI Pan et al. [2019c]. (d) Ours. (Best viewed on screen).*

7.5.1 Brightness Constancy by Event Data $\phi_{\mathbf{eve}}$

In case of the output data from DAVIS, we represent Eq. (7.1) in a different way. Besides images, each *event* is denoted by $(\mathbf{x}, t, \sigma)$. Polarity $\sigma = \pm 1$ denotes the direction of the intensity change. An event is fired when a change in the log intensity exceeds a threshold c.

$$\left|\log(\mathbf{L}(\mathbf{x}, t)) - \log(\mathbf{L}(\mathbf{x}, t_{ref})\right| \geq c \ . \tag{7.4}$$

Here, t is the current timestamp and t_{ref} is the timestamp of the previous event. When an event is triggered, t_{ref} and $\mathbf{L}(\mathbf{x}, t_{ref})$ at that pixel is updated to a new timestamp and a new intensity level. Following the EDI model Pan et al. [2019c], we represent the neighbouring image as

$$\mathbf{L}(\mathbf{x}, t) \ = \mathbf{L}(\mathbf{x}, f) \ \exp(c\, \mathbf{E}(\mathbf{x}, t)) \ , \tag{7.5}$$

where $\mathbf{E}(\mathbf{x}, t)$ is the integration of events between time f and t at a given pixel $\mathbf{x}$, and we dub $\mathbf{E}(t)$ as the event frame.

Assume the motion between $\triangle t = t - f$ is small. We adopt a first-order Taylor expansion to the right-hand side of Eq. (7.1) and obtain its approximation

$$\begin{aligned} &\mathbf{L}(\mathbf{x} + \mathbf{u}(\mathbf{x}), f + \triangle t) \\ &\quad \approx \mathbf{L}(\mathbf{x}, f) + u_{\mathbf{x}} \mathbf{L}(\mathbf{x}, f)^{(x)} + v_{\mathbf{x}} \mathbf{L}(\mathbf{x}, f)^{(y)} + \triangle t \ \mathbf{L}(\mathbf{x}, f)^{(t)} \\ &\quad = u_{\mathbf{x}} \mathbf{L}(\mathbf{x}, f)^{(x)} + v_{\mathbf{x}} \mathbf{L}(\mathbf{x}, f)^{(y)} + \mathbf{L}(\mathbf{x}, t) \ . \end{aligned} \tag{7.6}$$

Back to the left-hand side of Eq. (7.1), we have

$$\mathbf{L}(\mathbf{x}, f) \approx u_{\mathbf{x}}\mathbf{L}(\mathbf{x}, f)^{(x)} + v_{\mathbf{x}}\mathbf{L}(\mathbf{x}, f)^{(y)} + \mathbf{L}(\mathbf{x}, t) \ . \tag{7.7}$$

With the event model in Eq. (7.5), we can form the latent image as,

$$\begin{aligned}\mathbf{L}(\mathbf{x}, f) \approx\ & u_{\mathbf{x}}\mathbf{L}(\mathbf{x}, f)^{(x)} + v_{\mathbf{x}}\mathbf{L}(\mathbf{x}, f)^{(y)} \\ & + \mathbf{L}(\mathbf{x}, f) \exp\left(c\, \mathbf{E}(\mathbf{x}, t)\right) .\end{aligned}$$

Let $\nabla\mathbf{L}(\mathbf{x}, f) = (\mathbf{L}(\mathbf{x}, f)^{(x)}, \mathbf{L}(\mathbf{x}, f)^{(y)})^{\mathrm{T}}$, we therefore write the event-based photometric constancy constraint as

$$\boxed{\begin{aligned}\phi_{\mathrm{eve}}(\mathbf{L}, \mathbf{u}) = \sum_{\mathbf{x}\in\Omega} \| & \mathbf{L}(\mathbf{x}, f)(\exp(c\, \mathbf{E}(\mathbf{x}, t)) - 1) \\ & + [u_{\mathbf{x}}, v_{\mathbf{x}}]^{\mathrm{T}} \nabla\mathbf{L}(\mathbf{x}, f)\|_1 \ .\end{aligned}} \tag{7.8}$$

Different with Bardow et al. [2016]; Gehrig et al. [2019b]; Bryner et al. [2019] defining the brightness constancy constraint in the log space, we encode the relation between optical flow and events by our event-based brightness constancy constraint in terms of the original absolute intensity space.

7.5.2 Blur Image Formation Constraint $\phi_{\mathbf{blur}}$

In addition to event streams, DAVIS can provide intensity images at a much lower temporal rate than events. Images may suffer from motion blur due to the relative motion between the camera and objects. A general model of blur image formation is given by

$$\mathbf{B} = \mathbf{k} \otimes \mathbf{L}(f) \ , \tag{7.9}$$

where $\mathbf{B} \in \mathbb{R}^{H\times W}$ is the blurred image, $\otimes$ is the convolution operator, and $\mathbf{k}$ denotes the blur kernel. For a dynamic scenario, the spatially variant blur kernel is, in principle, defined for each pixel. Then

$$\mathbf{B}(\mathbf{x}) = \mathbf{k}(\mathbf{x}) \otimes \mathbf{L}(\mathbf{x}) \ . \tag{7.10}$$

We omit f in the following sections. The convolution of the two matrices is defined as,

$$\begin{aligned}\mathbf{B}(\mathbf{x}) &= \sum_{\mathbf{y}\in\Omega} \mathbf{k}(\mathbf{y})\mathbf{L}(\mathbf{x} - \mathbf{y}) \\ &= \sum_{\mathbf{y}\in\Omega} \mathbf{k}_{\mathbf{u}'(\mathbf{x})}(\mathbf{y})\mathbf{L}(\mathbf{x} - \mathbf{y}) \ ,\end{aligned} \tag{7.11}$$

where $\mathbf{x}, \mathbf{y} \in \Omega$, and $\mathbf{k}_{\mathbf{u}'(\mathbf{x})} \in \mathbb{R}^{H \times W}$ is the kernel map for each pixel. We use the subscript $\mathbf{u}'(\mathbf{x})$ to denote the index of $\mathbf{k}$ for pixel $\mathbf{x}$, and $\mathbf{k}_{\mathbf{u}'(\mathbf{x})}(\mathbf{y})$ is expressed as

$$k_{\mathbf{u}'(\mathbf{x})}(\mathbf{y}) = \begin{cases} \dfrac{1}{|\mathbf{u}'(\mathbf{x})|}, & \text{if } \mathbf{y} = \alpha \mathbf{u}'(\mathbf{x}),\ |\alpha| \leq \dfrac{1}{2} \\ \mathbf{0}, & \text{otherwise}, \end{cases} \tag{7.12}$$

where $\mathbf{u}'(\mathbf{x}) = \lambda \mathbf{u}(\mathbf{x})$ denotes flow during the exposure time T, and $\lambda = T/\triangle t$. It follows our assumption that flow during a small time interval has a constant velocity. Furthermore, each element of the kernel is non-negative and the sum of it is equal to one. Note that the kernel defined in Eq. (7.12) allows us to handle blurred images with a long exposure time T, as well as sharp images with short exposure time. When T is small, θ is small enough to result in a Dirac delta function as a blur kernel (e.g., convolving a signal with the delta function leaves the signal unchanged). The blur image formation constraint is denoted as

$$\boxed{\phi_{\text{blur}}(\mathbf{L}, \mathbf{u}) = \sum_{\mathbf{x},\mathbf{y} \in \Omega} \|\mathbf{k}_{\mathbf{u}'(\mathbf{x})}(\mathbf{y}) \mathbf{L}(\mathbf{x} - \mathbf{y}) - \mathbf{B}(\mathbf{x})\|^2 ,} \tag{7.13}$$

which can handle the blurred and sharp image in a unified framework.

7.5.3 Smoothness Term ϕ_{flow}, and ϕ_{im}

In general, conventional flow estimation models assume that flow vectors vary smoothly and have sparse discontinuities at edges of the image Kim et al. [2013]. Smoothness terms aim to regularize flow and the image by minimizing the difference between neighbouring pixels. For any pixel $\mathbf{x}$, vector $w(\mathbf{x}) = (w_{\mathbf{x}}^x, w_{\mathbf{x}}^y) \in \mathbb{R}^2$, and $\nabla \mathbf{u}(\mathbf{x}) \in \mathbb{R}^4$, define

$$w(\mathbf{x}) \nabla \mathbf{u}(\mathbf{x}) = \left(w_{\mathbf{x}}^x u_{\mathbf{x}}^{(x)}, w_{\mathbf{x}}^y u_{\mathbf{x}}^{(y)}, w_{\mathbf{x}}^x v_{\mathbf{x}}^{(x)}, w_{\mathbf{x}}^y v_{\mathbf{x}}^{(y)} \right)^{\mathrm{T}} .$$

Putting all the pixels together, we define $w \nabla \mathbf{u}$, where $w \in \mathbb{R}^{H \times W \times 2}$ and $\nabla \mathbf{u} \in \mathbb{R}^{H \times W \times 4}$.

Our flow cost is defined as

$$\boxed{\phi_{\text{flow}}(\nabla \mathbf{u}) = \|w \nabla \mathbf{u}\|_{1,2} = \sum_{\mathbf{x} \in \Omega} \|w(\mathbf{x}) \nabla \mathbf{u}(\mathbf{x})\| ,} \tag{7.14}$$

which is a mixed 1-2 norm (sum of 2-norms). We choose weight w where

$$w^x = \mu_3 \exp(-(\hat{\mathbf{L}}^{(x)} / \mu_4)^2) , \tag{7.15}$$

and similarly w^y, constants μ_3 and μ_4 are weight parameters, and $\hat{\mathbf{L}}$ is the input image of our optimization framework. In addition, we define an image smoothness term as

$$\boxed{\phi_{\text{im}}(\nabla \mathbf{L}) = \sum_{\mathbf{x}\in\Omega} \|\nabla \mathbf{L}(\mathbf{x})\|_1 \ .} \tag{7.16}$$

7.6 Optimization

Clearly, Eq. (7.3) is non-convex with respect to $\mathbf{u}$, and $\mathbf{L}$. Therefore, we perform the optimization over one variable at a time and optimize all parameters in an alternating manner.

- Fix latent image $\mathbf{L}$, and compute optical flow by optimizing Eq. (7.17) (See Section 7.6.1).
- Fix optical flow $\mathbf{u}$, and compute the latent image by optimizing Eq. (7.24) (See Section 7.6.2).

Here, we use the primal-dual algorithm Pock et al. [2009b,a]; Chambolle and Pock [2011] for its optimal convergence. In the following section, we describe details for each optimization step.

