Video Interpolation by Event-driven Anisotropic Adjustment of Optical Flow

Song Wu*[1], Kaichao You*[2], Weihua He**[2], Chen Yang[1], Yang Tian[2], Yaoyuan Wang[1], Ziyang Zhang**[1], and Jianxing Liao[1]

[1] Advanced Computing and Storage Lab, Huawei Technologies Co. Ltd
{wusong5533, yangchen2017.pku}@gmail.com
{wangyaoyuan1,zhangziyang11,liaojianx}@huawei.com
[2] Tsinghua University
{ykc20,hwh20,tiany20}@mails.tsinghua.edu.cn

Abstract. Video frame interpolation is a challenging task due to the ever-changing real-world scene. Previous methods often calculate the bi-directional optical flows and then predict the intermediate optical flows under the linear motion assumptions, leading to isotropic intermediate flow generation. Follow-up research obtained anisotropic adjustment through estimated higher-order motion information with extra frames. Based on the motion assumptions, their methods are hard to model the complicated motion in real scenes. In this paper, we propose an end-to-end training method **A^{2}OF** for video frame interpolation with event-driven **A**nisotropic **A**djustment of **O**ptical **F**lows. Specifically, we use events to generate optical flow distribution masks for the intermediate optical flow, which can model the complicated motion between two frames. Our proposed method outperforms the previous methods in video frame interpolation, taking supervised event-based video interpolation to a higher stage.

Keywords: Video Frame Interpolation · Bi-directional Optical Flow · Event-driven Distribution Mask

1 Introduction

Video frame interpolation(VFI) is a challenging task in computer vision, which is widely used in slow motion video generation, high rate frame conversion and video frames recovery, etc. The goal of VFI is to synthesize nonexistent intermediate frames between two consecutive frames. However, it is hard to synthesize high-quality intermediate frames due to the lack of corresponding motion information.

Optical flow is a common tool in VFI. SuperSloMo [7] linearly represent the intermediate optical flow from the target frame to original frame with bi-directional optical flow. They warp, blend and refine the original frames to obtain

* Song Wu and Kaichao You contribute equally to this paper. Work done while Song Wu, Kaichao You, Yang Tian are interns at Huawei.

** Weihua He and Ziyang Zhang are corresponding authors.

target frames. The key assumption in their method is uniform motion along a straight line, which runs counter to the laws of nonlinear motion in real world. Besides, they synthesize the intermediate optical flow with the same coefficient to the different directions in bi-directional optical flows. In this isotropic way, they can not obtain the inaccurate intermediate flow optical which has the different directions with bi-directional optical flow. Aiming at anisotropic adjustment of intermediate flow, some works try to extract more information through pretrained models, such as depth information [2] and contextual information [16]. More complex motion assumptions are designed in some succeeding works, such as QVI [30] and EQVI [12]. However, it is still difficult for them to describe the actual movement under the absence of intermediate motion information.As shown in Fig. 1(d), the intermediate optical flows obtained based on this inaccurate assumption fail to describe the correct direction of football.

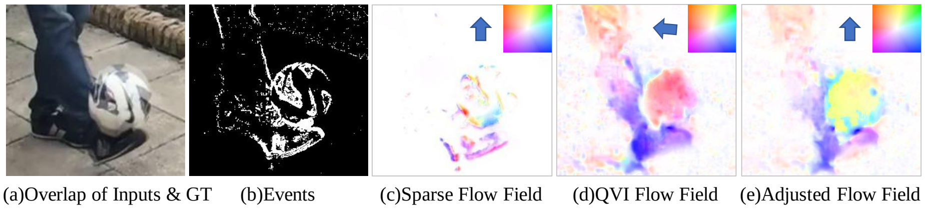

Fig. 1. The visualization of the generated optical flow. (c) demonstrates the sparse optical flow in Time Lens [27]. (d) demonstrates the final intermediate optical flow in QVI [30]. (e) demonstrates the final intermediate optical flows generated through the proposed model. These optical flows are used to warp the original frames to get initial intermediate frames. The arrows in these pictures denote the directions of football motion.

Event streams can record accurate motion information between two consecutive frames, which help to solve the absence of information in VFI model. Time Lens [27] uses event streams to compensate for motion information. Event streams generated through event camera asynchronously record the motion information for a period of time with the merits of high temporal resolution and low latency. Time Lens directly uses event to synthesize intermediate frame and estimate intermediate optical flow. The pre-warped frames through intermediate optical flow are aligned and refined by event-driven synthesis frames to obtain the final target frames. There are some limitations: although modeling motion is avoided in the process of obtaining intermediate optical flow, the warped result is limited by the sparse flow field estimated through event streams, as shown in Fig. 1(c), which lacks the dense features of the RGB images. The predicted sparse optical flow field is also easily affected by the noise of the event itself, which affects the final result. Besides, their model is not an end-to-end training model.

In this paper, we propose a video frame interpolation method $\mathbf{A^2OF}$ with event-driven **A**nisotropic **A**djustment of **O**ptical **F**low in an end-to-end training manner. Specifically, our model is based on bi-directional optical flows model, such as Super SloMo [7]. Firstly, we adopt and tailor IFNet in [6] in order to better model the motion from both event streams and frames. After getting the bi-directional optical flow, instead of linearly weighting them in both horizontal and vertical directions, anisotropic weights are learnt from events to blend and generate intermediate optical flow in the orthogonal directions. As shown in Fig. 1(e), the optical flows generated from our proposed models can demonstrate the correct direction of the football. Besides, we design an event-driven motion consistency loss based on the change of intensity to further improve the performance of our proposed model. We carry out a series of experiments to evaluate our proposed model on both synthetic and real event-frame datasets. Experimental results show that our model achieves the-state-of-art performance in all evaluated datasets.

Our main contributions can be summarized as:

1. To address the limitation in video frame interpolation model based on bi-directional optical flow, we design event-driven optical flow distribution masks generation module to provide anisotropic weights for the different directions of the optical flows.

2. In order to better compensate the motion information in the event to the VFI model, we design an event-driven motion consistency loss based on the change of intensity.

3. We design an end-to-end training video frame interpolation model, which outperforms the previous methods on both synthetic and real event-frames datasets.

