Learning to Reconstruct High Speed and High Dynamic Range Videos from Events

Yunhao Zou[1] Yinqiang Zheng[2] Tsuyoshi Takatani[3] Ying Fu[1*]
[1]Beijing Institute of Technology [2]The University of Tokyo [3]National Institute of Informatics

Abstract

Event cameras are novel sensors that capture the dynamics of a scene asynchronously. Such cameras record event streams with much shorter response latency than images captured by conventional cameras, and are also highly sensitive to intensity change, which is brought by the triggering mechanism of events. On the basis of these two features, previous works attempt to reconstruct high speed and high dynamic range (HDR) videos from events. However, these works either suffer from unrealistic artifacts, or cannot provide sufficiently high frame rate. In this paper, we present a convolutional recurrent neural network which takes a sequence of neighboring events to reconstruct high speed HDR videos, and temporal consistency is well considered to facilitate the training process. In addition, we setup a prototype optical system to collect a real-world dataset with paired high speed HDR videos and event streams, which will be made publicly accessible for future researches in this field. Experimental results on both simulated and real scenes verify that our method can generate high speed HDR videos with high quality, and outperform the state-of-the-art reconstruction methods.

1. Introduction

Compared with ordinary cameras that capture scene intensities at a fixed frame rate, event cameras work in a quite different way of detecting pixel-wise intensity changes. One unique feature of event camera is that an event is triggered whenever the intensity change of a pixel reaches certain contrast threshold, thus events are recorded asynchronously. Event cameras have quite a few advantages over conventional frame-based ones, *e.g.*, low latency, low power, high temporal resolution, and high dynamic range (HDR) [6]. Thanks to these features, event cameras are beneficial for different vision tasks including real-time object tracking [24], high speed motion estimation [17], optical flow estimation [7, 18], depth map prediction [42], egomotion estimation [41], and on-board robotics [37].

Since event cameras record intensity changes without any absolute intensity, they cannot be directly used for existing image-based vision algorithms. Very recently, researches have been conducted to reconstruct high speed and HDR intensity images/videos [22, 32, 33] from events, which open up new usage of event cameras. In spite of that, the reconstructed high speed and HDR videos are still unsatisfactory in terms of visual quality. This motivates us to explore a better way of video reconstruction from events.

One possible reason for the insufficiency of reconstruction quality might be the shortage of high-quality learning data. For example, existing researches [22, 32, 33] unanimously try to simulate events by using a simulator, such as ESIM [21]. However, high-speed HDR videos appropriate for data simulation are extremely rare, and the movement of a virtual event camera might not be realistic. Furthermore, although substantial efforts [28] have been made to improve the simulator, it still remains unknown how the simulated events comply with real events recorded by event cameras, especially when considering that complex factors like noise and data transfer bandwidth limitations are presented in a real event camera. This triggers us to develop proper imaging devices to capture paired high speed HDR videos and events. Besides, these works [22, 32, 33] either ignore temporal constraints, or use a suboptimal flow warping loss. These also affect the video reconstruction quality.

In this paper, we make full use of the high frame rate and HDR features of event streams, to reconstruct high speed HDR videos from events. Specifically, we propose a convolutional recurrent neural network for high speed HDR video reconstruction. In order to minimize the temporal discontinuity along frame sequences, a temporal consistency constraint is designed based on the physical formation of events. In addition, we collect paired high speed HDR and event data in real scenes through our elaborately designed imaging prototype, which will be accessible publicly to facilitate other researchers in this field. Experimental results show that our method can achieve state-of-the-art reconstruction performance, and introducing paired real-world data in the training stage further help the model to handle

*Corresponding Author: fuying@bit.edu.cn

real HDR scenes.

The main contributions of our work can be summarized as follows

1. We propose a convolutional recurrent neural network for the reconstruction of high speed HDR videos from events, along with a temporal consistency to constrain the temporal discontinuity.

2. We design a special imaging system to collect the high speed and high bit-depth HDR videos with the corresponding event streams, which stands out as a novel alternative for data preparation beyond numerical simulation.

3. We collect a high-quality real dataset which contains paired high speed HDR videos and event streams of outdoor dynamic scenes, and verify the effectiveness of our method on this real-world dataset.

2. Related Work

In this section, we introduce the most relevant works to the proposed method. First, we review some intensity image and video reconstruction methods. Then, HDR applications of events are introduced.

