Single Image Optical Flow Estimation with an Event Camera

Liyuan Pan[1,2], Miaomiao Liu[1,2], and Richard Hartley[1,2]

[1] Australian National University, Canberra, Australia,   [2] Australian Centre for Robotic Vision

`liyuan.pan@anu.edu.au`

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 paper, we propose a single image (potentially blurred) and events 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.

1. Introduction

Event cameras (such as DVS [28] and DAVIS [7]) 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 [42, 18], deblurring [38], and SLAM [26, 27, 50]. 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 paper, we aim to show how events can
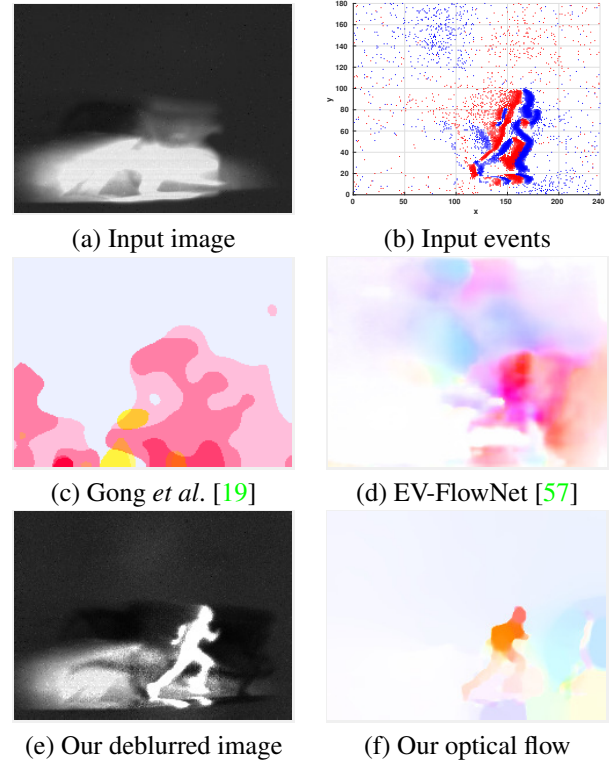

(a) Input image

(b) Input events

(c) Gong *et al.* [19]

(d) EV-FlowNet [57]

(e) Our deblurred image

(f) Our optical flow

Figure 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 [19] by using a single blurred image. (d) Flow result of [57], by using events. (e) and (f) are our results. Our methods is able to handle large motion scenery. (Best viewed on screen).*

improve flow estimates, even with a blurred image.

Optical flow estimation is an active topic in the computer vision community and serves as the backbone for event-based moving object segmentation [47], human pose estimation [11], and action recognition [1]. Traditional flow estimation approaches [20, 25, 54] are proposed based on the brightness consistency assumption for corresponding pixels across the image pair, and cannot handle the asynchronous event data [15]. A common trend [2, 16, 59, 17] to esti-

1

mate 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) [29]. Hence, several methods tends to fuse the intensity information and events [2, 3] to estimate flow.

To this end, we aim to utilize the output of DAVIS, which is events and intensity images, to improve optical flow estimates. A straightforward idea is to reconstruct images from events [38, 43], 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. 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*. [57] 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. 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 a *event-based brightness constancy* constraint on absolute intensity to encode the relation between optical flow and the event data. Besides, we utilize 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.

2. 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*. [6] propose an adaptation of the gradient-based Lucas-Kanade algorithm based on DVS. In [5], 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*. [4] 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*. [2] 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*. [16] present a unifying framework to estimate flow by finding the point trajectories on each image plane that are best aligned with events. Zhu *et al*. propose EV-FlowNet [57], 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 [59], 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 [13, 25, 54] by video. FlowNet 2.0 [24] develops a stacked architecture that includes warping of the second image with the intermediate flow. PWC-Net [48] 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 [30] 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 [51, 44, 14]. Walker *et al*. [51] use CNN to predict dense flow, while they assume the image is static.

Event-based image reconstruction. Image reconstruction [43, 52, 37] 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 [8, 45, 38]. Pan *et al*. [38] 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 re-

lies on the quality of the estimated flow. Researchers attempt to use flow to estimate the spatial-varying blur kernel and then restore images [53, 21, 22, 46, 35, 36, 34]. Recently, learning-based methods have brought significant improvements in image deblurring [19, 33, 56]. Gong *et al*. [19] 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.

3. 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

$$\mathbf{L}(\mathbf{x}, f) = \mathbf{L}(\mathbf{x} + \mathbf{u}(\mathbf{x}), t) \,, \tag{1}$$

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 (1) is under-determined, regularization terms are introduced to solve optical flow. Horn and Schunck [20] studied a variational formulation of the problem,

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

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.

4. 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 [38]. 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 [19], 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:

$$\min_{\mathbf{L}, \mathbf{u}} \mu_1 \phi_\mathrm{eve}(\mathbf{L}, \mathbf{u}) + \mu_2 \phi_\mathrm{blur}(\mathbf{L}, \mathbf{u}) + \phi_\mathrm{flow}(\nabla \mathbf{u}) + \phi_\mathrm{im}(\nabla \mathbf{L}) \,, \tag{3}$$

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. (3).