2 Related Works

Video Frame Interpolation. The goal of VFI is to predict the intermediate frames between the input frames. Some methods [14, 13, 17] mainly focused on single-frame interpolation, which is commonly ineffective and inflexible. In order to interpolate several frames at any time between consecutive frames, Super-SloMo [7] designs a video frame interpolation model based on optical flows. According to the uniform motion assumption, they linearly aggregate bi-directional optical flows to obtain intermediate optical flow from the target frame to the original frame in an isotropic way. The final synthesis intermediate frames are warped and blended from the original frames. Their uniform motion assumption does not well with nonlinear motion in real scenes. Besides, they can not model the real motion which has the different directions with bi-directional optical flow. For anisotropic adjustment of optical flow, some extra information from the pretrained model is added into VFI model, such as depth information [2] or contextual information [16]. QVI [30] and EQVI [12] design more complex motion assumptions, however, which may still deviates from the way that objects move in the real scenes. The absence of motion information between original frames is the main cause of inaccurate motion estimation and low-quality frames interpolation. They may have poor frame interpolation ability when encountering complicated motion scenes due to inaccurate motion descriptions. We propose event-driven distribution masks generation module in the bi-directional model to obtain intermediate optical flows in an anisotropic way for the different directions of optical flow. With the help of event streams, we can get more accurate motion descriptions and high-quality interpolation results.

Event Camera. Through the neuromorphic sensor like dynamic vision sensors (DVS), event camera generates the high-dynamic-range event data under low power consumption [3]. Previous works demonstrate that the event-frame-based models show promising potential in visual tasks like image deblurring [18, 8, 11], high-dynamic-range image restoration [29, 4, 31], and VFI [28, 27, 5]. Some methods [18, 28, 27] avoid modeling the process of motion. The former two methods mix the feature of event and frame through neural networks to directly predict intermediate frames. The disturbance of threshold in event camera has a negative influence on the performance of their models. Time Lens [27] directly predicts intermediate optical flow through event streams, which limits the accuracy of optical flows estimation due to the absence of visual detail in RGB frames. TimeReplayer [5] proposes an unsupervised learning method to aid the video frame interpolation process. Our model is designed based on bi-directional optical flows model. We introduce event streams into our model for the aggregation of bi-directional optical flows in an anisotropic way. The performance of our VFI model is further improved.

3 Method

3.1 Revisiting Bi-directional Optical Flow VFI Model

Given two consecutive frames I_0 and I_1 in a video, we can obtain bi-directional optical flow $F_{0\to 1}$ and $F_{1\to 0}$. $F_{0\to 1}$ and $F_{1\to 0}$ are the optical flows from I_1 to I_0 and the optical flow from I_0 to I_1, respectively. $F_{\tau\to 0}$ and $F_{\tau\to 1}$ are the intermediate optical flows from the target frame to original frames I_0 and I_1, respectively. Both $F_{\tau\to 0}$ and $F_{\tau\to 1}$ can be represented by bi-directional optical flows $F_{0\to 1}$ and $F_{1\to 0}$ as shown in Eq. (1) and Eq. (2):

$$F_{\tau\to 0} = g(\tau, F_{1\to 0}, M_{\text{etra}}) \; \textit{or} \; g(\tau, F_{0\to 1}, M_{\text{etra}}), \tag{1}$$

$$F_{\tau\to 1} = g(1-\tau, F_{1\to 0}, M_{\text{etra}}) \; \textit{or} \; g(1-\tau, F_{0\to 1}, M_{\text{etra}}). \tag{2}$$

We can blend the optical flow obtained from $F_{1\to 0}$ and $F_{0\to 1}$ to get more accurate optical flows based on temporal consistency. The final intermediate optical flows can be represented as:

$$F_{\tau\to 0} = \text{Blend}_F(g(\tau, F_{1\to 0}, M_{\text{etra}}), g(\tau, F_{0\to 1}, M_{\text{etra}}); \theta_{0-\tau}), \tag{3}$$

$$F_{\tau\to 1} = \text{Blend}_F(g(1-\tau, F_{1\to 0}, M_{\text{etra}}), g(1-\tau, F_{0\to 1}, M_{\text{etra}}); \theta_{\tau-1}), \tag{4}$$

where $\text{Blend}_F(\cdot)$ denotes blending operation between initial intermediate optical flows. $\theta.$ denotes the parameters in the blending operations, such as linearly weight in temporal consistency [7] or parameters in convolution neural networks.

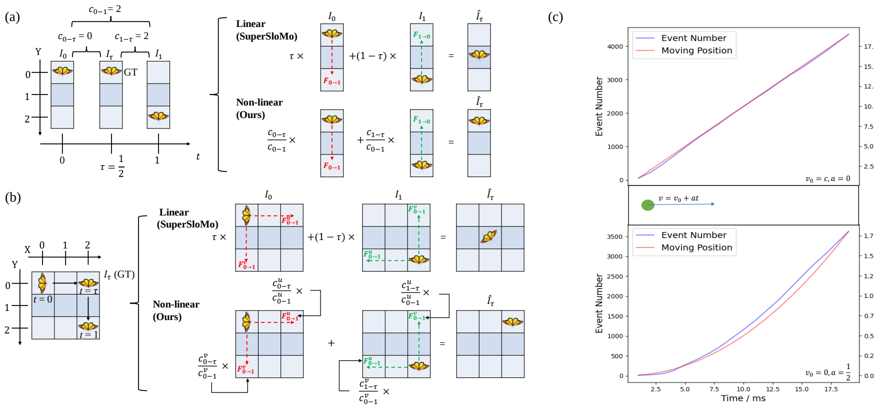

Fig. 2. (a-b) A toy example for the butterfly moving in $0-1$ period in one-dimensional and two-dimensional space, respectively. For two-dimensional movements, we use the number of events to generate optical flow masks in different directions. (c) Two enlightening examples on the relationship between the motion(the right axis) and event count(the left axis) in uniform rectilinear motion(the top graph) and uniformly accelerated rectilinear motion(the bottom graph). The unit of velocity v_0 and v are pixels per time step, and the unit of acceleration a is pixels per time step squared. The time step is 3ms.

According to the final intermediate optical flows, original frames I_0 and I_1 are warped through bilinear interpolation [32] and refined through convolution neural network to get pre-warped intermediate frames $\hat{I}_{0\to\tau}$ and $\hat{I}_{1\to\tau}$. The final intermediate frame $\hat{I}_\tau$ can be represented as the mixture of $\hat{I}_{0\to\tau}$, $\hat{I}_{1\to\tau}$ and extra pre-warped information such as visibility maps [7], depth information [2] and contextual information [16] as shown in Eq. (5):

$$\hat{I}_\tau = \text{Blend}_I(\hat{I}_{0\to\tau}, \hat{I}_{1\to\tau}, M_{\text{extra}}), \tag{5}$$

where Blend_I denotes blending operations between pre-warped intermediate frames.