Intensity Images and Videos Reconstruction. Though events have many advantages over traditional images including high temporal resolution and high dynamic range, they can be hardly utilized in practical applications due to the asynchrony and stream feature. Thus, reconstructing intensity images and videos from events becomes an active topic.

During early explorations of events-to-image reconstruction, Cook *et al*. [1] presented a network to interpret events by means of recurrently interconnected areas, to reconstruct light intensity as well as optical flow. Kim *et al*. [13] built a high-resolution mosaic of a scene based on probabilistic filtering. Recent years, with the rapid development of computing power, deep learning has been introduced to the area of reconstructing images and videos from events, and have achieved unprecedented results. Some approaches used the convenient event simulator ESIM [21] to generate events, and regarded the input images or the APS images as ground truth. For example, Rebecq *et al*. [22] proposed a convolutional recurrent neural network with flow warping loss to reconstruct videos from events. Wang *et al*. [33] directly used APS images as ground truth with some blurry ones removed, to form the training data, and they proposed a generative adversarial network to convert stacked events to images. Wang *et al*. also presented an adversarial learning based method [34] called EventSR to reconstruct, restore and super resolve intensity images together in an unsupervised manner. Mohammad *et al*. [10] modeled super-resolving event data to higher-resolution intensity images in an end-to-end network. These methods can effectively build images from events, but the visual quality is limited, especially on HDR scenes.

Since APS images are blurry due to the exposure mechanism of frame-based cameras, they are not good estimations for sharp video frames. Besides, simulating event streams from a single image along camera trajectory cannot well represent real object and camera motion. To address this problem, some recent works used event streams to help deblur APS frames based on the double integral relationship between events and APS frames [11, 19, 32]. Pan *et al*. [19] proposed the Event-based Double Integral (EDI) model to reconstruct sharp video from a single blurry APS frame and the corresponding event. Wang *et al*. [32] designed a deep unfolding network with sparse learning to reconstruct both high quality and high resolution images. Jiang *et al*. [11] also unfolded the optimization process of APS deblurring with an recurrent neural network. Restricted by APS frames, these methods either cannot provide high speed video reconstruction [11] considering that APS has limited frame rate, or are not able to handle HDR scenes [19, 32] due to the low dynamic range (LDR) of APS. Although the blurry APS frame can be helpful in sharp frame reconstruction, newly released event cameras do not support APS [29], and it is critical to investigate how to reconstruct high speed HDR images from event streams only.

HDR Reconstruction from Events. Since event camera captures intensity changes in logarithmic scale, it is sensitive to extreme dark scenes and does not suffer from overexposure in bright conditions. In other words, event streams have strong adaptation on HDR scenes. Kim *et al*. [13] popularized the idea of HDR image reconstruction from event streams, they built super-resolution accurate and high dynamic range mosaic of a scene under rotational camera motion assumption. Later, many learning based event-to-image reconstruction methods [10, 22, 33] trained their model on ordinary images, and directly generalized their models to HDR scenes in the testing stage. Although from their reconstruction results, details of the dark and bright regions can be recognized, the visual perception of these reconstructions does not match real scenes. This phenomenon is caused by their experimental settings that only LDR training samples are used to simulate events. To tackle this problem, Han *et al*. [8] proposed a hybrid HDR imaging system and fused a LDR image with an intensity map which was obtained from the corresponding event streams, to build an HDR image. Their results are more visually natural, but can be hardly used for high speed HDR video reconstruction restricted by the hardware prototype.

Different from these methods, our work are designed for reconstructing high speed HDR videos on real-world dataset. We design a convolutional recurrent neural network which can effectively exploit the features of adjacent event

frames and reconstruct high quality HDR videos with high frame rates. Besides, we setup a prototype optical system to collect paired high speed HDR videos and event streams, which will be publicly accessible for future researches.

3. Method

In this section, we first formulate the problem and illustrate our motivation. Then, the strategy to represent events is described. Next, we present the network architecture of our model. Finally, the implementation details are provided.