4.1. Brightness Constancy by Event Data ϕ_eve

In case of the output data from DAVIS, we represent Eq. (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.

$$|\log(\mathbf{L}(\mathbf{x}, t)) - \log(\mathbf{L}(\mathbf{x}, t_{ref})| \geq c \,. \tag{4}$$

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 [38], we represent the neighbouring image as

$$\mathbf{L}(\mathbf{x}, t) = \mathbf{L}(\mathbf{x}, f) \exp(c \, \mathbf{E}(\mathbf{x}, t)) \,, \tag{5}$$

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. (1) and obtain its approximation

$$\mathbf{L}(\mathbf{x} + \mathbf{u}(\mathbf{x}), f + \triangle t)$$
$$\approx \mathbf{L}(\mathbf{x}, f) + u_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(x)} + v_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(y)} + \triangle t \, \mathbf{L}(\mathbf{x}, f)^{(t)}$$
$$= u_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(x)} + v_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(y)} + \mathbf{L}(\mathbf{x}, t) \,. \tag{6}$$

Back to the left-hand side of Eq. (1), we have

$$\mathbf{L}(\mathbf{x}, f) \approx u_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(x)} + v_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(y)} + \mathbf{L}(\mathbf{x}, t) \,. \tag{7}$$

With the event model in Eq. (5), we can form the latent image as,

$$\mathbf{L}(\mathbf{x}, f) \approx u_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(x)} + v_\mathbf{x} \mathbf{L}(\mathbf{x}, f)^{(y)}$$
$$+ \mathbf{L}(\mathbf{x}, f) \exp(c \, \mathbf{E}(\mathbf{x}, t)) \,.$$

(a) Blurred image (b) Deblurred image [55]

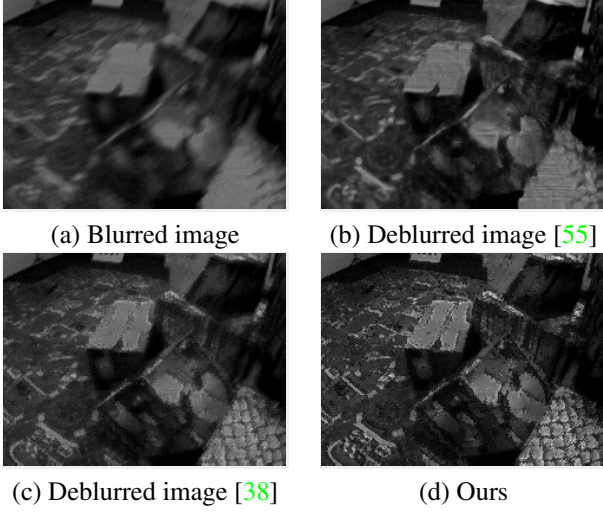(c) Deblurred image [38] (d) Ours

Figure 2. *An example of our deblurring result on the real dataset [31]. (a) The blurred image. (b) Deblurred by Zhang et al. [55]. (c) Deblurred by EDI [38]. (d) Ours. (Best viewed on screen).*

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

$$\phi_{\mathrm{eve}}(\mathbf{L}, \mathbf{u}) = \sum_{\mathbf{x} \in \Omega} \| \mathbf{L}(\mathbf{x}, f)(\exp(c\,\mathbf{E}(\mathbf{x}, t)) - 1) \\ + [u_{\mathbf{x}}, v_{\mathbf{x}}]^{\mathrm{T}} \nabla \mathbf{L}(\mathbf{x}, f) \|_1 \, . \quad (8)$$

Different with [2, 17, 9] 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.

4.2. Blur Image Formation Constraint ϕ_{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

$$\mathbf{B} = \mathbf{k} \otimes \mathbf{L}(f) \, , \quad (9)$$

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

$$\mathbf{B}(\mathbf{x}) = \mathbf{k}(\mathbf{x}) \otimes \mathbf{L}(\mathbf{x}) \, . \quad (10)$$

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

$$\mathbf{B}(\mathbf{x}) = \sum_{\mathbf{y} \in \Omega} \mathbf{k}(\mathbf{y}) \mathbf{L}(\mathbf{x} - \mathbf{y}) \\ = \sum_{\mathbf{y} \in \Omega} \mathbf{k}_{\mathbf{u}'(\mathbf{x})}(\mathbf{y}) \mathbf{L}(\mathbf{x} - \mathbf{y}) \, , \quad (11)$$

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

$$k_{\mathbf{u}'(\mathbf{x})}(\mathbf{y}) = \begin{cases} \dfrac{1}{|\mathbf{u}'(\mathbf{x})|}, & \text{if } \mathbf{y} = \alpha \mathbf{u}'(\mathbf{x}), \ |\alpha| \leq \dfrac{1}{2} \\ \mathbf{0}, & \text{otherwise} , \end{cases} \quad (12)$$