A Toy Example. We take the VFI model in SuperSloMo [7] as an example model as shown in Fig. 2(a) and Fig. 2(b) for motion in one dimension and two dimensions, respectively. For the convenience of description, we take SuperSloMo as an example model. As shown in Fig. 2(a), a butterfly rests at location Y_0 in $0-\tau$ period and flies from location Y_0 to Y_2 in $\tau-1$ period. I_0 records a butterfly locates in Y_0, while I_1 records a butterfly locates in Y_2. For SuperSloMo model, which assumes the butterfly moves at a constant speed along a line, the synthesized frame I_τ shows the butterfly locates in Y_1 when $\tau = \frac{1}{2}$. However, the actual coordinates of the butterfly are at Y_0. The frame interpolation of the butterfly in two-dimensional motion is similar to that in one-dimensional as shown in Fig. 2(b). The butterfly is predicted at location $Y_{(1,1)}$, while the actual coordinates of the butterfly are at $Y_{(0,2)}$. The reason for this phenomenon is that they synthesize the intermediate optical flow in an isotropic way. All directions of optical flow are linearly aggregated with the same coefficients. Note that there is no motion in the vertical direction in $0 - \tau$ period in Fig. 2. So the optical flow in the vertical direction should not be taken into account when we calculate the intermediate optical flow $F_{\tau \to 0}$. However, due to the absence of intermediate motion information between two frames, it is difficult to predict the different coefficients in an anisotropic way. Thus, we introduce the event data that records the real motion information into our model and use the event to estimate the optical flow mask to better model the motion in the corresponding period.

3.2 Event-driven Optical Flow Mask

Let's come back to the one-dimensional condition in Fig.2(a), no motion is observed in $0 - \tau$, leading to no event streams generated. However, the event is generated in $\tau - 1$ due to the motion of the butterfly. If we regard every change as an event, the polarities of the event data $E_{0 \to \tau}$ of such the 3×1 one-dimensional map could be encoded as $[0,\ 0,\ 0]^T$ at the timestamp $t = \tau$ in $0 - \tau$ period, and that of $E_{\tau \to 1}$ is $[-1,\ 0,\ 1]^T$ at the timestamp $t = 1$ in $\tau - 1$ period. The total event data in $0 - 1$ period could be encoded as $[-1,\ 0,\ 1]^T$, which is the same as the event data $E_{\tau \to 1}$. In this simple example, we could find that the event data could encode the ground-truth movement, which could be possibly used to distribute the bi-directional optical flows. We carry out a simulation experiment that a pixel of ball moves in uniform rectilinear motion or uniformly accelerated rectilinear motion and draw the curve of moving position and the curve of event number along the time. As shown in Fig. 2(c), the trend of the moving position curve is basically consistent with the velocity-time curve, and the event count can describe the slowness.

Herein, we propose event-driven optical flow masks $\omega.$ that anisotropically determines the weights of the bi-directional optical flows. The optical flow $F_{0 \to 1}$ contributes nothing to the synthesis frame $\hat{I}_\tau$ due to no motion and no event occurrence in $0 - \tau$ period, while the optical flow $F_{1 \to 0}$ is more important to the synthesis frame I_τ due to more motion and event from the butterfly in $\tau - 1$ period. Thus, the distribution of $F_{0 \to 1}$ should be set as 0 while the distribution of $F_{1 \to 0}$ should be set as 1 to generate the accurate intermediate flows. We find the distribution of bi-directional optical flow can be calculated by the ratio between the number of event occurrences in target period and that in the total period. Based on the above analysis, Eq. (1) and Eq. (2) can be rewrited as follows:

$$F_{\tau \to 0} = \omega_{0-\tau} \cdot F_{1 \to 0} \text{ or } -\omega_{0-\tau} \cdot F_{0 \to 1}, \tag{6}$$

$$F_{\tau \to 1} = \omega_{1-\tau} \cdot F_{0 \to 1} \text{ or } -\omega_{1-\tau} \cdot F_{1 \to 0}, \tag{7}$$

where $\omega_{0-\tau}$ and $\omega_{1-\tau}$ denotes the optical flow mask of bi-directional optical flows in $0 - \tau$ period and $\tau - 1$. We can obtain these weight through event streams as shown in Eq. (8):

$$\omega_{0-\tau} = \frac{c_{0-\tau}}{c_{0-1}} \text{ and } \omega_{1-\tau} = \frac{c_{1-\tau}}{c_{0-1}}, \tag{8}$$

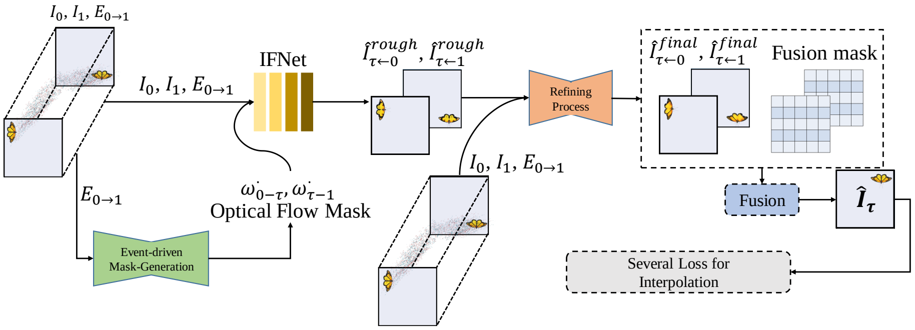
$I_0, I_1, E_{0\to1}$
$I_0, I_1, E_{0\to1}$
IFNet
$\hat{I}^{rough}_{\tau\leftarrow0}, \hat{I}^{rough}_{\tau\leftarrow1}$
Refining Process
$\hat{I}^{final}_{\tau\leftarrow0}, \hat{I}^{final}_{\tau\leftarrow1}$
Fusion mask
$E_{0\to1}$
$\omega^{\cdot}_{0-\tau}, \omega^{\cdot}_{\tau-1}$
Optical Flow Mask
$I_0, I_1, E_{0\to1}$
Fusion
$\hat{I}_\tau$
Event-driven Mask-Generation
Several Loss for Interpolation