7.6.1 Optical Flow Estimation

We fix the image, namely $\mathbf{L} = \hat{\mathbf{L}}$, and Eq. (7.3) reduces to

$$\min_{\mathbf{u}} \underbrace{\mu_1 \phi_{\text{eve}}(\mathbf{u}) + \mu_2 \phi_{\text{blur}}(\mathbf{u})}_{G(\mathbf{u})} + \underbrace{\phi_{\text{flow}}(\nabla \mathbf{u})}_{F(K\mathbf{u})} \ , \tag{7.17}$$

where $\phi_{\text{eve}}(\mathbf{u})$ and $\phi_{\text{flow}}(\nabla \mathbf{u})$ are convex, while $\phi_{\text{blur}}(\mathbf{u})$ is non-convex. As shown, we separate Eq. (7.17) into G and F, where $K\mathbf{u} = w\nabla \mathbf{u}$ is a linear function and $F(K\mathbf{u}) = \|K\mathbf{u}\|_{1,2} = \phi_{\text{flow}}(\nabla \mathbf{u})$. Let $\mathbf{u} \in X = \mathbb{R}^{2N}$, and $\nabla \mathbf{u} \in Y = \mathbb{R}^{4N}$, so $G : X \to \mathbb{R}$, and $F : Y \to \mathbb{R}$, where $N = HW$ is the number of pixels. In follows, we treat $\mathbf{u}$, $\nabla \mathbf{u}$ as vectors. The basis of the primal-dual formulation is to replace F in Eq. (7.17) by its double Fenchel dual F^{**}, so it becomes $\min_{\mathbf{u}\in X}(G(\mathbf{u}) + F^{**}(K\mathbf{u}))$, which is

$$\min_{\mathbf{u}\in X} \left(G(\mathbf{u}) + \max_{\mathbf{p}\in Y} \langle K\mathbf{u} \ , \mathbf{p} \rangle_X - F^*(\mathbf{p}) \right) \ . \tag{7.18}$$

Recall that the Fenchel dual (convex conjugate) F^* of function F is defined as

$$F^*(\mathbf{q}) = \sup_{\mathbf{p}\in Y} \left(\langle \mathbf{p}, \mathbf{q} \rangle - F(\mathbf{p}) \right) \ , \tag{7.19}$$

and that $F = F^{**}$ if F is a convex function (a norm is convex). The primal-dual algorithm of Chambolle and Pock [2011] consists of iterations starting from initial estimates $\mathbf{u}^0$, $\mathbf{p}^0$ and $\bar{\mathbf{u}}^0 = \mathbf{u}^0$:

$$\begin{aligned} \mathbf{p}^{n+1} &= \mathcal{P}_{F^*}(\mathbf{p}^n + \sigma K \bar{\mathbf{u}}^n) \\ \mathbf{u}^{n+1} &= \mathcal{P}_G(\mathbf{u}^n - \tau K^* \mathbf{p}^{n+1}) \\ \bar{\mathbf{u}}^{n+1} &= \mathbf{u}^{n+1} + \theta(\mathbf{u}^{n+1} - \mathbf{u}^n) \ . \end{aligned} \tag{7.20}$$

Here σ and τ are weight parameters, and $\mathcal{P}(\cdot)$ is the proximal operator

$$\mathcal{P}_g(x) = \arg\min_y (2g(y) + \|y - x\|^2) \ .$$

The hyperparameter θ is a number that controls the degree of 'extrapolation'. We use $\theta = 1$. We now discuss each step of this algorithm in the present case.

Updating p. It is well known that the Fenchel dual of a norm is the indicator function of the unit ball in the dual norm. In this case, $F^*(\cdot)$ is a mixed norm $\|\cdot\|_{1,2}$, and its dual is a norm $\|\cdot\|_{\infty,2}$. The indicator function is therefore a product B^N of N Euclidean 2-balls (each in $\mathbb{R}^4$). More precisely

$$F^*(\mathbf{p}) = \begin{cases} 0, & \text{if } \|\mathbf{p}_\mathbf{x}\| \leq 1 \text{ for all } \mathbf{x} \\ +\infty, & \text{otherwise} \ . \end{cases} \tag{7.21}$$

The proximal operator $\mathcal{P}_{F^*}$ is therefore given by

$$\begin{aligned} F^*(\bar{\mathbf{p}}) &= \arg\min_{\mathbf{p} \in Y} \left(2F^*(\mathbf{p}) + \|\bar{\mathbf{p}} - \mathbf{p}\|^2 \right) \\ &= \arg\min_{\mathbf{p} \in B^N} \|\bar{\mathbf{p}} - \mathbf{p}\|^2 \ . \end{aligned} \tag{7.22}$$

In other words, each $\bar{\mathbf{p}}_\mathbf{x}$ is projected to the nearest point in the unit ball, given by $\bar{\mathbf{p}}_\mathbf{x} / (\max(1, \|\bar{\mathbf{p}}_\mathbf{x}\|))$.

Updating u. The update equation from Eq. (7.20) is

$$\begin{aligned} \bar{\mathbf{u}} &= \mathbf{u}^n - \tau K^* \mathbf{p}^{n+1} \\ \mathbf{u}^{n+1} &= \mathcal{P}_{\tau G}(\bar{\mathbf{u}}) = \arg\min_\mathbf{u} \left(2\tau G(\mathbf{u}) + \|\mathbf{u} - \bar{\mathbf{u}}\|^2 \right) \ . \end{aligned}$$

(Note we use $\mathcal{P}_{\tau G}$ instead of $\mathcal{P}_G$). Minimizing by taking derivatives gives $\mathbf{u} = \bar{\mathbf{u}} - \tau \nabla G(\mathbf{u})$. We make the simplifying assumption that G is locally approximated to first order, and so $\nabla G(\mathbf{u}) = \nabla G(\mathbf{u}^n)$, which leads to the update step

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \tau \left(\nabla G(\mathbf{u}^n) + K^* \mathbf{p}^{n+1} \right) \ , \tag{7.23}$$

which is simply gradient descent of Eq. (7.18), fixing $\mathbf{p} = \mathbf{p}^{n+1}$. We obtain Algorithm 3.

Algorithm 3: Primal-Dual Minimization - Flow

Initialization: Choose τ, $\sigma > 0$, $n = 0$, and set $\bar{\mathbf{u}}^0 = \mathbf{u}^0$.
Iterations : Update $\mathbf{u}^n$, $\mathbf{p}^n$, $\bar{\mathbf{u}}^n$ as follows

```
while n < 20 do
    Dual ascent in p
    p̄ = p^n + σK ū^n,  p_x^{n+1} = p̄_x / max(1, ||p̄_x||) ∀x
    Primal descent in u
    u^{n+1} = u^n − τ(G(u^n) + K* p^{n+1})
    Extrapolation step
    ū^{n+1} = u^{n+1} + (u^{n+1} − u^n)
    n = n + 1
end
```

7.6.2 Deblurring

We fix optical flow, namely $\mathbf{u} = \hat{\mathbf{u}}$, and Eq. (7.3) reduces to

$$\min_{\mathbf{L}} \underbrace{\phi_{\text{im}}(\nabla \mathbf{L})}_{F_1(\nabla \mathbf{L})} + \underbrace{\mu_1 \phi_{\text{eve}}(\mathbf{L})}_{F_2(K\mathbf{L})} + \underbrace{\mu_2 \phi_{\text{blur}}(\mathbf{L})}_{G(\mathbf{L})} \ . \tag{7.24}$$

The convex conjugate F^* is defined as,

$$F^*(\mathbf{p}, \mathbf{q}) = F_1^*(\mathbf{p}) + F_2^*(\mathbf{q}) \ , \tag{7.25}$$

where $\mathbf{p} \in \mathbb{R}^{2N}$, and $\mathbf{q} \in \mathbb{R}^N$. Here, $\nabla \mathbf{L} \in \mathbb{R}^{2N}$. The primal-dual update process is expressed as follows,

$$\begin{aligned} \mathbf{p}^{n+1} &= \frac{\mathbf{p}^n + \gamma \nabla \bar{\mathbf{L}}^n}{\max(1, \mathbf{J}(\mathbf{p}^n + \gamma \nabla \bar{\mathbf{L}}^n))}, \\ \mathbf{q}^{n+1} &= \frac{\mathbf{q}^n + \gamma(\theta_2 \bar{\mathbf{L}}^n + [u,\ v]^{\mathrm{T}} \nabla \bar{\mathbf{L}}^n)}{\max(1, \mathbf{J}(\mathbf{q}^n + \gamma(\theta_2 \bar{\mathbf{L}}^n + [u,\ v]^{\mathrm{T}} \nabla \bar{\mathbf{L}}^n)))} \ , \end{aligned} \tag{7.26}$$

where η, γ are weight factors, and $\theta_2 = \exp(c\mathbf{E}(t)) - 1$.

$$\mathbf{L}^{n+1} = \mathcal{P}_{\eta G}(\bar{\mathbf{L}}) = \arg\min_{\mathbf{L}} \left(2\eta G(\mathbf{L}) + \|\mathbf{L} - \bar{\mathbf{L}}\|^2 \right) \ , \tag{7.27}$$

where $\bar{\mathbf{L}} = \mathbf{L}^n - \eta(\nabla^* \mathbf{p}^{n+1} + K^* \mathbf{q}^{n+1})$. We obtain Algorithm 4 for the minimization of the proposed energy function (7.24).

Algorithm 4: Primal-Dual Minimization - Deblurring

Initialization: Choose γ, $\eta > 0$, $n = 0$, and set $\bar{\mathbf{L}}^0 = \mathbf{L}^0$.
Iterations : Update $\mathbf{L}^n$, $\mathbf{p}^n$, $\mathbf{q}^n$ as follows

while $n < 5$ **do**
- Dual ascent in $\mathbf{p}$, $\mathbf{q}$
- $\bar{\mathbf{p}} = \mathbf{p}^n + \gamma \nabla \bar{\mathbf{L}}^n, \ \bar{\mathbf{q}} = \mathbf{q}^n + \gamma(\theta_2 \bar{\mathbf{L}}^n + [u, \ v]^{\mathrm{T}} \nabla \bar{\mathbf{L}}^n)$
- $\mathbf{p}_{\mathbf{x}}^{n+1} = \bar{\mathbf{p}}_{\mathbf{x}} / \max(1, \mathbf{J}(\bar{\mathbf{p}}_{\mathbf{x}})) \ \forall \mathbf{x}$
- $\mathbf{q}_{\mathbf{x}}^{n+1} = \bar{\mathbf{q}}_{\mathbf{x}} / \max(1, \mathbf{J}(\bar{\mathbf{q}}_{\mathbf{x}})) \ \forall \mathbf{x}$
- Primal descent in $\mathbf{L}$
- $\bar{\mathbf{L}} = \mathbf{L}^n - \eta(\nabla^* \mathbf{p}^{n+1} + K^* \mathbf{q}^{n+1}), \ \mathbf{L}^{n+1} = \mathcal{P}_{\eta G}(\bar{\mathbf{L}})$
- Extrapolation step
- $\bar{\mathbf{L}}^{n+1} = \mathbf{L}^{n+1} + (\mathbf{L}^{n+1} - \mathbf{L}^n)$
- $n = n + 1$

end

7.7 Experiments

7.7.1 Experimental Setup

Real dataset. We evaluate our method on three public real event datasets, namely, Multi-vehicle Stereo Event Camera dataset (MVSEC) Zhu et al. [2018b], Event-Camera dataset (ECD) Mueggler et al. [2017], and Blurred Event Dataset (BED) Scheerlinck et al. [2018]; Pan et al. [2019c]. MVSEC provides a collection of sequences captured by DAVIS for high-speed vehicles with ground truth optical flow.