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. (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

$$\phi_{\mathrm{blur}}(\mathbf{L}, \mathbf{u}) = \sum_{\mathbf{x}, \mathbf{y} \in \Omega} \| \mathbf{k}_{\mathbf{u}'(\mathbf{x})}(\mathbf{y}) \mathbf{L}(\mathbf{x} - \mathbf{y}) - \mathbf{B}(\mathbf{x}) \|^2 \, , \quad (13)$$

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

4.3. 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 [23]. 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

$$\phi_{\mathrm{flow}}(\nabla \mathbf{u}) = \|w\nabla \mathbf{u}\|_{1,2} = \sum_{\mathbf{x} \in \Omega} \|w(\mathbf{x})\nabla \mathbf{u}(\mathbf{x})\| , \quad (14)$$

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

$$w^x = \mu_3 \exp(-(\hat{\mathbf{L}}^{(x)}/\mu_4)^2) \, , \quad (15)$$

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

$$\phi_{\mathrm{im}}(\nabla \mathbf{L}) = \sum_{\mathbf{x} \in \Omega} \|\nabla \mathbf{L}(\mathbf{x})\|_1 \, . \quad (16)$$

5. Optimization

Clearly, Eq. (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. (17) (See Section 5.1).
- Fix optical flow $\mathbf{u}$, and compute the latent image by optimizing Eq. (24) (See Section 5.2).

Here, we use the primal-dual algorithm [40, 39, 12] for its optimal convergence. In the following section, we describe details for each optimization step.

5.1. Optical Flow Estimation

We fix the image, namely $\mathbf{L} = \hat{\mathbf{L}}$, and Eq. (3) reduces to

$$\min_{\mathbf{u}} \underbrace{\mu_1 \phi_{\text{eve}}(\mathbf{u}) + \mu_2 \phi_{\text{blur}}(\mathbf{u})}_{G(\mathbf{u})} + \underbrace{\phi_{\text{flow}}(\nabla \mathbf{u})}_{F(K\mathbf{u})}, \quad (17)$$

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. (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. (17) by its double Fenchel dual F^{**}, so it becomes $\min_{\mathbf{u} \in X}(G(\mathbf{u}) + F^{**}(K\mathbf{u}))$, which is

$$\min_{\mathbf{u} \in X} \left(G(\mathbf{u}) + \max_{\mathbf{p} \in Y} \langle K\mathbf{u}, \mathbf{p} \rangle_X - F^*(\mathbf{p}) \right). \quad (18)$$

Recall that the Fenchel dual (convex conjugate) F^* of function F is defined as

$$F^*(\mathbf{q}) = \sup_{\mathbf{p} \in Y} (\langle \mathbf{p}, \mathbf{q} \rangle - F(\mathbf{p})), \quad (19)$$

and that $F = F^{**}$ if F is a convex function (a norm is convex). The primal-dual algorithm of [12] consists of iterations starting from initial estimates $\mathbf{u}^0$, $\mathbf{p}^0$ and $\bar{\mathbf{u}}^0 = \mathbf{u}^0$:

$$\begin{aligned} \mathbf{p}^{n+1} &= \mathcal{P}_{F^*}(\mathbf{p}^n + \sigma K\bar{\mathbf{u}}^n) \\ \mathbf{u}^{n+1} &= \mathcal{P}_G(\mathbf{u}^n - \tau K^*\mathbf{p}^{n+1}) \\ \bar{\mathbf{u}}^{n+1} &= \mathbf{u}^{n+1} + \theta(\mathbf{u}^{n+1} - \mathbf{u}^n). \end{aligned} \quad (20)$$

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 $\| \cdot \|_{\infty,2}$ (details can be found in the supplementary material). The indicator function is therefore a product B^N of N Euclidean 2-balls (each in $\mathbb{R}^4$). More precisely

$$F^*(\mathbf{p}) = \begin{cases} 0, & \text{if } \|\mathbf{p_x}\| \leq 1 \text{ for all } \mathbf{x} \\ +\infty, & \text{otherwise}. \end{cases} \quad (21)$$

The proximal operator $\mathcal{P}_{F^*}$ is therefore given by

$$\begin{aligned} F^*(\bar{\mathbf{p}}) &= \arg\min_{\mathbf{p} \in Y} (2F^*(\mathbf{p}) + \|\bar{\mathbf{p}} - \mathbf{p}\|^2) \\ &= \arg\min_{\mathbf{p} \in B^N} \|\bar{\mathbf{p}} - \mathbf{p}\|^2. \end{aligned} \quad (22)$$

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. (20) is

$$\begin{aligned} \bar{\mathbf{u}} &= \mathbf{u}^n - \tau K^*\mathbf{p}^{n+1} \\ \mathbf{u}^{n+1} &= \mathcal{P}_{\tau G}(\bar{\mathbf{u}}) = \arg\min_{\mathbf{u}} (2\tau G(\mathbf{u}) + \|\mathbf{u} - \bar{\mathbf{u}}\|^2). \end{aligned}$$

(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

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \tau(\nabla G(\mathbf{u}^n) + K^*\mathbf{p}^{n+1}), \quad (23)$$

which is simply gradient descent of Eq. (18), fixing $\mathbf{p} = \mathbf{p}^{n+1}$. We obtain Algorithm 1.