Fig. 3. The pipeline of our proposed method **A^{2}OF**. The event is input to Event-driven Mask Generation network to get optical flow masks $\omega_{0-\tau}$ and $\omega_{\tau-1}$. We modify the structure of the IFNet [6] so that it can input event and use anisotropic flow mask as shown in supporting information. Two consecutive frames, corresponding event and optical flow mask are input to tailored IFNet to get intermediate optical flows and warp the original frames. The pre-warped frames $\hat{I}^{\text{rough}}_{\tau\to0}$ and $\hat{I}^{\text{rough}}_{\tau\to1}$ are fused to final intermediate frame $\hat{I}_\tau$under the supervision of a series of loss.

where c. denotes the number of event occurrences in target period. Note that large motion does not necessarily lead to large ratio: the optical flow between two frames represent the total amount of motion, while the ratio Eq. (8) indicates how the motion distributes over time.

In order to better analyze the motion in two-dimensional space and model the motion between two frames, we calculate different weights in the orthogonal directions as shown in Fig. 2(b). Specifically, as shown in Fig. 2(b) and Fig. 2(c), the original optical flow $F_{1\to0}$ can be divided into $F^u_{1\to0}$ in the horizontal direction and $F^v_{1\to0}$ in the vertical direction. $F_{0\to1}$ can be decomposed in the same way as $F^u_{0\to1}$ and $F^v_{0\to1}$. The subscripts of these decomposed optical flows indicate the optical flows are in which period, and the superscripts indicate the optical flows are in which direction. Note that we can still obtain intermediate optical flows in the different directions based on Eq. (6-8). Based on the time consistency, we can further combine initial intermediate optical flows $F^{\cdot}_{\tau\to0}$ or $F^{\cdot}_{\tau\to1}$ from $F^{\cdot}_{1\to0}$ and $F^{\cdot}_{0\to1}$, where $\cdot$ in the right corner denotes the direction of optical flow, such as horizontal or vertical direction. The final intermediate optical flows can be represented as:

$$F^u_{\tau\to0} = -(1-\tau)\cdot\omega^u_{0-\tau}\cdot F^u_{0\to1} + \tau\cdot\omega^u_{1-\tau}\cdot F^u_{1\to0}, \tag{9}$$

$$F^u_{\tau\to1} = (1-\tau)\cdot\omega^u_{1-\tau}\cdot F^u_{0\to1} - \tau\cdot\omega^u_{0-\tau}\cdot F^u_{1\to0}. \tag{10}$$

We can also obtain intermediate optical flow in the vertical direction as shown in Eq. (9-10).

Note that the event contains a certain amount of noise due to system noise and unfixed thresholds. Calculation to the weight with unprocessed events will result in inaccurate weight prediction. Thus, we propose a U-Net style convolution neural network named Event-driven Mask Generation Network to process the input event data. The architecture of Event-driven Mask Generation Network can be found in Supporting Information. As shown in Fig. 3, our event-driven mask generation network processes the input event $E_{0\rightarrow1}$ and outputs two channels feature map which encodes the number of events at each pixel in $0-\tau$ or $\tau-1$ period. The final weight $\omega^{\cdot}_{0\rightarrow\tau}$ and $\omega^{\cdot}_{1\rightarrow\tau}$ is calculated according to Eq. (8) with these two-channel feature map in the horizontal and vertical direction. The spatial size of $\omega^{\cdot}_{0\rightarrow\tau}$ and $\omega^{\cdot}_{1\rightarrow\tau}$ is with the same spatial size $H\times W$ as bi-directional optical flows. These event-driven optical flow masks are used to obtain the nonlinear intermediate flows $F^{\cdot}_{\tau\rightarrow0}$ and $F^{\cdot}_{\tau\rightarrow1}$ as shown in Eq. (9-10).

3.3 Pipeline for Our Event-driven Video Interpolation Model

In this section, we will specifically show the representation of events and the pipeline of our event-driven video interpolation model.

Event Representation. An original event e can be represented as a four-element tuple (x_e, y_e, p_e, t_e), where x_e and y_e denote spatial coordinates. p_e denotes the polarity of the event and t_e denotes the time of occurrence for the event. We should convert those event streams into 2-D frames to input the convolution neural network. Our event streams during the target period are represented as a four-channel frame as shown in [20]. The first and second channels encode the number of positive and negative polarities of events at each pixel, respectively. The third and fourth channels encode the timestamps of the latest triggered positive and negative events, respectively. The represented event data between two consecutive frames I_0 and I_1 is set as $E_{0\rightarrow1}$.

Event-driven Optical Flows Estimation. The image frames encode the details of the motion with the low temporal resolution, while the event records the motion information with high temporal resolution. A natural idea is to use these two complementary data together to predict the bi-directional optical flow. Time Lens [27] uses events to directly predict intermediate optical flow which lacks the details of the motion information from frames.

Thus, we take the frame data I_0, I_1, event $E_{0\rightarrow1}$ and anisotropic optical flow weights as the input to IFNet [6] to estimate the bi-directional optical flow and warp the original frames as shown in Fig. 3 and Supporting Information. We carefully tailor the IFNet and make it suitable for our task. Firstly, our tailored IFNet can process both event and image frames. In this way, the synthetic intermediate optical flow can correctly model the target motion due to the complementary of the two types of data with our anisotropic optical flow weights according to Eq. (9-10). Then, we deeper the IFNet to better process these two types of data. Our tailored IFNet has 4 blocks. Next, the optical flow mask from Event-driven Mask Generation network is input to each block in our tailored IFNet for blending and synthesizing intermediate optical flow. The output of IFNet is rough interpolated frames $\hat{I}^{\text{rough}}_{\tau\leftarrow1}$ and $\hat{I}^{\text{rough}}_{\tau\leftarrow0}$ which will be input to refinement network.