Synthetic dataset. For quantitative comparisons on optical flow, we build a synthetic dataset based on Sintel Butler et al. [2012] with images of size 1024×436, which uses the event simulator ESIM Rebecq et al. [2018] to generate event streams. While Sintel provides a blurred dataset, it mainly focuses on out of focus blur instead of motion blur. Therefore, it is not suitable for the evaluation of deblurring. To provide a quantitative deblurring comparison, we generate another synthetic dataset with events and motion blur, based on the real GoPro video dataset Nah et al. [2017], where the image size is 1280×720. It has ground-truth latent images and associated motion blurred images. We additionally use PWC-Net to estimate flow from sharp images as the ground-truth for flow evaluation.

Evaluations. For the evaluation of flow estimation results, we use error metrics, such as Mean Square Error (MSE), Average Endpoint Error (AEE), and Flow Error metric (FE).

$$\mathrm{AEE} = \frac{\sum_{\Omega} \|\mathbf{u}_{\mathrm{est}} - \mathbf{u}_{\mathrm{gt}}\|_1}{2HW}, \ \mathrm{MSE} = \frac{\sum_{\Omega} \|\mathbf{u}_{\mathrm{est}} - \mathbf{u}_{\mathrm{gt}}\|^2}{2HW}.$$

Table 7.1: *Results on the MVSEC Zhu et al. [2018a] and Sintel dataset Butler et al. [2012]. We evaluate optical flow by Mean Square Error (MSE), Average Endpoint Error (AEE) and Flow Error metric (FE). The first column 'GT images' means we use two ground-truth images to estimate flow. 'EDI image' means we use two reconstruct images to estimate flow by EDI model. EV-FlowNet Zhu et al. [2018a] provides a pre-trained model with cropped images (256×256) and events. Thus, we only show their results that comparing with the cropped ground-truth flow. Our model achieves competitive results compared with state-of-the-art methods. Our 'AEE' and 'FE' metric dropped two times as much as others.*

MVSEC dataset Zhu et al. [2018a]							
Input	GT images		EDI images and events		Events		
	SelFlow	PWC-Net	SelFlow	PWC-Net	EV-FlowNet	Zhu	Ours
AEE	0.5365	0.4392	1.4232	1.3677	1.3112	**0.6975**	0.9296
MSE	0.3708	0.1989	1.7882	1.6135	1.3501	-	**0.8700**
FE (%)	0.5163	0.0938	2.5079	2.4927	1.1038	1.7500	**0.4768**
Sintel dataset Butler et al. [2012]							
AEE	0.1191	0.1713	1.3895	1.5138	2.9714	-	**1.0735**
MSE	0.3645	0.5979	6.2693	7.6105	21.4982	-	**3.2342**
FE (%)	0.8155	1.1922	22.6290	21.9625	49.0136	-	**14.9061**

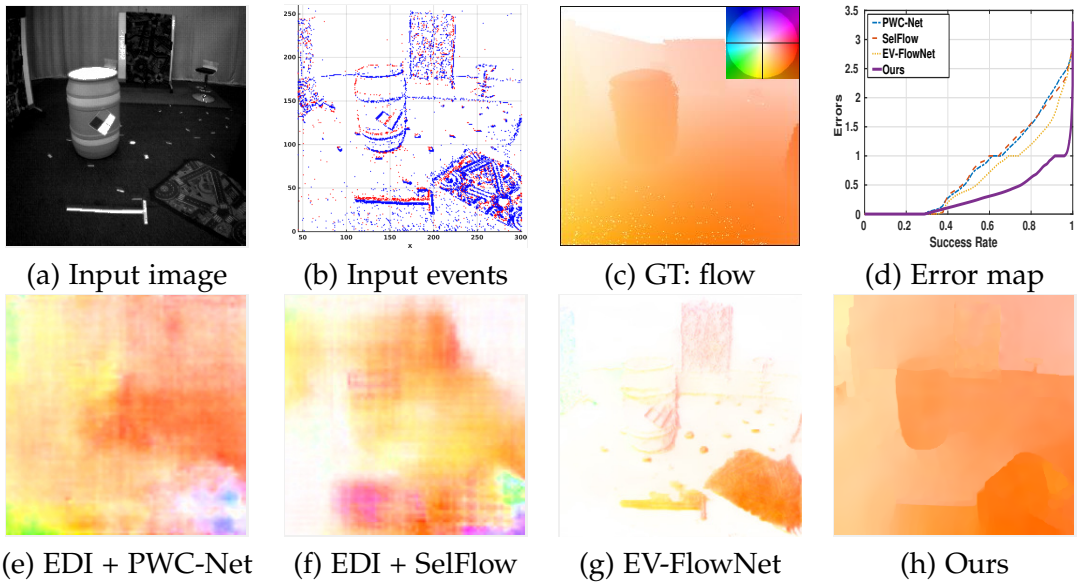
Table 7.2: *Ablation Study based on Sintel Dataset Butler et al. [2012].*

	without ϕ_{eve}	without ϕ_{blur}
AEE	2.3941	2.2594
MSE	5.3506	9.5267
FE (%)	18.0525	45.4516

FE metric is computed by counting the number of pixels having errors more than 3 pixels and 5% of its ground-truth over pixels with valid ground truth flow. We adopt the PSNR to evaluate deblurred images. The error map shows the distribution of the endpoint error of measurements compared with the ground-truth flow and the success rate is defined as the percentage of results with errors below a threshold.

Baseline methods. For optical flow, we compare with state-of-the-art event only based methods EV-FlowNet Zhu et al. [2018a], and Zhu Zhu et al. [2019]. Then, we compare with the state-of-the-art video (with the label 'GT images'.) only based method SelFlow Liu et al. [2019b], and PWC-Net Sun et al. [2018]. In addition, we build a two-step (event + image) framework as a baseline approach, which is 'EDI + SelFlow' and 'EDI + PWC-Net'. The two-step framework first use the image reconstruction method EDI Pan et al. [2019c] to restore intensity images, then applying flow estimation methods Sun et al. [2018]; Liu et al. [2019b] to the restored images to estimate flow. We compare our deblurring results with the state-of-the-art event-based deblurring approach Pan et al. [2019c] and blind deblurring methods Zhang et al. [2019]; Tao et al. [2018]; Gong et al. [2017b].

Implementation details. For all our real experiments, the image and events



(a) Input image (b) Input events (c) GT: flow (d) Error map

(e) EDI + PWC-Net (f) EDI + SelFlow (g) EV-FlowNet (h) Ours

Figure 7.3: *Results of our method compared with state-of-the-art methods on real dataset Zhu et al. [2018a]. (a) Input image. (b) Input events. (c) Ground-truth optical flow and the colour coded optical flow on the left corner. (d) Error Map shows the distribution of the Endpoint Error of estimates compared with the ground-truth flow. (e) Baseline: Flow result by Sun et al. [2018] based on two reconstructed images. The reconstructed image is estimated by EDI model Pan et al. [2019c] from a single image and its events. (f) Baseline: Flow result by Liu et al. [2019b] based on two reconstructed images. (g) Flow result by Zhu et al. [2018a] based on images and events. (h) Ours, by using an image and events as input. (Best viewed on screen).*

are from DAVIS. The framework is implemented by using MATLAB® with C++ wrappers. It takes around 20 seconds to process a real image (size 346 × 260) from DAVIS on a single i7 core running at 3.6 GHz.

7.7.2 Experimental Results

We compare our results with baselines on optical flow estimation and image deblurring on 5 (including real and synthetic) datasets. Our goal is to demonstrate that given a single blurred image and event stream, jointly optimising the image and optical flow would achieve better results than "event only", "single (blurred) image only", and stage-wise methods. We report quantitative comparisons in Table 7.1, 7.3 and qualitative comparisons in Fig. 7.1, 7.2, 7.3, 7.4 to show the effectiveness and generalization of our method. Ablation study in Table 7.2 shows the effectiveness of each term in our objective function (7.3).

As shown in Table 7.1 and Fig. 7.3, we achieve competitive results on flow estimation compared with event only based methods Zhu et al. [2018a, 2019]

Table 7.3: *Quantitative analysis on the GoPro dataset Nah et al. [2017]. This dataset provides ground-truth latent images and the associated motion blurred images. The ground-truth optical flow is estimated by PWC-Net from the sharp video. To demonstrate the efficiency of our optimization method, we use the output of 'EDI + PWC-Net' as the input to our method. Our optimization method can still show improvements.*

Input	EDI images and events		Events	Image and events		
	SelFlow	PWC-Net	EV-FlowNet	EDI + PWC-Net + Our optimization	Our initialization	Our results
AEE	2.0557	1.5806	2.0337	0.9796	3.7868	**0.8641**
MSE	5.7199	4.8951	10.5480	2.5952	8.3929	**2.1536**
FE(%)	0.1722	0.1049	0.2839	0.0895	0.1218	**0.0632**
PSNR	-	-	-	31.5595	29.3789	**31.9234**

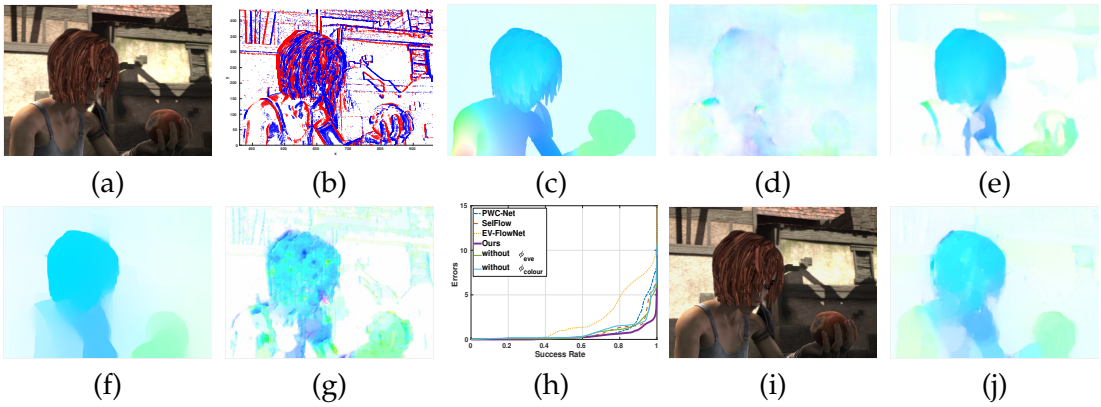


Figure 7.4: *An example of our method on dataset Butler et al. [2012]. (a) Input blurred image. (b) Input events. (c) Ground-truth optical flow. (d) Flow result by Sun et al. [2018] based on images estimated by EDI model Pan et al. [2019c]. (e) Flow result by Liu et al. [2019b] based on images estimated by EDI model. (f) Ours baseline result without term ϕ_{eve}. (g) Ours baseline result without term ϕ_{blur}. (h) Error Map. (i) Our deblurring result. (j) Our optical flow.*

on MVSEC dataset. Note that models in Zhu et al. [2018a, 2019] are trained on MVSEC while our model can still achieve competitive results without training. As BED and ECD do not provide ground-truth flow or sharp image for evaluation, we thus show qualitative comparisons in Fig. 7.1 and 7.2, which demonstrate the stability of our model under both blurred and non-blurred conditions.