Algorithm 1: 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

1 **while** $n < 20$ **do**
2   Dual ascent in $\mathbf{p}$
3      $\bar{\mathbf{p}} = \mathbf{p}^n + \sigma K\bar{\mathbf{u}}^n$, $\mathbf{p}_\mathbf{x}^{n+1} = \bar{\mathbf{p}}_\mathbf{x}/\max(1, \|\bar{\mathbf{p}}_\mathbf{x}\|) \,\forall \mathbf{x}$
4   Primal descent in $\mathbf{u}$
5      $\mathbf{u}^{n+1} = \mathbf{u}^n - \tau(G(\mathbf{u}^n) + K^*\mathbf{p}^{n+1})$
6   Extrapolation step
7      $\bar{\mathbf{u}}^{n+1} = \mathbf{u}^{n+1} + (\mathbf{u}^{n+1} - \mathbf{u}^n)$
8      $n = n + 1$
9 **end**

5.2. Deblurring

We fix optical flow, namely $\mathbf{u} = \hat{\mathbf{u}}$, and Eq. (3) reduces to

$$\min_{\mathbf{L}} \underbrace{\phi_{\text{im}}(\nabla \mathbf{L})}_{F_1(\nabla \mathbf{L})} + \underbrace{\mu_1 \phi_{\text{eve}}(\mathbf{L})}_{F_2(K\mathbf{L})} + \underbrace{\mu_2 \phi_{\text{blur}}(\mathbf{L})}_{G(\mathbf{L})}. \quad (24)$$

The convex conjugate F^* is defined as,

$$F^*(\mathbf{p}, \mathbf{q}) = F_1^*(\mathbf{p}) + F_2^*(\mathbf{q}), \quad (25)$$

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,

$$\mathbf{p}^{n+1} = \frac{\mathbf{p}^n + \gamma \nabla \bar{\mathbf{L}}^n}{\max(1, \mathrm{abs}(\mathbf{p}^n + \gamma \nabla \bar{\mathbf{L}}^n))},$$

$$\mathbf{q}^{n+1} = \frac{\mathbf{q}^n + \gamma(\theta_2 \bar{\mathbf{L}}^n + [u,\ v]^{\mathrm{T}} \nabla \bar{\mathbf{L}}^n)}{\max(1, \mathrm{abs}(\mathbf{q}^n + \gamma(\theta_2 \bar{\mathbf{L}}^n + [u,\ v]^{\mathrm{T}} \nabla \bar{\mathbf{L}}^n)))},$$
(26)

where η, γ are weight factors, and $\theta_2 = \exp(c\mathbf{E}(t)) - 1$.

$$\mathbf{L}^{n+1} = \mathcal{P}_{\eta G}(\bar{\mathbf{L}}) = \arg\min_{\mathbf{L}} \left(2\eta G(\mathbf{L}) + \|\mathbf{L} - \bar{\mathbf{L}}\|^2\right),$$
(27)

where $\bar{\mathbf{L}} = \mathbf{L}^n - \eta(\nabla^* \mathbf{p}^{n+1} + K^* \mathbf{q}^{n+1})$. We obtain Algorithm 2 for the minimization of the proposed energy function (24).

Algorithm 2: Primal-Dual Minimization - Deblurring

Initialization: Choose $\gamma, \eta > 0$, $n = 0$, and set $\bar{\mathbf{L}}^0 = \mathbf{L}^0$.
Iterations : Update $\mathbf{L}^n, \mathbf{p}^n, \mathbf{q}^n$ as follows

1 **while** $n < 5$ **do**
2   Dual ascent in $\mathbf{p}, \mathbf{q}$
3     $\bar{\mathbf{p}} = \mathbf{p}^n + \gamma \nabla \bar{\mathbf{L}}^n,\ \bar{\mathbf{q}} = \mathbf{q}^n + \gamma(\theta_2 \bar{\mathbf{L}}^n + [u,\ v]^{\mathrm{T}} \nabla \bar{\mathbf{L}}^n)$
4     $\mathbf{p}_{\mathbf{x}}^{n+1} = \bar{\mathbf{p}}_{\mathbf{x}}/\max(1, \mathrm{abs}(\bar{\mathbf{p}}_{\mathbf{x}}))\ \forall \mathbf{x}$
5     $\mathbf{q}_{\mathbf{x}}^{n+1} = \bar{\mathbf{q}}_{\mathbf{x}}/\max(1, \mathrm{abs}(\bar{\mathbf{q}}_{\mathbf{x}}))\ \forall \mathbf{x}$
6   Primal descent in $\mathbf{L}$
7     $\bar{\mathbf{L}} = \mathbf{L}^n - \eta(\nabla^* \mathbf{p}^{n+1} + K^* \mathbf{q}^{n+1}),\ \mathbf{L}^{n+1} = \mathcal{P}_{\eta G}(\bar{\mathbf{L}})$
8   Extrapolation step
9     $\bar{\mathbf{L}}^{n+1} = \mathbf{L}^{n+1} + (\mathbf{L}^{n+1} - \mathbf{L}^n)$
10     $n = n + 1$
11 **end**