Refinement Network. The warped images $\hat{I}^{\text{rough}}_{\tau\leftarrow 0}$, $\hat{I}^{\text{rough}}_{\tau\leftarrow 1}$ could encode the most nonlinear motion but poorly around the stationary objects due to the absence of event data in the static regions. To make the intermediate frames better, a refining process is necessary like the previous work [7, 30, 27]. Except for the additional input event, the overall structure of our refinement network is similar to that in [7]. Our refinement network is a U-Net style network with 6 encoders and 5 decoders with a shortcut between encoder and decoder of the same spatial scale. The details of our refinement network are provided in supporting information.As shown in Fig. 3, we input the data I_0, I_1, $E_{0\rightarrow 1}$,$\hat{I}^{\text{rough}}_{\tau\leftarrow 0}$, $\hat{I}^{\text{rough}}_{\tau\leftarrow 1}$ and bi-directional optical flow $F_{0\rightarrow 1}$, $F_{1\rightarrow 0}$ into a sub-network. Herein, the output is two fusion maps $V_{\tau\leftarrow 0}$, $V_{\tau\leftarrow 1}$ and two refined frames $\hat{I}^{\text{final}}_{\tau\leftarrow 0}$ and $\hat{I}^{\text{final}}_{\tau\leftarrow 1}$.Thus, the final intermediate frames $\hat{I}_\tau$ is defined after the Fusion process of Fusion$(\cdot)$:

$$\begin{aligned}\hat{I}_\tau &= \text{Fusion}\left(I^{\text{final}}_{\tau\leftarrow 0}, I^{\text{final}}_{\tau\leftarrow 1}, V_{\tau\leftarrow 0}, V_{\tau\leftarrow 1}\right)\\ &= V_{\tau\leftarrow 0}\cdot I^{\text{final}}_{\tau\leftarrow 0} + V_{\tau\leftarrow 1}\cdot I^{\text{final}}_{\tau\leftarrow 1},\end{aligned} \tag{11}$$

where $V_{\tau\leftarrow 0}$ and $V_{\tau\leftarrow 1}$ are two visibility maps which encode whether the objects are occluded. The pixel-addition of two visibility map equals 1, following [7].

Loss Function. Our event-driven video frame interpolation model can be trained in an end-to-end manner under the combination of event-driven motion consistency loss $\mathcal{L}_{\text{MC}}$, the reconstruction loss $\mathcal{L}_{\text{rec}}$, the perceptual loss $\mathcal{L}_{\text{per}}$, the warped loss $\mathcal{L}_{\text{warp}}$ and the smoothness loss $\mathcal{L}_{\text{smooth}}$. Note that the reconstruction loss $\mathcal{L}_{\text{rec}}$, the perceptual loss $\mathcal{L}_{\text{per}}$,the warped loss $\mathcal{L}_{\text{warp}}$ and the smoothness loss $\mathcal{L}_{\text{smooth}}$ are similar to [7].

Event-driven motion consistency loss $\mathcal{L}_{\text{mc}}$ measures the gap between estimated event count map $\hat{E}^{\text{count}}_{\cdot}$ and real event count map $E^{\text{count}}_{\cdot}$. We take event count map $\hat{E}^{\text{count}}_{0\rightarrow\tau}$ as an example. $\hat{E}^{\text{count}}_{0\rightarrow\tau}$ is calculated with the difference between two frames I_{diff} which is represented as

$$I_{\text{diff}} = \log(\frac{I^{\text{final}}_{\tau\leftarrow 0}}{I_0}). \tag{12}$$

Due to unknown and inflexible thresholds in event camera, it is difficult to directly get event count map through Eq. (12). So, we binarize the event count to reflect whether the event appears or not in each pixel. Specifically, in the event count map, 1 denotes there is at least one event while 0 denotes there is no event. How to binarize the I_{diff} to obtain estimated event count map $\hat{E}^{\text{count}}_{0\rightarrow\tau}$? Firstly, we get $E^{\text{count}}_{0\rightarrow\tau}$ through the first and second channel in $E_{0\rightarrow\tau}$:

$$E^{\text{count}}_{0\rightarrow\tau} = \text{sgn}(E_{0\rightarrow\tau}[0:2]), \tag{13}$$

where sgn$(\cdot)$ denotes the sign function and $[0:2]$ denotes the first and second channel of a tensor. Then, we sum all elements in $E^{\text{count}}_{0\rightarrow\tau}$ along each channel to obtain binarized threshold t_{positive} and t_{negative}, respectively. These thresholds record the number of locations where the event has occurred. Next, we sort I_{diff}

in descending order and select the top t_{positive} value $N_{t_{\text{positive}}}$ and the bottom $N_{t_{\text{negative}}}$ value as binarized thresholds for I_{diff}.

$$\hat{E}^{\text{count}}_{0\to\tau} = [\text{sgn}(I_{\text{diff}} - N_{t_{\text{positive}}}), \text{sgn}(N_{t_{\text{negative}}} - I_{\text{diff}})], \tag{14}$$

where $[\cdot,\cdot]$ denotes concat function in tensor. $\text{sgn}(I_{\text{diff}} - N_{t_{\text{positive}}})$ represents where there may be positive events. We want $\hat{E}^{\text{count}}_{0\to\tau}$ and $E^{\text{count}}_{\cdot}$ to be exactly the same, which means that the motion information recorded by both is almost the same. The definition of $\mathcal{L}_{\text{mc}}$ is

$$\mathcal{L}_{\text{mc}} = \frac{1}{N}\|G(\hat{E}^{\text{count}}_{0\to\tau}) - G(E^{\text{count}}_{0\to\tau})\|_1 + \frac{1}{N}\|G(\hat{E}^{\text{count}}_{\tau\to1}) - G(E^{\text{count}}_{\tau\to1})\|_1, \tag{15}$$

where $G(\cdot)$ denotes Gaussian Blur Function. We smooth the $\hat{E}^{\text{count}}_{\cdot}$ with $G(\cdot)$ to alleviate the effect of noise in the event. The definition of the reconstruction loss, the perception loss, the warping loss and the smoothness loss here are the same as SuperSloMo [7] does.

The total loss $\mathcal{L}$ of our model is

$$\mathcal{L} = \lambda_{mc} \cdot \mathcal{L}_{\text{mc}} + \lambda_{rec} \cdot \mathcal{L}_{\text{rec}} + \lambda_{per} \cdot \mathcal{L}_{\text{per}} + \lambda_{warp} \cdot \mathcal{L}_{\text{warp}} + \lambda_{smooth} \cdot \mathcal{L}_{\text{smooth}}. \tag{16}$$

Note that all the loss weights in Eq. (16) is set empirically on the validation set. Specifically, $\lambda_{mc} = 1.0$, $\lambda_{rec} = 1.0$, $\lambda_{per} = 0.2$, $\lambda_{warp} = 0.8$ and $\lambda_{smooth} = 0.8$.