3.1. Formulation and Motivation

Event cameras capture a stream of asynchronous spikes, and an event is triggered if the logarithm of the brightness change at certain pixel reaches a given contrast threshold. Thus, the event information is recorded in the form of 4-tuples

$$e = (x, y, q, t), \tag{1}$$

with pixel locations (x, y), polarity $q \in \{\pm 1\}$ and the precise timestamp t. Let S and $I_{xy}(t)$ denote the contrast threshold and intensity at time t and location (x, y), the event generation process can be expressed as

$$log(I_{xy}(t)) - \log(I_{xy}(t - \Delta t)) = qS, \tag{2}$$

where $t-\Delta t$ is the timestamp of last event at location (x, y). The captured data of an event camera is a set of continuous event streams $\{e_i\}$.

HDR videos capture the irradiance of a scene sequentially. A widely used HDR generation method [3] firstly takes multiple LDR images under different exposures, then recovers the irradiance by dividing the image with the corresponding exposure time, and finally merges these irradiance maps to reproduce HDR images. In this way, ordinary HDR video production systems [12, 31] take bursts of images of a scene with alternating short and long exposures to produce an HDR frame. However, due to the limitation of exposure time, these prototypes cannot be well used for high frame rate video production.

Considering that event cameras measure intensity changes in the logarithmic scale, thus they can well handle HDR scenes. Previous works [10, 13, 22, 33] already proved that events had great potential for HDR images and videos reconstruction. However, these methods can only handle HDR scenes instead of producing standard HDR images, as they are interested in differential information rather than the absolute scene intensity. Therefore, these methods would inevitably suffer from artifacts, which make the reconstruction results less realistic. To avoid this problem, we propose an approach which is specially designed for high speed HDR video reconstruction. Assuming that we have an event stream $\mathcal{E} = \{\mathbf{E}_i\}$ and the corresponding ground truth HDR frames $\mathcal{H} = \{\mathbf{H}_i\}$, our goal is to reconstruct high speed HDR $\mathcal{H}$ from $\mathcal{E}$ through an end-to-end network.

3.2. Event Stacking

In order to utilize convolutional neural networks to deal with event data, we need to embed event streams into voxel grids (also event frames), which contain spatial features akin to ordinary image frames. Previous works [22, 28, 33] proposed several ways to integrate event streams into tensors, including stacking based on time, stacking based on the number of events, and stacking between frames. As for video reconstruction, it is natural to stack events between two ground truth frames to keep a consistent timestamp of the reconstructed frames. Specifically, given a stream of events which lays between two consecutive ground truth frames, and the spanning time is denoted as ΔT. We first divide the event stream into B temporal bins, and each bin of events is merged to form a grayscale image. Thus, all the events between the duration of ΔT are represented by a $U \times V \times B$ voxel grid, where $U \times V$ denotes the spatial size of the event sensor. The simple merging operation of events may lose some scene information, but would definitely help convolutional neural networks to process event data. In the experiments, we merge positive and negative events separately to remain more details. Therefore, we obtain a voxel grid $\mathbf{E}$ with $2B$ channels for the events between ΔT.

3.3. Network Architecture

We design a convolutional recurrent neural network to reconstruct HDR videos from a stream of events, and the overview is illustrated in Fig. 1. Our model takes $T = 2N + 1$ consecutive event voxel grids $\{\mathbf{E}_{t-N}, \ldots, \mathbf{E}_{t+N}\}$ to build the HDR frame $\mathbf{H}_t$ at timestamp t. In our network, features of different event frames are fused, and passed along with the event sequence through an internal recurrent memory state. We also present a novel consistency loss to keep continuity along temporal dimension. Our network mainly includes a shared feature extractor, a deformable convolution based alignment module, a convolutional recurrent fusion and reconstruction module, and a pretrained consistency loss module.

Shared feature extractor. In this module, event frames are downsampled to a low spatial resolution feature space. It is inspired by previous video enhancing methods [35, 38, 39], where the effectiveness of using a shared feature extractor/encoder to transform the input data to feature maps before alignment has been proved. Actually, the shared feature extractor can encode consecutive frames into the same feature space, which is beneficial for the following alignment module. We simply employ several strided convolution layers to encode the frames into feature space, and obtain $2N + 1$ output feature maps $\{\mathbf{F}_{t-N}, \ldots, \mathbf{F}_{t+N}\}$.