---

6. Experiments

6.1. Experimental Setup

Real dataset. We evaluate our method on three public real event datasets, namely, Multi-vehicle Stereo Event Camera dataset (MVSEC) [58], Event-Camera dataset (ECD) [31], and Blurred Event Dataset (BED) [45, 38]. 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 [10] with images of size 1024×436, which uses the event simulator ESIM [41] 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 [32], 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) (seeing details in the supplementary material). 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 [57], and Zhu *et al*. [59]. Then, we compare with the state-of-the-art video (with the label 'GT images'.) only based method SelFlow [30], and PWC-Net [48]. 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 [38] to restore intensity images, then applying flow estimation methods [48, 30] to the restored images to estimate flow. We compare our deblurring results with the state-of-the-art event-based deblurring approach [38] and blind deblurring methods [55, 49, 19].

Implementation details. For all our real experiments, the image and events 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.

6.2. 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 1, 3 and qualitative comparisons in Fig. 1, 2, 3, 4 to show the effectiveness and generalization of our method. Ablation study in Table 2 shows the effectiveness of each term in our objective function (3).

As shown in Table 1 and Fig. 3, we achieve competitive results on flow estimation compared with event only based methods [57, 59] on MVSEC dataset. Note that models in [57, 59] 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. 1 and 2, which demonstrate the stability of our model under both blurred and non-blurred conditions.

Table 1. *Results on the MVSEC [57] and Sintel dataset [10]. 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 [57] 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 [57]						
Input	GT images		EDI images and events		Events		
	SelFlow [30]	PWC-Net [48]	SelFlow [30]	PWC-Net [48]	EV-FlowNet [57]	Zhu *et al.* [59]	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 [10]						
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**

(a) Input image (b) Input events (c) Ground-truth flow (d) Error map

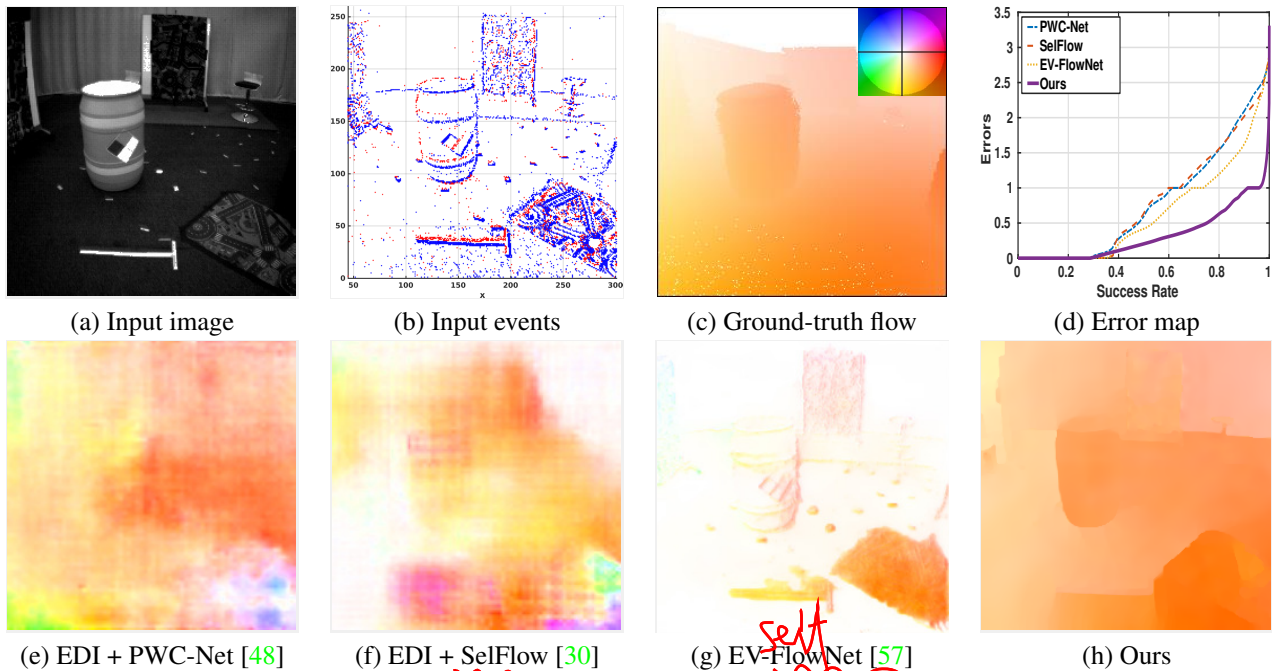(e) EDI + PWC-Net [48] (f) EDI + SelFlow [30] (g) EV-FlowNet [57] (h) Ours

Figure 3. *Results of our method compared with state-of-the-art methods on real dataset [57]. (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 [48] based on two reconstructed images. The reconstructed image is estimated by EDI model [38] from a single image and its events. (f) Baseline: Flow result by [30] based on two reconstructed images. (g) Flow result by [57] based on images and events. (h) Ours, by using an image and events as input. (Best viewed on screen).*

Table 2. *Ablation Study based on Sintel Dataset [10].*

	without ϕ_{eve}	without ϕ_{blur}
AEE	2.3941	2.2594
MSE	5.3506	9.5267
FE (%)	18.0525	45.4516

We show flow comparisons in Table 1 and Fig. 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 3 and Fig. 5,

we provide deblurring comparisons on GoPro dataset [32]. Our approach outperform all the 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. More results can be found in the supplementary material.