4 Experiments

4.1 Implementation Details

Our proposed model and all experiments are implemented in Pytorch [21]. We use adam optimizer with standard settings in [9]. For training, our model is trained end-to-end on 4 NVIDIA Tesla V100 GPUs for the total 500 epochs. The batch size of each training step is 28. The initial learning rate is 10^{-4} and is multiplied by 0.1 per 200 epochs. We calculate the peak-signal-to-noise ratio (PSNR), structural similarity(SSIM), and interpolation-error (IE) as the quantitative metric to evaluate the performance of our proposed method.

4.2 Datasets

We firstly evaluate our model on three common high-speed synthetic datasets with synthetic events: Adobe240 [26], GoPro [15] and Middlebury [1]. Besides, we also evaluate our model on the real frame-event dataset: High-Quality Frames (HQF) [25], High Speed Event-RGB(HS-ERGB) [27]. All the synthetic training sets are collected to train our model. Then, we compare with other previous state-of-the-art methods.

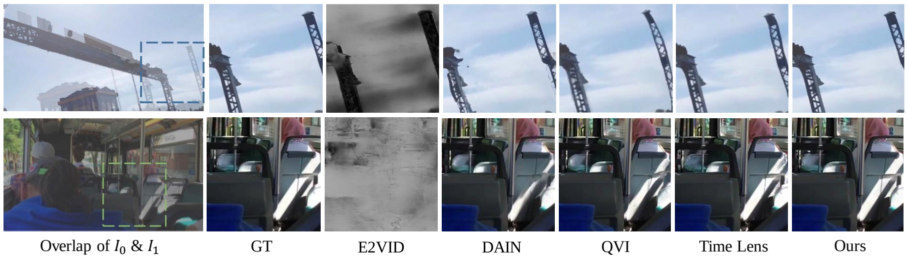

Fig. 4. Visual comparisons with different methods on the synthetic dataset Adobe240.

Synthetic datasets. Adobe240 consists of 112 different sequences for training and 8 sequences for testing. GoPro dataset consists of 22 different videos for training and 11 videos for testing. Captured by GoPro cameras, both of them own 240 fps and 1280×720 resolutions in all the sequences. We use ESIM [22] to generate event streams between two consecutive frames. We crop patches with spatial size 384×384 for training.

Real dataset. High-Quality Frames(HQF) is collected through the DAVIS240 event camera [24] which can generate both event streams and corresponding frames. It consists of 14 different frames sequences with the corresponding event streams. The resolutions are 240×180 in all sequences. We crop patches with spatial size 128×128 for training. HS-ERGB is collected through Gen4M 720p event camera and FLIP BackFly S RGB camera. This dataset is divided into far-away sequences and close planar scenes.

4.3 Comparisons with Previous Methods

Synthetic datasets. We compare our proposed video frame interpolation (VFI) model with previous VFI models. Previous methods can be classified into three categories: frame-based approach, event-based approach and frame-event-based approach. For frame-based approach, we compare our model with SuperSloMo [7], DAIN [2], SepConv [17], QVI [30]. For the event-based approach, we compare our model with event-based video reconstruction method E2VID [23]. For event-frame-based approach, we compare our model with Time Lens [27]. We make a fair comparison on three synthetic datasets Adobe240 [26], GoPro [15] and Middlebury [1]. The comparison results are as shown in Tab. 1. For training on Adobe240 and GoPro, we select 1 frame from every 8 frames in the original sequence, use the selected frames to form the input sequences, and use the remaining frames as the label for interpolated frames. Middlebury is only used for the test due to the sequence length. Besides, the event streams are generated through ESIM [22] between selected frames is also available to our model. For the input frames sequences from selected frames and events, the skipped frames are reconstructed through our proposed method and compared with the ground

Table 1. Quantitative comparison with previous methods on the synthetic datasets Adobe240, GoPro and Middlebury. Note that we directly evaluate the released model in each method without re-training or fine-tuning.

Adobe240		All frames in 7 skips			Middle frame in 7 skips		
Method	Input	PSNR	SSIM	IE	PSNR	SSIM	IE
E2VID [23]	Event	10.40	0.570	75.21	10.32	0.573	76.01
SepConv	RGB	32.31	0.930	7.59	31.07	0.912	8.78
DAIN	RGB	32.08	0.928	7.51	30.31	0.908	8.94
SuperSloMo	RGB	31.05	0.921	8.19	29.49	0.900	9.68
QVI	RGB	32.87	0.939	6.93	31.89	0.925	7.57
Time Lens	RGB+E	35.47	0.954	5.92	34.83	0.949	6.53
Ours(A^{2}OF)	RGB+E	**36.59**	**0.960**	**5.58**	**36.21**	**0.957**	**5.96**
GoPro		All frames in 7 skips			Middle frame in 7 skips		
Method	Input	PSNR	SSIM	IE	PSNR	SSIM	IE
E2VID	Event	9.74	0.549	79.49	9.88	0.569	80.08
SepConv	RGB	29.81	0.913	8.87	28.12	0.887	10.78
DAIN	RGB	30.92	0.901	8.60	28.82	0.863	10.71
SuperSloMo	RGB	29.54	0.880	9.36	27.63	0.840	11.47
QVI	RGB	31.39	0.931	7.09	29.84	0.911	8.57
Time Lens	RGB+E	34.81	0.959	5.19	34.45	0.951	5.42
Ours(A^{2}OF)	RGB+E	**36.61**	**0.971**	**4.23**	**35.95**	**0.967**	**4.62**
Middlebury (other)		All frames in 3 skips			Middle frame in 3 skips		
Method	Input	PSNR	SSIM	IE	PSNR	SSIM	IE
E2VID	Event	11.26	0.427	69.73	11.12	0.407	70.35
SepConv	RGB	25.51	0.824	6.74	25.12	0.811	7.06
DAIN	RGB	26.67	0.838	6.17	25.96	0.793	6.54
SuperSloMo	RGB	26.14	0.825	6.33	25.53	0.805	6.85
QVI	RGB	26.31	0.827	6.58	25.72	0.798	6.73
Time Lens	RGB+E	32.13	0.908	4.07	31.57	0.893	4.62
Ours(A^{2}OF)	RGB+E	**32.59**	**0.916**	**3.92**	**31.81**	**0.903**	**4.13**

truth skipped frames. The average performance of the whole 7 skipped frames and the center one are both calculated for fair comparison. Meanwhile, only 3 frames are skipped and used to calculate reconstruction metrics due to the sequence length limitation in Middlebury. The results are summarized in Tab. 1. Note that we directly evaluate the released model in each comparison method without re-training or fine-tuning.