Deformable convolution based alignment. Previous event-to-image reconstruction methods utilized optical flow to align different frames [10], or employed a flow warp-

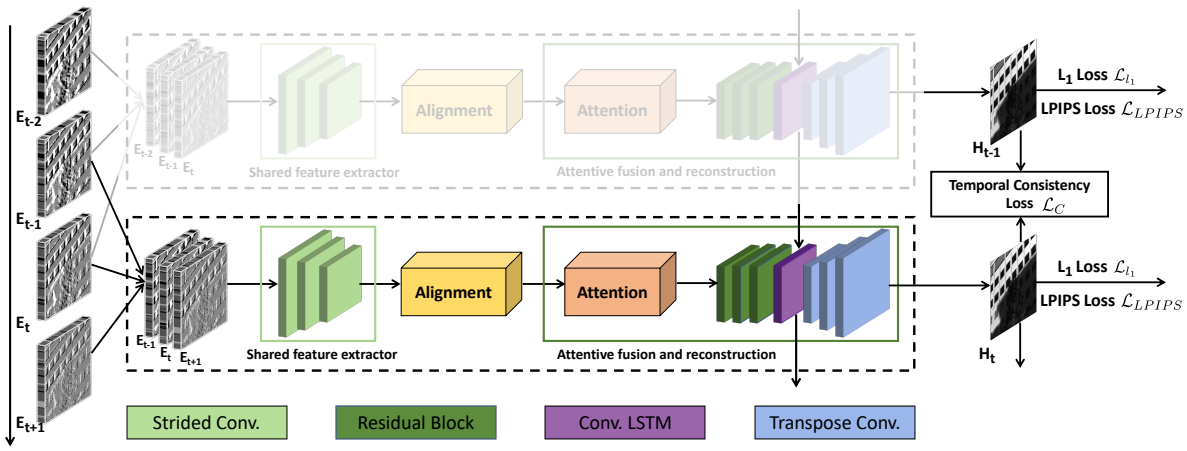


Figure 1. The overview of our recurrent convolutional neural network for HDR video reconstruction from events.

ing loss to alleviate the temporal discontinuity [22]. However, obtaining an accurate flow is difficult, and erroneous flow estimation would cause motion artifacts [30]. We follow [35] to employ pyramidal deformable convolutions [2] for feature alignment, which learn offsets of normal convolution kernels to obtain aligned features.

In the alignment module, our goal is to align features of different event frames $\mathbf{F}_{t+i}$ to the feature of the central frame $\mathbf{F}_t$. Assuming that a convolution kernel has K locations. Take a common 3×3 kernel for example, we have $K = 9$ and the regular grid $\mathcal{R} = \{(-1,-1),(-1,0),\ldots,(1,0),(1,1)\}$, which denotes the locations of an ordinary convolution operation. For each location $\mathbf{p}_0$ on the output feature map, the aligned feature can be expressed as

$$\mathbf{F}^a_{t+i}(\mathbf{p}_0) = \sum_{\mathbf{p}_n \in \mathcal{R}} \mathbf{w}(\mathbf{p}_0) \cdot \mathbf{F}_{t+i}(\mathbf{p}_0 + \mathbf{p}_n + \Delta\mathbf{p}_n), \quad (3)$$

where $\mathbf{w}$ is the weights for each location in $\mathcal{R}$, and $\mathbf{p}_n$ and $\Delta\mathbf{p}_n$ denote the pre-specified offset and learnable offset of n-th location in deformable convolutions. Eq. (3) illustrates the operation of a simple deformable convolutional layers that the convolutions are sampled on an extra offset $\Delta\mathbf{p}_n$, comparing to ordinary convolutional layers.

To predict the learnable offset $\Delta\mathbf{P} = \{\Delta\mathbf{p}_n\}_{\mathbf{p}_n \in \mathcal{R}}$ for the $(t+i)$-th event feature, the feature of the $(t+i)$-th frame $\mathbf{F}_{t+i}$ and the central frame $\mathbf{F}_t$ are sent to the offset predicting operation f, and can be expressed as

$$\Delta\mathbf{P}_{t+i} = f(\mathbf{F}_{t+i}, \mathbf{F}_t). \quad (4)$$

we employ pyramidal processing and cascading refinement to enlarge the receptive field of the offsets and align larger movement like [35, 39]. Specifically, assuming that the pyramidal architecture consists of L levels, and the feature of the l-th layer $\mathbf{F}^l_{t+i}$ is downsampled through strided convolutions with a factor of 2 on the $(l-1)$-th feature $\mathbf{F}^{l-1}_{t+i}$.