We show flow comparisons in Table 7.1 and Fig. 7.4 on the Sintel dataset. While Sintel provides a blurred dataset mainly focusing on out-of-focus blur (including slightly motion blur), our method can achieve competitive results on flow estimation. Also, we gained a 1 dB increase on the PSNR metric for image deblurring. In Table 7.3 and Fig. 7.5, we provide deblurring comparisons on GoPro dataset Nah et al. [2017]. Our approach outperform all the

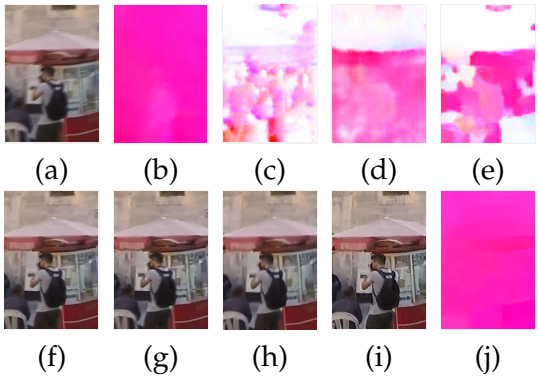


Figure 7.5: *An example of our method on dataset Nah et al. [2017]. (a) The blurred image. (b) The ground-truth flow. (c) Flow result by Zhu et al. [2018a], using the events as input. (d) Flow result by Sun et al. [2018] based on images estimated by the EDI model Pan et al. [2019c]. (e) Flow result by Liu et al. [2019b] based on images estimated by the EDI model. (f) The ground-truth latent images at time t. (g)Deblurred result by Pan et al. [2019c]. (h) Deblurred result by Zhang et al. [2019]. (i) Our deblurred image. (j) Our estimated optical flow.*

baseline methods on flow estimation and image deblurring, which further indicated that 1) including a single image helps achieve better flow estimate than event only based approaches especially in regions with no events, 2) two-stages approaches suffer from image artifacts (even images from EDI) which motivate us to jointly perform image refinement and flow estimate.

Ablation Study. To provide a deep understanding of our model, we evaluate the influence of ϕ_{eve} and ϕ_{blur} in Table 7.2. The significantly decreased performance indicates the contribution of each term in our model. In Table 7.3, we add a comparison to demonstrate the efficiency of our optimization strategy. With a better flow input from 'EDI + PWC-Net', we can still achieve significant improvement. Note, the threshold c is estimated based on Pan et al. [2019c] and our initial flow is simply computed using Eq. (7.2) on event frames.

7.8 Conclusion

In this chapter, we jointly estimate optical flow and the sharp intensity image based on a single image (potentially blurred) and events from DAVIS. Under our formulation, events are high-efficiency data that can reinforce flow estimation. Extensive experiments on different datasets produce competitive results that show the generalization ability, effectiveness and accuracy of our model. While our approach can handle high dynamic cases, we still have difficulties in tackling low texture scenarios, and unstably with noise event

data like other methods. Our future work will explore events representation to build a learning-based end-to-end flow estimation Neural Network with the image.

Chapter 8

Summary and Future Work

This thesis addresses the problem of blur image restoration and high-temporal resolution video reconstruction with an event camera. We first solve the single image restoration problem in the frequency domain. Then, we proposed algorithms with RGBD images and stereo videos to tackle the challenging problem. Furthermore, we explored new sensors (event cameras) to reconstruct a high frame rate, blur-free video by fusing a blurred image with its event sequence.

In Chapter 2, our proposed *phase-only image* based kernel estimation approach is simple (implemented in a few lines of code). The resulting image deblurring algorithm achieves better quantitative results (using PSNR, SSIM, and SSD) than the state-of-the-art methods by extensive evaluation on the benchmark datasets. While our approach can handle the general blur cases, it still suffers from low lighting conditions like other deblurring methods. Our future work will explore how to remove blurs less sensitive to lighting conditions.

Chapter 3 presented a joint optimization framework to estimate the 6 DoF camera motion and deblur the image from a single blurry image. To alleviate the difficulties, we exploit the availability of depth maps (either from noisy measurements or learned through a deep neural network) and a small motion model for the camera. Under our formulation, the solution of one sub-task contribute to the solution of the other sub-tasks. Extensive experiments on both synthetic and real image datasets have demonstrated the superiority of our framework over very recent state-of-the-art blind image deblurring methods. In the future, we plan to exploit more general parametric camera trajectories to improve the performance in challenging real-world scenarios further.

Chapter 4 presented a joint optimization framework to tackle the challenging task of stereo video deblurring where scene flow estimation, Moving object segmentation, and video deblurring is solved in a coupled manner. Under our formulation, the motion cues from scene flow estimation and blur information could reinforce each other, and produce superior results than

conventional scene flow estimation or stereo deblurring methods. We have demonstrated the benefits of our framework on extensive synthetic and real stereo sequences. We plan to extend our approach to deal with multiple frames to achieve better stereo deblurring in the future.

Chapter 5 proposed an **Event-based Double Integral (EDI)** model to naturally connect intensity images and event data captured by the event camera, which also takes the blur generation process into account. In this way, our model can be used to recover latent sharp images and reconstruct intermediate frames at high frame-rate. We also proposed a simple yet effective method to solve our EDI model. Due to the simplicity of our optimization process, our method is efficient as well. Extensive experiments show that our method can generate high-quality, high frame-rate videos efficiently under different conditions, such as low lighting and complex dynamic scenes. In the future, to handle discontinuities at the flicker of reconstructed video, we plan to explore a practical denoising approach for event data.

Chapter 6 proposed a **multiple Event-based Double Integral (mEDI)** model to naturally connect intensity images and events recorded by an event camera (DAVIS), which also takes the blur generation process into account. In this way, our model can be used to recover the latent sharp images and reconstruct intermediate frames at a high frame rate. We also propose a simple yet effective method to solve our **mEDI** model. Due to the simplicity of our optimization process, our approach is efficient as well. Extensive experiments have shown that our method can generate high-quality, high frame-rate videos efficiently under different conditions, such as low lighting and complex dynamic scenes. Our future work will explore how to fuse the image and event data source more efficiently.

Chapter 7 estimated optical flow and the sharp intensity image jointly based on a single image (potentially blurred) and events from DAVIS. Under our formulation, events are high-efficiency data that can reinforce flow estimation. Extensive experiments on different datasets produce competitive results that have shown the generalization ability, effectiveness, and accuracy of our model. While our approach can handle high dynamic cases, we still have difficulties tackling low texture scenarios and unstably with noise event data. Our future work will explore events representation to build a learning-based end-to-end flow estimation Neural Network with the image.

Bibliography

Aftab, K. and Hartley, R., 2015. Convergence of iteratively re-weighted least squares to robust m-estimators. In *Winter Conference on Applications of Computer Vision (WACV)*, 480–487. IEEE. (cited on page 27)

Amir, A.; Taba, B.; Berg, D.; Melano, T.; McKinstry, J.; Di Nolfo, C.; Nayak, T.; Andreopoulos, A.; Garreau, G.; Mendoza, M.; Kusnitz, J.; Debole, M.; Esser, S.; Delbruck, T.; Flickner, M.; and Modha, D., 2017. A low power, fully event-based gesture recognition system. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 127)

Bardow, P.; Davison, A. J.; and Leutenegger, S., 2016. Simultaneous optical flow and intensity estimation from an event camera. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 884–892. doi:10.1109/CVPR.2016.102. (cited on pages 12, 13, 83, 84, 100, 103, 127, 128, and 132)

Barranco, F.; Fermüller, C.; and Aloimonos, Y., 2014. Contour motion estimation for asynchronous event-driven cameras. *Proceedings of the IEEE*, 102, 10 (2014), 1537–1556. (cited on page 127)

Barranco, F.; Fermuller, C.; and Aloimonos, Y., 2015. Bio-inspired motion estimation with event-driven sensors. In *International Work-Conference on Artificial Neural Networks*, 309–321. Springer. (cited on pages 13 and 128)

Barua, S.; Miyatani, Y.; and Veeraraghavan, A., 2016. Direct face detection and video reconstruction from event cameras. In *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 1–9. doi:10.1109/WACV.2016.7477561. (cited on pages 12, 83, 84, and 103)

Belongie, S., 1999. Rodrigues' rotation formula. *From MathWorld–A Wolfram Web Resource, created by Eric W. Weisstein. http://mathworld. wolfram. com/RodriguesRotationFormula. html*, (1999). (cited on page 43)

Benosman, R.; Clercq, C.; Lagorce, X.; Ieng, S.-H.; and Bartolozzi, C., 2013. Event-based visual flow. *IEEE Trans. Neural Networks Learning Syst.*, 25, 2 (2013), 407–417. (cited on pages 13 and 128)

Benosman, R.; Ieng, S.-H.; Clercq, C.; Bartolozzi, C.; and Srinivasan, M., 2012. Asynchronous frameless event-based optical flow. *Neural Networks*, 27 (2012), 32–37. (cited on pages 13 and 128)

Blake, A. and Zisserman, A., 1987. *Visual Reconstruction*. MIT Press, Cambridge, MA, USA. ISBN 0-262-02271-0. (cited on page 27)

Brandli, C.; Berner, R.; Yang, M.; Liu, S.-C.; and Delbruck, T., 2014a. A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49, 10 (2014), 2333–2341. (cited on pages 9, 13, 82, 83, 84, 86, 93, 100, 103, 104, 105, 121, and 126)

Brandli, C.; Muller, L.; and Delbruck, T., 2014b. Real-time, high-speed video decompression using a frame- and event-based DAVIS sensor. In *IEEE Int. Symp. Circuits Syst. (ISCAS)*, 686–689. doi:10.1109/ISCAS.2014.6865228. (cited on pages 13, 83, 84, 101, 104, and 129)

Brandli, C.; Muller, L.; and Delbruck, T., 2014c. Real-time, high-speed video decompression using a frame-and event-based davis sensor. In *Circuits and Systems (ISCAS), 2014 IEEE International Symposium on*, 686–689. IEEE. (cited on pages 12 and 103)

Bryner, S.; Gallego, G.; Rebecq, H.; and Scaramuzza, D., 2019. Event-based, direct camera tracking from a photometric 3d map using nonlinear optimization. In *2019 International Conference on Robotics and Automation (ICRA)*, 325–331. IEEE. (cited on page 132)