Ablation Study. To provide a deep understanding of our model, we evaluate the influence of ϕ_{eve} and ϕ_{blur} in Table

Table 3. *Quantitative analysis on the GoPro dataset [32]. 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 [30]	PWC-Net [48]	EV-FlowNet [57]	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**

(a) Input image   (b) Input events   (c) Ground-truth flow   (d) EDI + PWC-Net [48]   (e) EDI + SelFlow [30]

(f) Ours - Without ϕ_{eve} (g) Ours - Without ϕ_{blur} (h) Error map   (i) Ours - deblurred   (j) Ours - optical flow
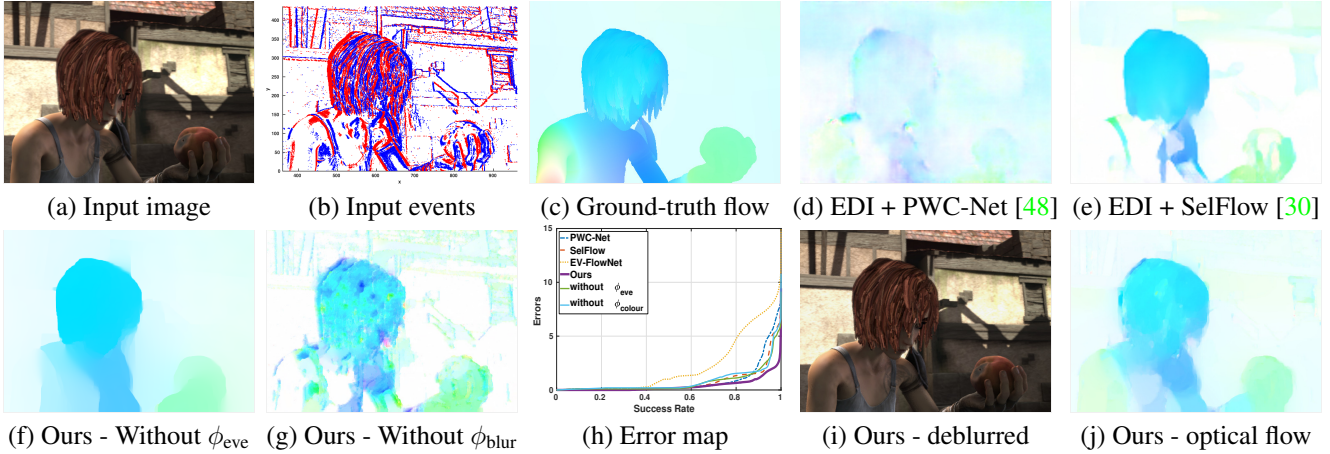
Figure 4. *An example of our method on dataset [10]. (a) Input blurred image. (b) Input events. (c) Ground-truth optical flow. (d) Flow result by [48] based on images estimated by EDI model [38]. (e) Flow result by [30] based on images estimated by EDI model. (f) Ours baseline result without term ϕ_{eve}. (g) Ours baseline result without term ϕ_{blur}. (h) Error Map. (i) Our deblurring result. (j) Our optical flow.*

(a)   (b)   (c)   (d)   (e)
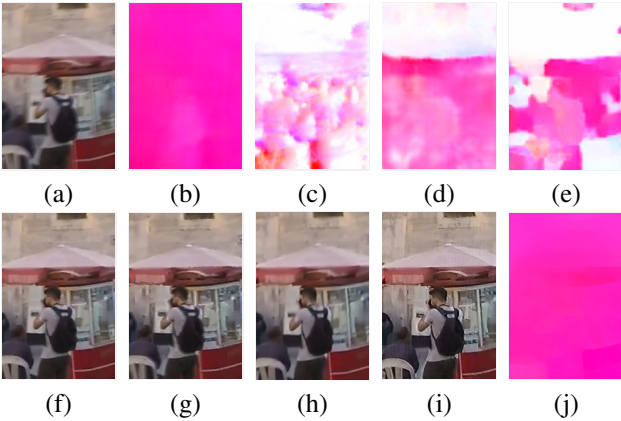
(f)   (g)   (h)   (i)   (j)

Figure 5. *An example of our method on dataset [32]. (a) The blurred image. (b) The ground-truth flow. (c) Flow result by [57], using the events as input. (d) Flow result by [48] based on images estimated by the EDI model [38]. (e) Flow result by [30] based on images estimated by the EDI model. (f) The ground-truth latent images at time t. (g)Deblurred result by [38]. (h) Deblurred result by [55]. (i) Our deblurred image. (j) Our estimated optical flow.*

2. The significantly decreased performance indicates the contribution of each term in our model. In Table 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 [38] and our initial flow is simply computed using Eq. (2) on event frames.