From Tab. 1, our proposed method outperforms all the previous methods and achieves the-state-of-art performance on Adobe240, GoPro and Middlebury. The frame-event based approaches outperform the frame-only or event-only approaches. As shown in Fig. 4, due to better use of events with event-driven optical flow mask and event-driven motion consistency loss, our model achieves the best visual quality. Specifically, for the second row in Fig. 4, all objects are static while only the intensity changes. It's a common issue in warping based interpolation method, which is also mentioned by Time Lens. The results indicate improvement by event-driven motion consistency loss in our method.

Table 2. Quantitative comparison with previous methods on the real frame-event datasets HQF. Note that we directly evaluate the released model in each comparison method without re-training or fine-tuning.

HQF		3 skips		1 skip	
Method	Input	PSNR	SSIM	PSNR	SSIM
E2VID	Event	6.70	0.315	6.70	0.315
RRIN	RGB	26.11	0.778	29.76	0.874
BMBC	RGB	26.32	0.781	29.96	0.875
DAIN	RGB	26.10	0.782	29.82	0.875
SuperSloMo	RGB	25.54	0.761	28.76	0.861
Time Lens	RGB+E	30.57	0.900	32.49	0.927
Ours(A^{2}OF)	RGB+E	**31.85**	**0.932**	**33.94**	**0.945**
HS-ERGB (far)		7 skips		5 skip	
Method	Input	PSNR	SSIM	PSNR	SSIM
E2VID	Event	7.01	0.372	7.05	0.374
RRIN	RGB	23.73	0.703	25.26	0.738
BMBC	RGB	24.14	0.710	25.62	0.742
DAIN	RGB	27.13	0.748	27.92	0.780
SuperSloMo	RGB	24.16	0.692	25.66	0.727
Time Lens	RGB+E	32.31	0.869	33.13	0.877
Ours(A^{2}OF)	RGB+E	**33.15**	**0.883**	**33.64**	**0.891**
HS-ERGB (close)		7 skips		5 skip	
Method	Input	PSNR	SSIM	PSNR	SSIM
E2VID	Event	7.68	0.427	7.73	0.432
RRIN	RGB	27.46	0.800	28.69	0.813
BMBC	RGB	27.99	0.808	29.22	0.820
DAIN	RGB	28.50	0.801	29.03	0.807
SuperSloMo	RGB	27.27	0.775	28.35	0.788
Time Lens	RGB+E	31.68	0.835	32.19	0.839
Ours(A^{2}OF)	RGB+E	**32.55**	**0.852**	**33.21**	**0.865**

Real frame-event dataset. We also evaluate and compare our method with the state-of-the-art methods on the real dataset of HQF: E2VID [23], RRIN [10], BMBC [19], DAIN [2], SuperSloMo [7] and Time Lens [27]. The results are summarized in Tab. 2. We have two experiment settings for training on HQF: Firstly, we select 1 frame from every 4 frames in the original sequences which is viewed as '3-skip' as shown in Tab. 2. Second, we select 1 frame from every 2 frames in the original sequences which is viewed as '1-skip' as shown in Tab. 2. For HS-ERGB, we select 1 frame from every 8 frames or select 1 frame from evey 6 frames. Note that our model is finetuned with real frames-event datasets based on the model trained on synthetic datasets.

Similar results as the experiments on synthetic datasets could be indicted by quantitative comparison as shown in Tab. 2. As shown in Fig. 5, only our method could reconstruct the legs of the moving horse in the second row. Due to the absence of visual details from frames in E2VID, synthesized frames through the only event has a large the gap with real intermediate frames. DAIN and QVI fail to synthesize high-quality interpolated frames in the regions with complex

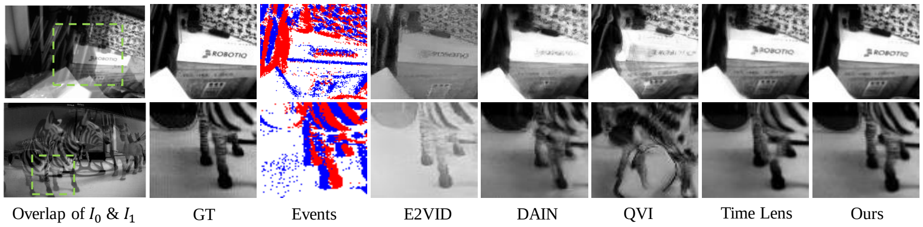

Fig. 5. Visual comparisons with different methods on HQF with real events.

Table 3. Ablation studies. EDOF denotes our proposed event-driven optical flow mask. MCL denotes our proposed event-driven Motion Consistency Loss.

EDOF	MCL	PSNR	SSIM	IE
×	✓	33.82	0.952	7.13
✓	×	35.27	0.963	5.75
✓	✓	36.61	0.971	4.23

motions such as the letters in the first row and the legs in the second row. Their methods cannot model the complicated motion in the real scenes very well.

4.4 Ablation Studies

To study the contribution of each module in our proposed model, we perform the ablation study on the GoPro dataset as shown in Tab. 3. Firstly, to evaluate the influence of Event-driven Optical Flow mask, we replace the mask with linear weights in SuperSloMo [7]. Experimental results show that our proposed event-driven optical flow mask has a positive effect in VFI model with the improvement of PSNR, SSIM and IE. As for our proposed event-driven motion consistency loss, we train a model without the supervision of this loss. This loss can be used to constrain the motion information coming from the change of intensity and event streams to be similar. Results show that this loss can further improve the performance of our proposed model.

5 Conclusion

In this paper, we propose a video frame interpolation method with event-driven anisotropic flow adjustment in an end-to-end training strategy. Besides, we design an event-driven motion consistency loss based on the change of intensity to constrain the gap between the estimated motion information and that from event streams. Instead of proposing complex motion assumptions like previous work, leading to isotropic intermediate flow generation or anisotropic adjustment through learned higher-order motion information, we use events to generate event-driven optical masks for the different directions, assigning the weight of bi-directional optical in intermediate optical flow in an anisotropic way.The proposed event-driven motion consistency loss further improves our method. The experiment results show that our model performs better than previous methods and achieves the state-of-the-art performance.