Butler, D. J.; Wulff, J.; Stanley, G. B.; and Black, M. J., 2012. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision*, Part IV, LNCS 7577, 611–625. Springer-Verlag. (cited on pages xxvii, xxx, 137, 138, and 140)

Calabrese, E.; Taverni, G.; Awai Easthope, C.; Skriabine, S.; Corradi, F.; Longinotti, L.; Eng, K.; and Delbruck, T., 2019. Dhp19: Dynamic vision sensor 3d human pose dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 0–0. (cited on page 127)

Candès, E. J. and Recht, B., 2009. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9, 6 (2009), 717. (cited on page 4)

Chambolle, A. and Pock, T., 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40, 1 (2011), 120–145. (cited on pages 46, 72, 73, 134, and 135)

Chen, H.; Gu, J.; Gallo, O.; Liu, M.-Y.; Veeraraghavan, A.; and Kautz, J., 2018. Reblur2deblur: Deblurring videos via self-supervised learning. In *IEEE Int. Conf. Comput. Photography (ICCP)*, 1–9. IEEE. (cited on pages 8 and 12)

Cho, S. and Lee, S., 2009. Fast motion deblurring. In *ACM Transactions on Graphics (TOG)*, vol. 28, 145. ACM. (cited on pages 6, 20, and 30)

Cho, S.; Wang, J.; and Lee, S., 2011. Handling outliers in non-blind image deconvolution. In *Proceedings of the IEEE International Conference on Computer Vision*, 495–502. IEEE. (cited on page 30)

Cho, S.; Wang, J.; and Lee, S., 2012. Video deblurring for hand-held cameras using patch-based synthesis. *ACM Transactions on Graphics (TOG)*, 31, 4 (2012), 64. (cited on pages 11, 20, 41, and 85)

Cook, M.; Gugelmann, L.; Jug, F.; Krautz, C.; and Steger, A., 2011. Interacting maps for fast visual interpretation. In *Int. Joint Conf. Neural Netw. (IJCNN)*, 770–776. doi:10.1109/IJCNN.2011.6033299. (cited on pages 12 and 103)

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B., 2016. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3213–3223. (cited on page 67)

Dai, S. and Wu, Y., 2008. Motion from blur. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–8. IEEE. (cited on pages 4, 63, and 80)

Delbruck, T.; Hu, Y.; and He, Z., 2020. V2e: From video frames to realistic dvs event camera streams. *arXiv preprint arXiv:2006.07722*, (2020). (cited on page 107)

Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; Van Der Smagt, P.; Cremers, D.; and Brox, T., 2015. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2758–2766. (cited on pages 14, 62, and 128)

Dunlap, R. A., 1997. *The golden ratio and Fibonacci numbers*. World Scientific. (cited on page 92)

Eboli, T.; Sun, J.; and Ponce, J., 2020. End-to-end interpretable learning of non-blind image deblurring. *arXiv preprint arXiv:2007.01769*, (2020). (cited on page 7)

Eilertsen, G.; Kronander, J.; Denes, G.; Mantiuk, R. K.; and Unger, J., 2017. Hdr image reconstruction from a single exposure using deep cnns. *ACM transactions on graphics (TOG)*, 36, 6 (2017), 1–15. (cited on pages xxvi and 122)

Endo, Y.; Kanamori, Y.; and Kuriyama, S., 2019. Animating landscape: Self-supervised learning of decoupled motion and appearance for single-image video synthesis. *arXiv preprint arXiv:1910.07192*, (2019). (cited on pages 14 and 129)

Faktor, A. and Irani, M., 2014. Video segmentation by non-local consensus voting. In *Proc. Brit. Mach. Vis. Conf.*, vol. 2, 8. (cited on pages xxi, 61, 65, 75, 76, and 77)

Fan, Q.; Zhong, F.; Lischinski, D.; Cohen-Or, D.; and Chen, B., 2015. Jumpcut: non-successive mask transfer and interpolation for video cutout. *ACM Trans. Graph.*, 34, 6 (2015), 195–1. (cited on page 61)

Fergus, R.; Singh, B.; Hertzmann, A.; Roweis, S. T.; and Freeman, W. T., 2006. Removing camera shake from a single photograph. In *ACM Transactions on Graphics (TOG)*, vol. 25, 787–794. ACM. (cited on pages 5, 41, 58, 85, and 104)

Franke, U. and Joos, A., 2000. Real-time stereo vision for urban traffic scene understanding. In *IEEE Intelligent Vehicles Symposium*. (cited on page 54)

Gallego, G.; Delbruck, T.; Orchard, G.; Bartolozzi, C.; Taba, B.; Censi, A.; Leutenegger, S.; Davison, A.; Conradt, J.; Daniilidis, K.; et al., 2019. Event-based vision: A survey. *arXiv preprint arXiv:1904.08405*, (2019). (cited on pages 12, 103, and 127)

Gallego, G.; Lund, J. E.; Mueggler, E.; Rebecq, H.; Delbruck, T.; and Scaramuzza, D., 2017. Event-based, 6-dof camera tracking from photometric depth maps. *IEEE transactions on pattern analysis and machine intelligence*, 40, 10 (2017), 2402–2412. (cited on page 107)

Gallego, G.; Rebecq, H.; and Scaramuzza, D., 2018. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 14, 127, and 128)

Gehrig, D.; Loquercio, A.; Derpanis, K. G.; and Scaramuzza, D., 2019a. End-to-end learning of representations for asynchronous event-based data. In *Proceedings of the IEEE International Conference on Computer Vision*. (cited on page 103)

Gehrig, D.; Loquercio, A.; Derpanis, K. G.; and Scaramuzza, D., 2019b. End-to-end learning of representations for asynchronous event-based data. In *Proceedings of the IEEE International Conference on Computer Vision*, 5633–5643. (cited on pages 127 and 132)

Gehrig, D.; Rebecq, H.; Gallego, G.; and Scaramuzza, D., 2018. Asynchronous, photometric feature tracking using events and frames. In *European Conference on Computer Vision*. (cited on pages 12, 84, 103, and 126)

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R., 2013. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, (2013), 0278364913491297. (cited on pages xx, 46, 57, 73, 74, and 75)

Geiger, A.; Lenz, P.; and Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3354–3361. (cited on page 54)

Geiger, A.; Ziegler, J.; and Stiller, C., 2011. Stereoscan: Dense 3d reconstruction in real-time. In *IEEE Intelligent Vehicles Symposium (IV)*, 963–968. (cited on pages 67, 73, and 74)

Godard, C.; Mac Aodha, O.; and Brostow, G., 2017. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6602–6611. (cited on pages xix, 38, 48, and 50)

Gong, D.; Yang, J.; Liu, L.; Zhang, Y.; Reid, I.; Shen, C.; Hengel, A. v. d.; and Shi, Q., 2017a. From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 14, 58, 60, 62, and 64)

Gong, D.; Yang, J.; Liu, L.; Zhang, Y.; Reid, I.; Shen, C.; van den Hengel, A.; and Shi, Q., 2017b. From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2319–2328. (cited on pages xviii, xix, xxvi, xxix, 1, 6, 7, 11, 19, 21, 28, 29, 30, 35, 38, 42, 48, 49, 85, 93, 95, 105, 117, 126, 129, 130, and 138)

Gupta, A.; Joshi, N.; Zitnick, C. L.; Cohen, M.; and Curless, B., 2010. Single image deblurring using motion density functions. In *European Conference on Computer Vision*, 171–184. Springer. (cited on pages 5, 6, 10, 11, 19, 20, 28, 38, 41, 54, 58, 63, and 80)

Hariharan, B.; Arbeláez, P.; Girshick, R.; and Malik, J., 2015. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 447–456. (cited on page 61)

He, K.; Zhang, X.; Ren, S.; and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778. (cited on page 62)

Hirsch, M.; Schuler, C. J.; Harmeling, S.; and Schölkopf, B., 2011. Fast removal of non-uniform camera shake. In *Proceedings of the IEEE International Conference on Computer Vision*, 463–470. (cited on pages 4 and 41)

Hirschmuller, H., 2008. Stereo processing by semiglobal matching and mutual information. *IEEE transactions on pattern analysis and machine intelligence*, 30, 2 (2008), 328–341. (cited on page 73)

Horn, B. K. and Schunck, B. G., 1981. Determining optical flow. *J. Artificial Intell.*, 17, 1-3 (1981), 185–203. (cited on pages 127 and 129)

Hu, Z.; Xu, L.; and Yang, M.-H., 2014. Joint depth estimation and camera shake removal from single blurry image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2893–2900. (cited on pages xix, 1, 6, 10, 11, 19, 20, 28, 38, 41, 48, 49, 54, and 58)

Ilg, E.; Mayer, N.; Saikia, T.; Keuper, M.; Dosovitskiy, A.; and Brox, T., 2017. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 14, 62, and 128)

Im, S.; Ha, H.; Choe, G.; Jeon, H.-G.; Joo, K.; and So Kweon, I., 2015. High quality structure from small motion for rolling shutter cameras. In *Proceedings of the IEEE International Conference on Computer Vision*. (cited on page 39)

Jang, W.-D. and Kim, C.-S., 2017. Online video object segmentation via convolutional trident network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5849–5858. (cited on page 61)

Jason, J. Y.; Harley, A. W.; and Derpanis, K. G., 2016. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *European Conference on Computer Vision Workshops*, 3–10. Springer. (cited on pages 127 and 128)

Ji, H. and Liu, C., 2008. Motion blur identification from image gradients. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 1–8. IEEE. (cited on page 20)

Jia, J., 2014. Mathematical models and practical solvers for uniform motion deblurring. *Motion Deblurring: Algorithms and Systems*, (2014), 1. (cited on page 54)

Jin, M.; Meishvili, G.; and Favaro, P., 2018. Learning to extract a video sequence from a single motion-blurred image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xxii, xxiii, xxiv, xxv, xxvi, xxix, 9, 39, 42, 61, 82, 83, 85, 91, 93, 94, 95, 97, 101, 105, 109, 117, 118, 120, and 121)

Joshi, N.; Szeliski, R.; and Kriegman, D. J., 2008. Psf estimation using sharp edge prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–8. (cited on page 20)

Kiefer, J., 1953. Sequential minimax search for a maximum. *Proceedings of the American mathematical society*, 4, 3 (1953), 502–506. (cited on page 118)

Kim, H.; Handa, A.; Benosman, R.; Ieng, S.-H.; and Davison, A. J., 2014. Simultaneous mosaicing and tracking with an event camera. In *Proc. Brit. Mach. Vis. Conf.* doi:10.5244/C.28.26. (cited on pages 12, 84, and 103)

Kim, H.; Leutenegger, S.; and Davison, A. J., 2016. Real-time 3D reconstruction and 6-DoF tracking with an event camera. In *European Conference on Computer Vision*, 349–364. doi:10.1007/978-3-319-46466-4_21. (cited on pages 12, 84, 103, and 126)