7. Conclusion

In this paper, 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.

8. Acknowledgment

References

[1] A. Amir, B. Taba, D. Berg, T. Melano, J. McKinstry, C. Di Nolfo, T. Nayak, A. Andreopoulos, G. Garreau, M. Mendoza, J. Kusnitz, M. Debole, S. Esser, T. Delbruck, M. Flickner, and D. Modha. A low power, fully event-based gesture recognition system. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017. 1

[2] P. Bardow, A. J. Davison, and S. Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 884–892, 2016. 1, 2, 4

[3] F. Barranco, C. Fermüller, and Y. Aloimonos. Contour motion estimation for asynchronous event-driven cameras. *Proceedings of the IEEE*, 102(10):1537–1556, 2014. 2

[4] F. Barranco, C. Fermuller, and Y. Aloimonos. Bio-inspired motion estimation with event-driven sensors. In *International Work-Conference on Artificial Neural Networks*, pages 309–321. Springer, 2015. 2

[5] R. Benosman, C. Clercq, X. Lagorce, S.-H. Ieng, and C. Bartolozzi. Event-based visual flow. *IEEE Trans. Neural Netw. Learn. Syst.*, 25(2):407–417, 2013. 2

[6] R. Benosman, S.-H. Ieng, C. Clercq, C. Bartolozzi, and M. Srinivasan. Asynchronous frameless event-based optical flow. *Neural Networks*, 27:32–37, 2012. 2

[7] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck. A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014. 1

[8] C. Brandli, L. Muller, and T. Delbruck. Real-time, high-speed video decompression using a frame- and event-based DAVIS sensor. In *IEEE Int. Symp. Circuits Syst. (ISCAS)*, pages 686–689, June 2014. 2

[9] S. Bryner, G. Gallego, H. Rebecq, and D. Scaramuzza. Event-based, direct camera tracking from a photometric 3d map using nonlinear optimization. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 325–331. IEEE, 2019. 4

[10] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *Eur. Conf. Comput. Vis. (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012. 6, 7, 8

[11] E. Calabrese, G. Taverni, C. Awai Easthope, S. Skriabine, F. Corradi, L. Longinotti, K. Eng, and T. Delbruck. Dhp19: Dynamic vision sensor 3d human pose dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 1

[12] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011. 5

[13] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 2758–2766, 2015. 2

[14] Y. Endo, Y. Kanamori, and S. Kuriyama. Animating landscape: Self-supervised learning of decoupled motion and appearance for single-image video synthesis. *arXiv preprint arXiv:1910.07192*, 2019. 2

[15] G. Gallego, T. Delbruck, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. Davison, J. Conradt, K. Daniilidis, et al. Event-based vision: A survey. *arXiv preprint arXiv:1904.08405*, 2019. 1

[16] G. Gallego, H. Rebecq, and D. Scaramuzza. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2018. 1, 2

[17] D. Gehrig, A. Loquercio, K. G. Derpanis, and D. Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5633–5643, 2019. 1, 4

[18] D. Gehrig, H. Rebecq, G. Gallego, and D. Scaramuzza. Asynchronous, photometric feature tracking using events and frames. In *Eur. Conf. Comput. Vis. (ECCV)*, 2018. 1

[19] D. Gong, J. Yang, L. Liu, Y. Zhang, I. Reid, C. Shen, A. van den Hengel, and Q. Shi. From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 2319–2328, 2017. 1, 3, 6

[20] B. K. Horn and B. G. Schunck. Determining optical flow. *J. Artificial Intell.*, 17(1-3):185–203, 1981. 1, 3

[21] T. Hyun Kim and K. Mu Lee. Segmentation-free dynamic scene deblurring. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 2766–2773, 2014. 3

[22] T. Hyun Kim and K. Mu Lee. Generalized video deblurring for dynamic scenes. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5426–5434, 2015. 3

[23] T. Hyun Kim, H. Seok Lee, and K. Mu Lee. Optical flow via locally adaptive fusion of complementary data costs. In *Int. Conf. Comput. Vis. (ICCV)*, pages 3344–3351, 2013. 4

[24] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, July 2017. 2

[25] J. Y. Jason, A. W. Harley, and K. G. Derpanis. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *European Conference on Computer Vision Workshops*, pages 3–10. Springer, 2016. 1, 2

[26] H. Kim, S. Leutenegger, and A. J. Davison. Real-time 3D reconstruction and 6-DoF tracking with an event camera. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 349–364, 2016. 1

[27] B. Kueng, E. Mueggler, G. Gallego, and D. Scaramuzza. Low-latency visual odometry using event-based feature tracks. In *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, pages 16–23. IEEE, 2016. 1

[28] P. Lichtsteiner, C. Posch, and T. Delbruck. A 128×128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE journal of solid-state circuits*, 43(2):566–576, 2008. 1