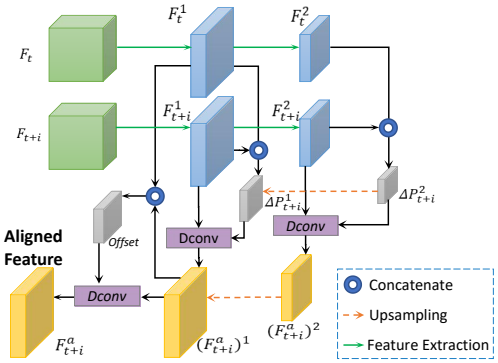


Figure 2. The architecture of the deformable convolution based alignment. Here, we only illustrate two input frame features and two pyramidal layers as an example.

After obtaining all of the L features, we calculate the offset for the l-th layer from the upsampled $(l+1)$-th offsets and the l-th pyramidal feature, as shown in Fig. 2. This process can be interpreted as

$$\Delta\mathbf{P}^l_{t+i} = f(\mathbf{F}_{t+i}, \mathbf{F}_t, \mathcal{U}(\Delta\mathbf{P}^{l+1}_{t+i})), \quad (5)$$

where $\mathcal{U}$ denotes bilinear upsampling operation. Thus, the aligned feature of the l-th level can be expressed as

$$(\mathbf{F}^a_{t+i})^l = g(\mathrm{DConv}(\mathbf{F}^l_{t+i}, \Delta\mathbf{P}^l_{t+i}), \mathcal{U}((\mathbf{F}^a_{t+i})^{l+1})). \quad (6)$$

In Eq. (6), g is convolutional layers to generate aligned features, and DConv is the deformable convolution described in Eq. (3). In this way, we obtain the aligned feature $(F^a_{t+i})^1$ for the 1-st pyramidal layer. We further use the feature F^1_t of the reference frame to generate the final aligned feature F^a_{t+i} from $(F^a_{t+i})^1$. For each of the T frames, we could obtain the corresponding aligned feature from Fig. 2. The number of pyramid levels L is set to 3 in the experiments.

Attentive fusion and reconstruction. Through the deformable convolution based alignment module, we obtain

T aligned features, and they are stacked together to form a $T \times C \times H \times W$ feature, where C, H and W is the size of the aligned features for each event frame. Here, we employ attention mechanism, which is widely used in high level vision tasks such as semantic segmentation [5], to further trace both spatial and temporal dependency. Considering that attention modules are computation consuming, we separately employ height, width and temporal/channel attention blocks which exploit the feature correlation along three dimensions. As shown in Fig. 3, three attention blocks work independently, and their features are summed up to produce the fused feature map.

Then, the obtained feature map is passed through a recurrent residual network to reconstruct HDR video frames. Specifically, several residual blocks are used to extract the hidden features from attention module. Then, we utilize a ConvLSTM [26] module to pass the reconstruction feature along with the temporal frame sequence. The ConvLSTM can be described as

$$\begin{aligned} \mathbf{i}_t &= \sigma(\mathbf{W}_i^X * \mathbf{X}_t + \mathbf{W}_i^Y * \mathbf{Y}_{t-1}), \\ \mathbf{f}_t &= \sigma(\mathbf{W}_f^X * \mathbf{X}_t + \mathbf{W}_f^Y * \mathbf{Y}_{t-1}), \\ \mathbf{o}_t &= \sigma(\mathbf{W}_o^X * \mathbf{X}_t + \mathbf{W}_o^Y * \mathbf{Y}_{t-1}), \\ \mathbf{c}_t &= \mathbf{f}_t \circ \mathbf{c}_{t-1} \\ &\quad + \mathbf{i}_t \circ tanh(\mathbf{W}_c^X * \mathbf{X}_t + \mathbf{W}_c^Y * \mathbf{Y}_{t-1}), \\ \mathbf{Y}_t &= \mathbf{o}_t \circ tanh(\mathbf{c}_t), \end{aligned} \tag{7}$$

where $\mathbf{i}_t$, $\mathbf{f}_t$, $\mathbf{o}_t$, $\mathbf{c}_t$ are represented as input gate, forget gate, output gate and memory cell of t-th moment, respectively, and $\mathbf{W}$ is the corresponding learnable weights. Besides, $\mathbf{X}_t$, $\mathbf{Y}_t$ are the input feature and hidden state of t-th moment, $\sigma(\cdot)$ and $tanh(\cdot)$ are Sigmoid and Tanh activation functions, and $*$ and $\circ$ are denoted as convolutional operator and Hadamard product.