Kim, T. H.; Lee, H. S.; and Lee, K. M., 2013. Optical flow via locally adaptive fusion of complementary data costs. In *Proceedings of the IEEE International Conference on Computer Vision*, 3344–3351. (cited on page 133)

Kim, T. H. and Lee, K. M., 2014. Segmentation-free dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2766–2773. (cited on pages 11, 21, 41, 60, 63, 64, 66, 75, 80, and 129)

Kim, T. H. and Lee, K. M., 2015. Generalized video deblurring for dynamic scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5426–5434. (cited on pages xix, xx, xxi, xxii, 7, 11, 18, 21, 38, 41, 48, 49, 54, 55, 57, 58, 59, 60, 65, 74, 77, 78, 85, 105, and 129)

Kim, T. H.; Lee, K. M.; Scholkopf, B.; and Hirsch, M., 2017. Online video deblurring via dynamic temporal blending network. In *Proceedings of the IEEE International Conference on Computer Vision*. (cited on pages 41, 61, and 64)

Kim, T. H.; Nah, S.; and Lee, K. M., 2018. Dynamic video deblurring using a locally adaptive blur model. *IEEE transactions on pattern analysis and machine intelligence*, 40, 10 (Oct 2018), 2374–2387. doi:10.1109/TPAMI.2017.2761348. (cited on pages 61 and 76)

Köhler, R.; Hirsch, M.; Mohler, B.; Schölkopf, B.; and Harmeling, S., 2012. Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In *European Conference on Computer Vision*, 27–40. (cited on pages xvii, xviii, xix, xxix, 5, 30, 31, 32, 33, 34, 35, and 38)

Kolmogorov, V., 2006. Convergent tree-reweighted message passing for energy minimization. *IEEE transactions on pattern analysis and machine intelligence*, 28, 10 (2006), 1568–1583. (cited on page 72)

Kovesi, P., 2003. Phase congruency detects corners and edges. In *DICTA*. (cited on page 22)

Krishnan, D. and Fergus, R., 2009. Fast image deconvolution using hyper-laplacian priors. In *Proc. Adv. Neural Inf. Process. Syst.*, 1033–1041. (cited on page 71)

Krishnan, D.; Tay, T.; and Fergus, R., 2011. Blind deconvolution using a normalized sparsity measure. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 233–240. (cited on pages 20, 41, 48, 54, 58, 71, 85, and 104)

Kueng, B.; Mueggler, E.; Gallego, G.; and Scaramuzza, D., 2016a. Low-latency visual odometry using event-based feature tracks. In *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 16–23. Daejeon, Korea. doi:10.1109/IROS.2016.7758089. (cited on pages 12, 84, and 103)

Kueng, B.; Mueggler, E.; Gallego, G.; and Scaramuzza, D., 2016b. Low-latency visual odometry using event-based feature tracks. In *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 16–23. IEEE. (cited on page 126)

Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; and Matas, J., 2018a. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8183–8192. (cited on pages xxix, 21, and 35)

Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; and Matas, J., 2018b. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 61, 74, and 75)

Lai, W.-S.; Ding, J.-J.; Lin, Y.-Y.; and Chuang, Y.-Y., 2015. Blur kernel estimation using normalized color-line prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 64–72. (cited on pages 41, 58, 85, and 104)

Lai, W. S.; Huang, J. B.; Hu, Z.; Ahuja, N.; and Yang, M. H., 2016. A comparative study for single image blind deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1701–1709. doi: 10.1109/CVPR.2016.188. (cited on page 38)

Lee, D.; Park, H.; Kyu Park, I.; and Lee, K. M., 2018. Joint blind motion deblurring and depth estimation of light field. In *European Conference on Computer Vision*. (cited on page 41)

Levin, A., 2007. Blind motion deblurring using image statistics. In *Advances in Neural Information Processing Systems*, 841–848. (cited on page 6)

Levin, A.; Weiss, Y.; Durand, F.; and Freeman, W. T., 2009. Understanding and evaluating blind deconvolution algorithms. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1964–1971. (cited on pages xvii, xviii, xxix, 3, 10, 19, 29, 30, and 35)

Levin, A.; Weiss, Y.; Durand, F.; and Freeman, W. T., 2011. Understanding blind deconvolution algorithms. *IEEE transactions on pattern analysis and machine intelligence*, 33, 12 (2011), 2354–2367. (cited on page 4)

Li, B.; Dai, Y.; and He, M., 2018a. Monocular depth estimation with hierarchical fusion of dilated cnns and soft-weighted-sum inference. *Pattern Recognition*, (2018). (cited on page 39)

Li, L.; Pan, J.; Lai, W.-S.; Gao, C.; Sang, N.; and Yang, M.-H., 2018b. Learning a discriminative prior for blind image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6616–6625. (cited on pages 10, 19, 21, and 42)

Li, L.; Pan, J.; Lai, W.-S.; Gao, C.; Sang, N.; and Yang, M.-H., 2018c. Learning a discriminative prior for blind image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 58)

Lichtsteiner, P.; Posch, C.; and Delbruck, T., 2008. A 128×128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE journal of solid-state circuits*, 43, 2 (2008), 566–576. (cited on pages 9, 82, 84, 86, 100, 103, 105, 107, and 126)

Lin, G.; Milan, A.; Shen, C.; and Reid, I., 2017. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 62)

Liu, H.-C.; Zhang, F.-L.; Marshall, D.; Shi, L.; and Hu, S.-M., 2017a. High-speed video generation with an event camera. *The Visual Computer*, 33, 6-8 (2017), 749–759. (cited on pages 13, 84, and 104)

Liu, L.; Campbell, D.; Li, H.; Zhou, D.; Song, X.; and Yang, R., 2020. Learning 2d-3d correspondences to solve the blind perspective-n-point problem. *arXiv preprint arXiv:2003.06752*, (2020). (cited on page 1)

Liu, L. and Li, H., 2019. Lending orientation to neural networks for cross-view geo-localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5624–5633. (cited on page 12)

Liu, L.; Li, H.; and Dai, Y., 2017b. Efficient global 2d-3d matching for camera localization in a large-scale 3d map. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. (cited on pages 12 and 103)

Liu, L.; Li, H.; and Dai, Y., 2019a. Stochastic attraction-repulsion embedding for large scale image localization. In *Proceedings of the IEEE International Conference on Computer Vision*, 2570–2579. (cited on pages 12 and 103)

Liu, L.; Li, H.; Dai, Y.; and Pan, Q., 2017c. Robust and efficient relative pose with a multi-camera system for autonomous driving in highly dynamic environments. *IEEE Transactions on Intelligent Transportation Systems*, 19, 8 (2017), 2432–2444. (cited on page 54)

Liu, L.; Li, H.; Yao, H.; and Zha, R., 2021. Pluckernet: Learn to register 3d line reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1842–1852. (cited on page 1)

Liu, M. and Delbruck, T., 2018. Adaptive time-slice block-matching optical flow algorithm for dynamic vision sensors. In *Proc. Brit. Mach. Vis. Conf.* (cited on page 127)

Liu, P.; Lyu, M.; King, I.; and Xu, J., 2019b. Selflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xxvii, 14, 128, 138, 139, 140, and 141)

Liu, Z.; Li, X.; Luo, P.; Loy, C. C.; and Tang, X., 2018. Deep learning markov random field for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 40, 8 (2018), 1814–1828. (cited on page 62)

Liwicki, S.; Zach, C.; Miksik, O.; and Torr, P. H., 2016. Coarse-to-fine planar regularization for dense monocular depth estimation. In *European Conference on Computer Vision*, 458–474. Springer. (cited on page 39)

Menze, M. and Geiger, A., 2015. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3061–3070. (cited on pages xx, xxi, 14, 59, 62, 63, 65, 66, 69, 70, 72, and 76)

Michaeli, T. and Irani, M., 2014. Blind deblurring using internal patch recurrence. In *European Conference on Computer Vision*, 783–798. Springer. (cited on page 6)

Moré, J. J., 1978. The levenberg-marquardt algorithm: implementation and theory. In *Numerical analysis*, 105–116. Springer. (cited on pages 45 and 92)

Mueggler, E.; Rebecq, H.; Gallego, G.; Delbruck, T.; and Scaramuzza, D., 2017. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *The International Journal of Robotics Research*, 36, 2 (2017), 142–149. (cited on pages xxii, xxv, xxvi, 89, 90, 93, 115, 116, 119, 131, and 137)

Nah, S.; Kim, T. H.; and Lee, K. M., 2017. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xvii, xviii, xix, xxiii, xxv, xxvi, xxvii, xxix, xxx, 7, 11, 18, 19, 21, 26, 28, 29, 31, 32, 33, 34, 35, 42, 48, 49, 61, 64, 80, 85, 92, 93, 94, 95, 97, 105, 117, 118, 119, 120, 121, 137, 140, and 141)

Nah, S.; Son, S.; and Lee, K. M., 2019a. Recurrent neural networks with intra-frame iterations for video deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 12)

Nah, S.; Son, S.; and Lee, K. M., 2019b. Recurrent neural networks with intra-frame iterations for video deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 129)

Nayar, S. and Ben-Ezra, M., 2004. Motion-based motion deblurring. *IEEE transactions on pattern analysis and machine intelligence*, 26, 6 (2004), 689–698. (cited on page 60)

Oppenheim, A. and Lim, J., 1981. The importance of phase in signals. *Proceedings of the IEEE*, 69 (1981), 529–541. (cited on page 19)

Pan, J.; Hu, Z.; Su, Z.; Lee, H.-Y.; and Yang, M.-H., 2016a. Soft-segmentation guided object motion deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 459–468. (cited on pages 1, 10, 11, 19, 20, 38, 41, 56, 58, 60, and 67)

Pan, J.; Hu, Z.; Su, Z.; and Yang, M.-H., 2014. Deblurring text images via l0-regularized intensity and gradient prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2901–2908. (cited on pages 6, 10, 19, and 58)

Pan, J.; Sun, D.; Pfister, H.; and Yang, M.-H., 2016b. Blind image deblurring using dark channel prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1628–1636. (cited on pages xviii, 4, 10, 18, 19, 26, 30, 35, 41, and 58)

Pan, J.; Sun, D.; Pfister, H.; and Yang, M.-H., 2017a. Deblurring images via dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, (2017). (cited on pages xix, xx, xxii, xxiii, xxvi, xxix, 31, 32, 33, 34, 41, 46, 47, 48, 50, 51, 82, 85, 93, 95, 97, 104, 120, and 121)

Pan, L.; Chowdhury, S.; Hartley, R.; Liu, M.; Zhang, H.; and Li, H., 2021. Dual pixel exploration: Simultaneous depth estimation and image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4340–4349. (cited on page 1)

Pan, L.; Dai, Y.; and Liu, M., 2019a. Single image deblurring and camera motion estimation with depth map. In *Winter Conference on Applications of Computer Vision (WACV)*, 2116–2125. IEEE. (cited on pages 6, 20, and 129)