[29] M. Liu and T. Delbruck. Adaptive time-slice block-matching optical flow algorithm for dynamic vision sensors. In *British Machine Vis. Conf. (BMVC)*, 2018. 2

[30] P. Liu, M. Lyu, I. King, and J. Xu. Selflow: Self-supervised learning of optical flow. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2019. 2, 6, 7, 8

[31] E. Mueggler, H. Rebecq, G. Gallego, T. Delbruck, and D. Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *Int. J. Robot. Research*, 36(2):142–149, 2017. 4, 6

[32] S. Nah, T. H. Kim, and K. M. Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, July 2017. 6, 7, 8

[33] S. Nah, S. Son, and K. M. Lee. Recurrent neural networks with intra-frame iterations for video deblurring. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2019. 3

[34] L. Pan, Y. Dai, and M. Liu. Single image deblurring and camera motion estimation with depth map. In *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, pages 2116–2125. IEEE, 2019. 3

[35] L. Pan, Y. Dai, M. Liu, and F. Porikli. Simultaneous stereo video deblurring and scene flow estimation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, July 2017. 3

[36] L. Pan, Y. Dai, M. Liu, F. Porikli, and Q. Pan. Joint stereo video deblurring, scene flow estimation and moving object segmentation. *IEEE Transactions on Image Processing*, 29:1748–1761, 2019. 3

[37] L. Pan, R. Hartley, C. Scheerlinck, M. Liu, X. Yu, and Y. Dai. High frame rate video reconstruction based on an event camera. *arXiv preprint arXiv:1903.06531*, 2019. 2

[38] L. Pan, C. Scheerlinck, X. Yu, R. Hartley, M. Liu, and Y. Dai. Bringing a blurry frame alive at high frame-rate with an event camera. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2019. 1, 2, 3, 4, 6, 7, 8

[39] T. Pock, A. Chambolle, D. Cremers, and H. Bischof. A convex relaxation approach for computing minimal partitions. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 810–817. IEEE, 2009. 5

[40] T. Pock, D. Cremers, H. Bischof, and A. Chambolle. An algorithm for minimizing the mumford-shah functional. In *Int. Conf. Comput. Vis. (ICCV)*, pages 1133–1140. IEEE, 2009. 5

[41] H. Rebecq, D. Gehrig, and D. Scaramuzza. Esim: an open event camera simulator. In *Conference on Robot Learning*, pages 969–982, 2018. 6

[42] H. Rebecq, T. Horstschäfer, G. Gallego, and D. Scaramuzza. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time. *IEEE Robotics and Automation Letters*, 2(2):593–600, 2016. 1

[43] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2019. 2

[44] P. Rosello. Predicting future optical flow from static video frames. *Retrieved on: Jul*, 18, 2016. 2

[45] C. Scheerlinck, N. Barnes, and R. Mahony. Continuous-time intensity estimation using event cameras. In *Proc. Asian Conf. Comp. Vis.*, December 2018. 2, 6

[46] A. Sellent, C. Rother, and S. Roth. Stereo video deblurring. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 558–575. Springer, 2016. 3

[47] T. Stoffregen, G. Gallego, T. Drummond, L. Kleeman, and D. Scaramuzza. Event-based motion segmentation by motion compensation. In *Int. Conf. Comput. Vis. (ICCV)*, pages 7244–7253, 2019. 1

[48] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2018. 2, 6, 7, 8

[49] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia. Scale-recurrent network for deep image deblurring. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2018. 6

[50] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza. Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios. *IEEE Robot. Autom. Lett.*, 3(2):994–1001, 2018. 1

[51] J. Walker, A. Gupta, and M. Hebert. Dense optical flow prediction from a static image. In *Int. Conf. Comput. Vis. (ICCV)*, pages 2443–2451, 2015. 2

[52] L. Wang, S. M. M. I. , Y.-S. Ho, and K.-J. Yoon. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2019. 2

[53] Y. Xu, X. Hu, and S. Peng. Blind motion deblurring using optical flow. *Optik-International Journal for Light and Electron Optics*, 126(1):87–94, 2015. 3

[54] Z. Yin and J. Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2018. 1, 2

[55] H. Zhang, Y. Dai, H. Li, and P. Koniusz. Deep stacked hierarchical multi-patch network for image deblurring. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2019. 4, 6, 8

[56] S. Zhou, J. Zhang, W. Zuo, H. Xie, J. Pan, and J. Ren. Davanet: Stereo deblurring with view aggregation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019. 3

[57] A. Zhu, L. Yuan, K. Chaney, and K. Daniilidis. Ev-flownet: Self-supervised optical flow estimation for event-based cameras. In *Robotics: Science and Systems (RSS)*, Pittsburgh, Pennsylvania, June 2018. 1, 2, 6, 7, 8

[58] A. Z. Zhu, D. Thakur, T. Özaslan, B. Pfrommer, V. Kumar, and K. Daniilidis. The multivehicle stereo event camera dataset: An event camera dataset for 3d perception. *IEEE Robot. Autom. Lett.*, 3(3):2032–2039, 2018. 6

[59] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 989–997, 2019. 1, 2, 6, 7