In this way, the internal memory state $\mathbf{c}_t$ of recurrent neural network is able to remember information of features from successive sequences to help the reconstruction of the current frame and alleviate temporal discontinuity, when reconstructing HDR video frames.

Temporal consistency loss. Previous works [10, 22] employed flow warping error [16] for temporal consistency loss. However, their results can be suboptimal due to the lack of accurate flow estimation. To avoid using optical flow, we propose a novel strategy for temporal consistency loss. Several works [19, 32] have analyzed that the intensity change of two successive sharp frames can be represented by the integral of events between these two frames. Thus, given two consecutive ground truth frames $\hat{\mathbf{H}}_{t-1}$, $\hat{\mathbf{H}}_t$, we could inversely obtain the event frame E_t

$$\mathbf{E}_t = \mathcal{C}(\hat{\mathbf{H}}_{t-1}, \hat{\mathbf{H}}_t), \tag{8}$$

where $\mathcal{C}$ denotes the integral relationship between frames and events. Apparently, we could simply regard $\mathcal{C}$ as a process similar to the ESIM simulator [21], but it would be more accurate to derive it from training data. We use a UNet-like convolutional neural network [23] to learn the mapping $\mathcal{C}$ from frames to events. The network is pretrained before training our main model, and it serves as a temporal consistency loss module which helps the successive reconstructions to be more similar with real scenes. The temporal consistency loss can be described as

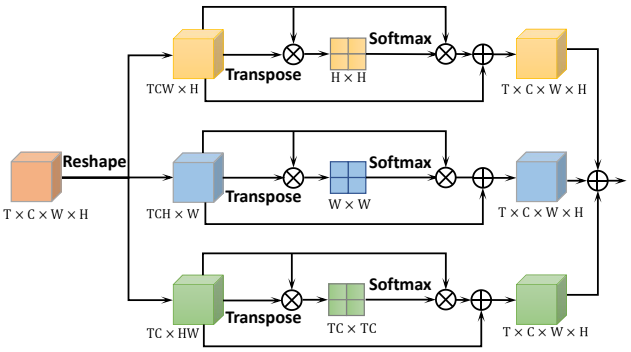


Figure 3. The architecture of the attentions modules. The branches from top to bottom calculate the vertical, horizontal, and temporal/channel correlation, respectively.

$$\mathcal{L}_C = \sum_{i=1}^{T} \|\mathbf{E}_t - \mathcal{C}(\mathbf{H}_{t-1}, \mathbf{H}_t)\|_2^2. \tag{9}$$

3.4. Learning Details

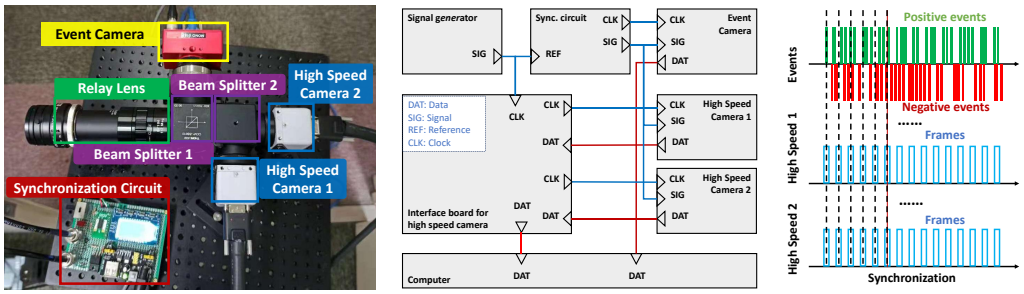
Given the reconstructed video sequence $\mathbf{H}_i$ and the corresponding ground truth $\hat{\mathbf{H}}_i$, first we employ widely used l_1 loss to evaluate the reconstruction loss

$$\mathcal{L}_{l_1} = \sum_{i=1}^{T} \|\mathbf{H}_i - \hat{\mathbf{H}}_i\|. \tag{10}$$