Pan, L.; Dai, Y.; Liu, M.; and Porikli, F., 2017b. Simultaneous stereo video deblurring and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xx, xxii, xxix, 1, 7, 11, 14, 19, 21, 38, 40, 41, 54, 55, 57, 58, 60, 62, 74, 75, 78, 85, 105, 117, and 129)

Pan, L.; Dai, Y.; Liu, M.; and Porikli, F., 2018. Depth map completion by jointly exploiting blurry color images and sparse depth maps. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1377–1386. (cited on pages 20, 85, and 105)

Pan, L.; Dai, Y.; Liu, M.; Porikli, F.; and Pan, Q., 2020. Joint stereo video deblurring, scene flow estimation and moving object segmentation. *IEEE Transactions on Image Processing*, 29 (2020), 1748–1761. doi:10.1109/TIP.2019.2945867. (cited on pages 1, 105, and 129)

Pan, L.; Hartley, R.; Liu, M.; and Dai, Y., 2019b. Phase-only image based kernel estimation for single image blind deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6034–6043. (cited on pages xvii, 3, and 104)

Pan, L.; Hartley, R.; Scheerlinck, C.; Liu, M.; Yu, X.; and Dai, Y., 2020a. High frame rate video reconstruction based on an event camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2020). (cited on pages 9 and 129)

Pan, L.; Liu, M.; and Hartley, R., 2020b. Single image optical flow estimation with an event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1672–1681. (cited on pages 12 and 103)

Pan, L.; Scheerlinck, C.; Yu, X.; Hartley, R.; Liu, M.; and Dai, Y., 2019c. Bringing a blurry frame alive at high frame-rate with an event camera.

In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (cited on pages xxiv, xxv, xxvi, xxvii, 9, 101, 102, 110, 120, 122, 123, 126, 127, 129, 130, 131, 137, 138, 139, 140, and 141)

Papari, G. and Petkov, N., 2011. Edge and line oriented contour detection: State of the art. *Image and Vision Computing*, 29, 2-3 (2011), 79–103. (cited on page 19)

Papazoglou, A. and Ferrari, V., 2013. Fast object segmentation in unconstrained video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1777–1784. (cited on pages xxi, 61, 65, 75, 76, and 77)

Park, H. and Lee, K. M., 2017a. Joint estimation of camera pose, depth, deblurring, and super-resolution from a blurred image sequence. In *Proceedings of the IEEE International Conference on Computer Vision*. (cited on page 40)

Park, H. and Lee, K. M., 2017b. Joint estimation of camera pose, depth, deblurring, and super-resolution from a blurred image sequence. In *Proceedings of the IEEE International Conference on Computer Vision*. (cited on page 60)

Perazzi, F.; Pont-Tuset, J.; McWilliams, B.; Van Gool, L.; Gross, M.; and Sorkine-Hornung, A., 2016. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 75)

Perrone, D. and Favaro, P., 2014. Total variation blind deconvolution: The devil is in the details. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2909–2916. (cited on page 58)

Pichaikuppan, V. R. A.; Narayanan, R. A.; and Rangarajan, A., 2014. Change detection in the presence of motion blur and rolling shutter effect. In *European Conference on Computer Vision*, 123–137. Springer. (cited on pages xvii and 1)

Pock, T.; Chambolle, A.; Cremers, D.; and Bischof, H., 2009a. A convex relaxation approach for computing minimal partitions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 810–817. IEEE. (cited on page 134)

Pock, T.; Cremers, D.; Bischof, H.; and Chambolle, A., 2009b. An algorithm for minimizing the mumford-shah functional. In *Proceedings of the IEEE International Conference on Computer Vision*, 1133–1140. IEEE. (cited on page 134)

Posch, C.; Matolin, D.; and Wohlgenannt, R., 2010. A QVGA 143dB dynamic range asynchronous address-event PWM dynamic image sensor with lossless pixel-level video compression. In *IEEE Intl. Solid-State Circuits Conf. (ISSCC)*, 400–401. doi:10.1109/ISSCC.2010.5433973. (cited on pages 12 and 103)

Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; and Flannery, B. P., 1988. Numerical recipes in c. *Cambridge University Press*, 1 (1988), 3. (cited on page 118)

Purohit, K.; Shah, A.; and Rajagopalan, A., 2019. Bringing alive blurred moments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6830–6839. (cited on pages 39, 85, and 105)

Ranftl, R.; Vineet, V.; Chen, Q.; and Koltun, V., 2016. Dense monocular depth estimation in complex dynamic scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4058–4066. (cited on page 39)

Rebecq, H.; Gehrig, D.; and Scaramuzza, D., 2018. Esim: an open event camera simulator. In *Conference on Robot Learning*, 969–982. (cited on pages 121 and 137)

Rebecq, H.; Horstschäfer, T.; Gallego, G.; and Scaramuzza, D., 2016. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time. *IEEE Robotics and Automation Letters*, 2, 2 (2016), 593–600. (cited on page 126)

Rebecq, H.; Horstschäfer, T.; Gallego, G.; and Scaramuzza, D., 2017. EVO: A geometric approach to event-based 6-DOF parallel tracking and mapping in real-time. *IEEE Robot. Autom. Lett.*, 2 (2017). doi:10.1109/LRA.2016.2645143. (cited on pages 84 and 103)

Rebecq, H.; Ranftl, R.; Koltun, V.; and Scaramuzza, D., 2019. Events-to-video: Bringing modern computer vision to event cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xxiv, xxv, xxvi, 13, 100, 101, 104, 108, 110, 116, 121, 122, 123, 127, and 129)

Rebecq, H.; Ranftl, R.; Koltun, V.; and Scaramuzza, D., 2020. High speed and high dynamic range video with an event camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2020). (cited on pages 13 and 104)

Reinbacher, C.; Graber, G.; and Pock, T., 2016. Real-time intensity-image reconstruction for event cameras using manifold regularisation. In *Proc. Brit. Mach. Vis. Conf.* (cited on pages xxiii, xxv, xxvi, 12, 13, 83, 84, 93, 94, 95, 103, 104, 118, 120, 121, and 123)

Ren, W.; Pan, J.; Cao, X.; and Yang, M.-H., 2017. Video deblurring via semantic segmentation and pixel-wise non-linear kernel. In *Proceedings of the IEEE International Conference on Computer Vision*. (cited on pages 58 and 60)

Rosello, P., 2016. Predicting future optical flow from static video frames. *Retrieved on: Jul*, 18 (2016). (cited on pages 14 and 129)

Rudin, L. I.; Osher, S.; and Fatemi, E., 1992. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60, 1-4 (1992), 259–268. (cited on pages 4, 92, and 111)

Scharstein, D.; Hirschmüller, H.; Kitajima, Y.; Krathwohl, G.; Nešić, N.; Wang, X.; and Westling, P., 2014. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Conference on Pattern Recognition*, 31–42. Springer. (cited on pages xvii, 8, and 46)

Scheerlinck, C.; Barnes, N.; and Mahony, R., 2018. Continuous-time intensity estimation using event cameras. In *Proc. Asian Conf. Comp. Vis.*, 308–324. Springer. (cited on pages xxii, xxiii, xxiv, xxv, xxvi, xxix, 13, 82, 83, 84, 91, 93, 94, 95, 101, 104, 109, 117, 118, 120, 121, 122, 123, 129, and 137)

Scheerlinck, C.; Barnes, N.; and Mahony, R., 2019a. Asynchronous spatial image convolutions for event cameras. *IEEE Robot. Autom. Lett.*, 4, 2 (April 2019), 816–822. doi:10.1109/LRA.2019.2893427. (cited on pages 84 and 100)

Scheerlinck, C.; Rebecq, H.; Gehrig, D.; Barnes, N.; Mahony, R.; and Scaramuzza, D., 2020. Fast image reconstruction with an event camera. In *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 156–163. (cited on pages 13 and 104)

Scheerlinck, C.; Rebecq, H.; Stoffregen, T.; Barnes, N.; Mahony, R.; and Scaramuzza, D., 2019b. Ced: Color event camera dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 0–0. (cited on pages xxvi, 121, and 122)

Schuler, C. J.; Hirsch, M.; Harmeling, S.; and Schölkopf, B., 2012. Blind correction of optical aberrations. In *European Conference on Computer Vision*, 187–200. Springer. (cited on page 54)

Sellent, A.; Rother, C.; and Roth, S., 2016. Stereo video deblurring. In *European Conference on Computer Vision*, 558–575. Springer. (cited on pages xx, xxi, xxii, 11, 18, 21, 38, 41, 54, 55, 57, 58, 59, 60, 73, 74, 75, 77, 78, 79, 85, 105, and 129)

Seok Lee, H. and Mu Lee, K., 2013. Dense 3d reconstruction from severely blurred images using a single moving camera. In *Proceedings of the IEEE*

Conference on Computer Vision and Pattern Recognition, 273–280. (cited on pages 54 and 64)

Sevilla-Lara, L.; Sun, D.; Jampani, V.; and Black, M. J., 2016. Optical flow with semantic segmentation and localized layers. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 63)

Shankar Nagaraja, N.; Schmidt, F. R.; and Brox, T., 2015. Video segmentation with just a few strokes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3235–3243. (cited on page 61)

Shedligeri, P. and Mitra, K., 2019. Photorealistic image reconstruction from hybrid intensity and event-based sensor. *Journal of Electronic Imaging*, 28, 6 (2019), 063012. (cited on pages 13, 84, and 104)

Shi, J.; Xu, L.; and Jia, J., 2014. Discriminative blur detection features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2965–2972. (cited on pages xvii, xviii, 6, and 24)

Shi, J.; Xu, L.; and Jia, J., 2015. Just noticeable defocus blur detection and estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 657–665. (cited on page 54)

Stoffregen, T.; Gallego, G.; Drummond, T.; Kleeman, L.; and Scaramuzza, D., 2019. Event-based motion segmentation by motion compensation. In *Proceedings of the IEEE International Conference on Computer Vision*, 7244–7253. (cited on pages 12, 103, and 127)

Stoffregen, T.; Scheerlinck, C.; Scaramuzza, D.; Drummond, T.; Barnes, N.; Kleeman, L.; and Mahony, R., 2020. Reducing the Sim-to-Real gap for event cameras. In *European Conference on Computer Vision (ECCV)*. (cited on page 103)

Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; and Cremers, D., 2012. A benchmark for the evaluation of rgb-d slam systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 573–580. (cited on pages 30, 47, and 49)

Su, S.; Delbracio, M.; Wang, J.; Sapiro, G.; Heidrich, W.; and Wang, O., 2017. Deep video deblurring for hand-held cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 21, 38, 41, 61, and 64)

Sun, D.; Yang, X.; Liu, M.-Y.; and Kautz, J., 2018. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xxvii, 14, 62, 128, 138, 139, 140, and 141)