References

1. Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Black, M.J., Szeliski, R.: A database and evaluation methodology for optical flow. Int. J. Comput. Vis. **92**(1), 1–31 (2011)
2. Bao, W., Lai, W., Ma, C., Zhang, X., Gao, Z., Yang, M.: Depth-aware video frame interpolation. In: 2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 3703–3712 (2019)
3. Gallego, G., Delbruck, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A., Conradt, J., Daniilidis, K., et al.: Event-based vision: A survey. IEEE Trans. Pattern Anal. Mach. Intell. (2019)
4. Han, J., Zhou, C., Duan, P., Tang, Y., Xu, C., Xu, C., Huang, T., Shi, B.: Neuromorphic camera guided high dynamic range imaging. In: 2020 IEEE Conference on Computer Vision and Pattern Recognition,CVPR 2020, Seattle, WA, USA, June 13-19, 2020. pp. 1730–1739 (2020)
5. He, W., You, K., Qiao, Z., Jia, X., Zhang, Z., Wang, W., Lu, H., Wang, Y., Liao, J.: TimeReplayer: Unlocking the Potential of Event Cameras for Video Interpolation. In: CVPR (2022)
6. Huang, Z., Zhang, T., Heng, W., Shi, B., Zhou, S.: RIFE: real-time intermediate flow estimation for video frame interpolation. CoRR **abs/2011.06294** (2020)
7. Jiang, H., Sun, D., Jampani, V., Yang, M., Learned-Miller, E.G., Kautz, J.: Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 9000–9008 (2018)
8. Jiang, Z., Zhang, Y., Zou, D., Ren, J., Lv, J., Liu, Y.: Learning event-based motion deblurring. In: 2020 IEEE Conference on Computer Vision and Pattern Recognition,CVPR 2020, Seattle, WA, USA, June 13-19, 2020. pp. 3320–3329 (2020)
9. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2015)
10. Li, H., Yuan, Y., Wang, Q.: Video frame interpolation via residue refinement. In: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020. pp. 2613–2617 (2020)
11. Lin, S., Zhang, J., Pan, J., Jiang, Z., Zou, D., Wang, Y., Chen, J., Ren, J.: Learning event-driven video deblurring and interpolation. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part VIII. pp. 16155–16164 (2020)
12. Liu, Y., Xie, L., Li, S., Sun, W., Qiao, Y., Dong, C.: Enhanced quadratic video interpolation. In: Bartoli, A., Fusiello, A. (eds.) Computer Vision - ECCV 2020 Workshops - Glasgow, UK, August 23-28, 2020, Proceedings, Part IV. pp. 41–56 (2020)
13. Liu, Z., Yeh, R.A., Tang, X., Liu, Y., Agarwala, A.: Video frame synthesis using deep voxel flow. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. pp. 4463–4471 (2017)
14. Long, G., Kneip, L., Alvarez, J.M., Li, H., Zhang, X., Yu, Q.: Learning image matching by simply watching video. In: Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI. pp. 434–450 (2016)
15. Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 257–265 (2017)

16. Niklaus, S., Liu, F.: Context-aware synthesis for video frame interpolation. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 1701–1710 (2018)
17. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive separable convolution. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017
18. Pan, L., Scheerlinck, C., Yu, X., Hartley, R., Liu, M., Dai, Y.: Bringing a blurry frame alive at high frame-rate with an event camera. In: 2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 6820–6829. Computer Vision Foundation / IEEE (2019)
19. Park, J., Ko, K., Lee, C., Kim, C.S.: Bmbc: Bilateral motion estimation with bilateral cost volume for video interpolation. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV
20. Park, P.K.J., Cho, B.H., Park, J.M., Lee, K., Kim, H.Y., Kang, H.A., Lee, H.G., Woo, J., Roh, Y., Lee, W.J., Shin, C., Wang, Q., Ryu, H.: Performance improvement of deep learning based gesture recognition using spatiotemporal demosaicing technique. In: 2016 IEEE International Conference on Image Processing, ICIP 2016, Phoenix, AZ, USA, September 25-28, 2016. pp. 1624–1628 (2016)
21. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library pp. 8024–8035 (2019)
22. Rebecq, H., Gehrig, D., Scaramuzza, D.: Esim: an open event camera simulator. In: 2nd Annual Conference on Robot Learning, CoRL 2018, Zürich, Switzerland, 29-31 October 2018, Proceedings. pp. 969–982 (2018)
23. Rebecq, H., Ranftl, R., Koltun, V., Scaramuzza, D.: High speed and high dynamic range video with an event camera. IEEE Trans. Pattern Anal. Mach. Intell. **43**(6), 1964–1980 (2021)
24. Stoffregen, T., Scheerlinck, C., Scaramuzza, D., Drummond, T., Barnes, N., Kleeman, L., Mahony, R.: Reducing the sim-to-real gap for event cameras. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16. pp. 534–549 (2020)
25. Stoffregen, T., Scheerlinck, C., Scaramuzza, D., Drummond, T., Barnes, N., Kleeman, L., Mahony, R.E.: Reducing the sim-to-real gap for event cameras. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVII. pp. 534–549 (2020)
26. Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., Wang, O.: Deep video deblurring for hand-held cameras. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 237–246 (2017)
27. Tulyakov, S., Gehrig, D., Georgoulis, S., Erbach, J., Gehrig, M., Li, Y., Scaramuzza, D.: Time lens: Event-based video frame interpolation. In: 2021 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021. pp. 16155–16164 (2021)
28. Wang, B., He, J., Yu, L., Xia, G., Yang, W.: Event enhanced high-quality image recovery. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIII. pp. 155–171 (2020)
29. Wang, L., Ho, Y.S., Yoon, K.J., et al.: Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In: 2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 10081–10090 (2019)

30. Xu, X., Si-Yao, L., Sun, W., Yin, Q., Yang, M.: Quadratic video interpolation. In: Wallach, H.M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E.B., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. pp. 1645–1654 (2019)
31. Zhang, S., Zhang, Y., Jiang, Z., Zou, D., Ren, J., Zhou, B.: Learning to see in the dark with events. In: Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16. pp. 666–682 (2020)
32. Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.A.: View synthesis by appearance flow. In: Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV. pp. 286–301 (2016)