Since simply using l_1 reconstruction loss would suffer from blurry artifacts, we introduce the Learned Perceptual Image Patch Similarity (LPIPS [40]) loss for high level and structural similarity, which is denoted as $\mathcal{L}_{LPIPS}$. Together with the temporal consistency loss, the full loss function for the high speed HDR reconstruction from events is

$$\mathcal{L} = \mathcal{L}_{l_1} + \tau_1 \mathcal{L}_{LPIPS} + \tau_2 \mathcal{L}_C. \tag{11}$$

In the experiment, we empirically set τ_1, τ_2 to 2 and 0.2. In the training stage, our network is initialized by Kaiming initialization [9], and the losses are minimized with the adaptive moment estimation method [14], and we set the momentum parameter to 0.9. The learning rate is initially set to 10^{-4}, and divided by 10 every 50 epochs. We set



(a) Optical system (b) Electronic system (c) Data synchronization

Figure 4. The hardware implementation of our high speed HDR video imaging system with events. (a) The optical system. It contains two high speed cameras and an event camera. (b) The electronic system. It controls three cameras to synchronously capture the information of same scene. (c) The synchronize data captured by three cameras, respectively.

batch size as 4 and the whole process is trained for 100 epochs. Our model is implemented using the deep learning framework PyTorch [20], and we use NVIDIA TITAN V GPUs to train our model.

4. Experiments

In this section, we first describe our imaging system for real-world HDR video and events acquisition. Then, we provide details of our simulated and real data, as well as our experimental settings. After that, qualitative and quantitative compared results are evaluated. Finally, we conduct experiments for ablation study.

4.1. A Real-World HDR Video and Events Imaging System

In this part, we build a novel imaging system to capture paired high speed HDR videos and the corresponding event streams. To the best of our knowledge, this kind of paired real data has never been explored by previous works, due to the following reasons. First, capturing high speed HDR video itself is not an easy work, since the requirement of high speed limits the exposure time, and definitely increases the difficulty of merging images under different exposures. Second, how to align the timestamps and view field of high speed HDR camera and event camera is difficult, since two cameras capture different modal information and both suffer from the effect of noise.

To solve the above mentioned problems, we design a elaborate system to synchronously capture paired high speed HDR video and the corresponding event stream. In general, we use an event camera to capture event streams, and two high speed cameras are used to capture synchronized LDR frames, which are later merged to form an HDR frame. By aligning these cameras carefully through an elaborately designed system, we record paired high speed HDR videos and the corresponding event streams. Our entire hardware prototype is illustrated in Fig. 4. From Fig. 4(a) we can see that the lights from the objective scenes first travel through a relay lens. Then, a Thorlabs CCM1-BS013 beam splitter is utilized to divide the incident light into two equivalent components with different directions. For one direction, an iniVation DAVIS346 event camera is used to capture the event stream. For the other one, another Thorlabs CCM1-BS013 beam splitter is employed for further transmitting the lights to two Photron IDP-Express R2000 high speed cameras, in order to capture two synchronous videos. On the basis of the HDR generating method of [3] which merges several LDR images with different exposure times, we cap one of our high speed camera with a Thorlabs ND513B neutral density (ND) filter to weaken the incoming irradiance. An ND filter can shelter the energy of light uniformly along spatial and spectral dimensions. In this way, we obtain two LDR images with different scene irradiance, and avoid the alternative exposures of two images, which are difficult to control when capturing high speed videos. In our implementation, we choose an ND filter which can filter around 95% scene irradiance. Considering that both high speed cameras record images of 8-bit, we can obtain 12-bit HDR images after the merging operation. For these three cameras used in our hardware, the field of views are strictly aligned, and the timestamps are controlled by a specially designed circuits, as shown in Fig. 4(b)(c). Our HDR data reaches a high frame rate of 2000 fps.

4.2. Dataset Preparation

Simulated Data. An ideal simulated dataset for high speed HDR video reconstruction should meet several criteria, *i.e.*, high speed, HDR, and neither background nor foreground is static. However, existing HDR video dataset [4, 15, 27] do not guarantee the three requirements simultaneously. Therefore, we use non-HDR video sequences to train our method, like [22, 33]. Many previous works [10, 22, 33, 34] used ESIM simulator [21] to render events from single images along the virtual camera trajectory. This approach provides enough training data, which is beneficial for learn-

ing based methods. We directly use the simulated dataset from [33] for a coarse estimation of our model, which consists of 29 video sequences. We randomly select 15 of them for training purposes, and the other sequences are used for evaluation.