Sun, J.; Cao, W.; Xu, Z.; and Ponce, J., 2015. Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 769–777. (cited on pages xvii, xxix, 5, 6, 10, 11, 19, 20, 21, 39, 42, 54, 58, 85, 93, 95, 105, 117, and 121)

Sun, L.; Cho, S.; Wang, J.; and Hays, J., 2013. Edge-based blur kernel estimation using patch priors. In *IEEE International Conference on Computational Photography*, 1–8. IEEE. (cited on pages 20, 58, 85, and 104)

Sundaram, N.; Brox, T.; and Keutzer, K., 2010. Dense point trajectories by gpu-accelerated large displacement optical flow. In *European Conference on Computer Vision*, 438–451. Springer. (cited on page 61)

Tai, Y.-W.; Chen, X.; Kim, S.; Kim, S. J.; Li, F.; Yang, J.; Yu, J.; Matsushita, Y.; and Brown, M. S., 2013. Nonlinear camera response functions and image deblurring: Theoretical analysis and practice. *IEEE transactions on pattern analysis and machine intelligence*, 35, 10 (2013), 2498–2512. (cited on pages 64 and 80)

Tai, Y.-W.; Du, H.; Brown, M. S.; and Lin, S., 2010. Correction of spatially varying image and video motion blur using a hybrid camera. *IEEE transactions on pattern analysis and machine intelligence*, 32, 6 (2010), 1012–1028. (cited on page 67)

Taniai, T.; Sinha, S. N.; and Sato, Y., 2017. Fast multi-frame stereo scene flow with motion segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 14 and 62)

Tao, X.; Gao, H.; Shen, X.; Wang, J.; and Jia, J., 2018. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xviii, xix, xxii, xxiii, xxiv, xxv, xxvi, xxix, 7, 11, 18, 19, 21, 31, 32, 33, 34, 42, 61, 74, 75, 78, 82, 85, 89, 93, 94, 95, 97, 101, 105, 116, 117, 120, 121, and 138)

Tsai, Y.-H.; Yang, M.-H.; and Black, M. J., 2016a. Video segmentation via object flow. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3899–3908. (cited on page 61)

Tsai, Y.-H.; Zhong, G.; and Yang, M.-H., 2016b. Semantic co-segmentation in videos. In *European Conference on Computer Vision*, 760–775. Springer. (cited on page 62)

Ugray, Z.; Lasdon, L.; Plummer, J.; Glover, F.; Kelly, J.; and Martí, R., 2007. Scatter search and local nlp solvers: A multistart framework for global optimization. *INFORMS Journal on Computing*, 19, 3 (2007), 328–340. (cited on page 92)

Vasu, S. and Rajagopalan, A., 2017. From local to global: Edge profiles to camera motion in blurred images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4447–4456. (cited on page 41)

Vasu, S.; Reddy Maligireddy, V.; and Rajagopalan, A. N., 2018. Non-blind deblurring: Handling kernel uncertainty with cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 7)

Veksler, O., 2001. Stereo matching by compact windows via minimum ratio cycle. In *Proceedings of the IEEE International Conference on Computer Vision*, vol. 1, 540–547. IEEE. (cited on page 27)

Vidal, A. R.; Rebecq, H.; Horstschaefer, T.; and Scaramuzza, D., 2018. Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios. *IEEE Robot. Autom. Lett.*, 3, 2 (2018), 994–1001. (cited on pages 103 and 126)

Vogel, C.; Schindler, K.; and Roth, S., 2015. 3d scene flow estimation with a piecewise rigid scene model. *International Journal of Computer Vision*, 115, 1 (2015), 1–28. (cited on pages xx, 55, 57, and 74)

Walker, J.; Gupta, A.; and Hebert, M., 2015. Dense optical flow prediction from a static image. In *Proceedings of the IEEE International Conference on Computer Vision*, 2443–2451. (cited on pages 14 and 129)

Wang, L.; , S. M. M. I.; Ho, Y.-S.; and Yoon, K.-J., 2019. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 13, 100, 104, and 129)

Wang, T.; Han, B.; and Collomosse, J., 2014. Touchcut: Fast image and video segmentation using single-touch interaction. *Computer Vision and Image Understanding*, 120 (2014), 14–30. (cited on page 61)

Wang, W.; Shen, J.; and Porikli, F., 2015. Saliency-aware geodesic video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3395–3402. (cited on page 61)

Wang, W.; Shen, J.; Yang, R.; and Porikli, F., 2018. Saliency-aware video object segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 40, 1 (Jan 2018), 20–33. doi:10.1109/TPAMI.2017.2662005. (cited on page 61)

Whyte, O.; Sivic, J.; Zisserman, A.; and Ponce, J., 2012. Non-uniform deblurring for shaken images. *International Journal of Computer Vision*, 98, 2 (2012), 168–186. (cited on pages 5, 11, 28, 35, 41, 63, and 80)

Wiener, N., 1950. *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications*. MIT press. (cited on page 2)

Wu, Z.; Shen, C.; and Van Den Hengel, A., 2019. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90 (2019), 119–133. (cited on pages 54, 56, 62, 67, and 74)

Wulff, J. and Black, M. J., 2014. Modeling blurred video with layers. In *European Conference on Computer Vision*, 236–252. Springer. (cited on pages 11, 21, 41, 60, and 67)

Xu, L. and Jia, J., 2010. Two-phase kernel estimation for robust motion deblurring. In *European Conference on Computer Vision*, 157–170. Springer. (cited on page 20)

Xu, L. and Jia, J., 2012. Depth-aware motion deblurring. In *IEEE International Conference on Computational Photography*, 1–8. (cited on pages 10, 19, 20, 38, 41, and 60)

Xu, L.; Lu, C.; Xu, Y.; and Jia, J., 2011. Image smoothing via l0 gradient minimization. *ACM Trans. Graph.*, 30 (2011), 174:1–174:12. (cited on page 27)

Xu, L.; Tao, X.; and Jia, J., 2014. Inverse kernels for fast spatial deconvolution. In *European Conference on Computer Vision*, 33–48. Springer. (cited on pages 10 and 19)

Xu, L.; Zheng, S.; and Jia, J., 2013. Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1107–1114. (cited on pages 4, 6, 10, 19, 20, 27, 35, 38, 41, 54, 58, 85, and 104)

Xu, Y.; Hu, X.; and Peng, S., 2015. Blind motion deblurring using optical flow. *Optik-International Journal for Light and Electron Optics*, 126, 1 (2015), 87–94. (cited on page 129)

Yamaguchi, K.; McAllester, D.; and Urtasun, R., 2013. Robust monocular epipolar flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1862–1869. (cited on pages 63 and 73)

Yan, Y.; Ren, W.; Guo, Y.; Wang, R.; and Cao, X., 2017a. Image deblurring via extreme channels prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xviii, xix, xx, xxiii, xxvi, 10, 18, 19, 30, 31, 32, 33, 34, 35, 41, 46, 47, 48, 50, 51, 58, 85, 93, 97, 105, 120, and 121)

Yan, Y.; Xu, C.; Cai, D.; and Corso, J. J., 2017b. Weakly supervised actor-action segmentation via robust multi-task ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1298–1307. (cited on page 61)

Yang, J.; Ye, X.; Li, K.; Hou, C.; and Wang, Y., 2014. Color-guided depth recovery from rgb-d data using an adaptive autoregressive model. *IEEE Transactions on Image Processing*, 23, 8 (2014), 3443–3458. (cited on page 47)

Yin, Z. and Shi, J., 2018. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages 14, 62, 127, and 128)

Yu, F. and Gallup, D., 2014. 3d reconstruction from accidental motion. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 39)

Yu, X.; Xu, F.; Zhang, S.; and Zhang, L., 2014. Efficient patch-wise non-uniform deblurring for a single image. *IEEE Transactions on Multimedia*, 16, 6 (2014), 1510–1524. (cited on pages 85 and 104)

Zhang, H.; Dai, Y.; Li, H.; and Koniusz, P., 2019. Deep stacked hierarchical multi-patch network for image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xxvi, xxvii, 11, 131, 138, and 141)

Zhang, J.; Pan, J.; Ren, J.; Song, Y.; Bao, L.; Lau, R. W.; and Yang, M.-H., 2018. Dynamic scene deblurring using spatially variant recurrent neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on pages xxix, 21, 93, 95, 117, and 121)

Zheng, S.; Xu, L.; and Jia, J., 2013. Forward motion deblurring. In *Proceedings of the IEEE International Conference on Computer Vision*. (cited on pages 10, 19, 20, 39, and 41)

Zhong, Y.; Dai, Y.; and Li, H., 2017. Self-supervised learning for stereo matching with self-improving ability. *arXiv preprint arXiv:1709.00930*, (2017). (cited on pages 47, 48, and 50)

Zhou, D.; Frémont, V.; Quost, B.; Dai, Y.; and Li, H., 2017. Moving object detection and segmentation in urban environments from a moving platform. *Image and Vision Computing*, (2017). (cited on pages 75 and 76)

Zhou, S.; Zhang, J.; Pan, J.; Xie, H.; Zuo, W.; and Ren, J., 2019a. Spatio-temporal filter adaptive network for video deblurring. *arXiv preprint arXiv:1904.12257*, (2019). (cited on page 12)

Zhou, S.; Zhang, J.; Zuo, W.; Xie, H.; Pan, J.; and Ren, J., 2019b. Davanet: Stereo deblurring with view aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 129)

Zhou, S.; Zhang, J.; Zuo, W.; Xie, H.; Pan, J.; and Ren, J. S., 2019c. Davanet: Stereo deblurring with view aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. (cited on page 12)

Zhou, Y. and Komodakis, N., 2014. A map-estimation framework for blind deblurring using high-level edge priors. In *European Conference on Computer Vision*, 142–157. Springer. (cited on page 56)

Zhu, A.; Yuan, L.; Chaney, K.; and Daniilidis, K., 2018a. Ev-flownet: Self-supervised optical flow estimation for event-based cameras. In *Proceedings of Robotics: Science and Systems*. Pittsburgh, Pennsylvania. doi:10.15607/RSS.2018.XIV.062. (cited on pages xxvi, xxvii, xxx, 12, 14, 84, 103, 126, 127, 128, 138, 139, 140, and 141)

Zhu, A. Z.; Atanasov, N.; and Daniilidis, K., 2017. Event-based visual inertial odometry. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5816–5824. (cited on pages 12, 84, and 103)

Zhu, A. Z.; Thakur, D.; Özaslan, T.; Pfrommer, B.; Kumar, V.; and Daniilidis, K., 2018b. The multivehicle stereo event camera dataset: An event camera dataset for 3d perception. *IEEE Robot. Autom. Lett.*, 3, 3 (2018), 2032–2039. (cited on page 137)

Zhu, A. Z.; Yuan, L.; Chaney, K.; and Daniilidis, K., 2019. Unsupervised event-based learning of optical flow, depth, and egomotion. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 989–997. (cited on pages 14, 127, 128, 138, 139, and 140)