Real-World Data. In order to improve the performance of our model when reconstructing real high speed HDR videos, we use our imaging system to capture 12 typical outdoor scenes. All scenes are of high dynamic range, and cannot be well recorded by normal cameras due to the over-exposure or lack of details in the dark regions. Each video is 3.8 seconds long with 7680 frames, which infers a 2000 fps acquisition speed. We randomly choose 8 of them as training data to train the networks, and the rest are served as testing data. A glimpse of our paired real-world dataset is illustrated in Fig. 5. Noting that our HDR data reaches a high frame rate of 2000 fps, the number of events between any two consecutive frames is limited. Thus, we stack the events between a time span over five neighboring HDR frames. Since the stack of event frames can be overlapped, we still have a 2000 fps paired real data.

4.3. Experimental Settings

In the experiments, we consider three state-of-the-arts as compared methods, including the high-pass filter based non-deep method [25] (HF), the event-to-video reconstruction method [22] (E2VID), and a model which is pretrained on a high quality event to frame dataset [28] (EF). Note that we have tried our best to reproduce the best results competitive methods with the codes that are released publicly. All compared methods are conducted on the datasets mentioned above.

The reconstruction results of all compared methods are evaluated by four image quality metrics, including Peak Signal-to-Noise Ratio (PSNR), Structural Similarity [36] (SSIM), Learned Perceptual Image Patch Similarity [40] (LPIPS), and our temporal consistency loss (TC) which is introduced in Section 3.3. PSNR and SSIM measure the 2D spatial fidelity, LPIPS evalutes the perceptual similarity, and TC measures the temporal fidelity of an image. Larger values of PSNR and SSIM suggest better results, while smaller values of LPIPS and TC show better reconstruction.

4.4. Comparisons with State-of-the-arts

Experiments on Simulated Data. Tab. 1 summarize the numerical results according to the average results of all metrics. The best performance is highlighted in **bold**. We can see that our method provides better results in most scenes for all error metrics, and the averaging results of all scenes significantly outperforms all compared methods in both spatial and temporal domain, which demonstrates the superiority of our proposed convolutional recurrent neural network. To visualize the experimental results, several representative restored videos are shown in Fig. 6. The event frame and representative reconstructed frames of HF/E2VID/EF/Ours and ground truth are shown from left to right. We can see that the reconstructed result of our method is more sharp and closer to ground truth, which is consistent with the numerical results.

Table 1. Performance comparisons on simulated data.

Method	PSNR	SSIM	LPIPS	TC
HF	10.99	0.2708	0.4434	1.2051
E2VID	12.78	0.5753	0.3541	1.0305
EF	13.23	0.5914	0.3030	0.9729
Ours	**15.31**	**0.7084**	**0.2424**	**0.5198**

Table 2. Performance comparisons on real data.

Method	PSNR	SSIM	LPIPS	TC
HF	9.12	0.1673	0.8307	0.4515
E2VID	14.72	0.4781	0.4200	0.4451
EF	14.77	0.4064	0.5020	0.4191
Ours	**16.41**	**0.4783**	**0.3737**	**0.3539**

Experiments on Real-World Data. To further evaluate the effectiveness of our method, we compare our work with existing methods on our captured real-world dataset. The averaging results are provided in Tab. 2. We can see that our method outperforms all competing methods in both spatial and temporal metrics, which is consistent with the simulated experiments. To visualize the experimental results, several representative reconstructed frames are shown in Fig. 7. The event frame and representative reconstructed frames of HF/E2VID/EF/Ours, and ground truth are shown from left to right. It can be seen that frames recovered from our method approximate ground truth well and is significantly better than other reconstruction methods. It demonstrates the effectiveness of our method on real-world dataset.

4.5. Ablation Study

To evaluate the effect of the recurrence of ConvLSTM and temporal consistency loss, we compare our method without recurrent module, without temporal consistency loss and our entire model on simulated dataset as an example. The corresponding results are provided in Tab. 3. We can see that our method performs better than that without

Table 3. Evaluation results of our reconstruction with and without different components under simulated dataset.

Model	PSNR	SSIM	LPIPS	TC
w/o recurrent	14.71	0.6362	0.2891	0.5872
w/o TC loss	15.17	0.6721	0.2455	0.5549
Ours	**15.31**	**0.7084**	**0.2424**	**0.5198**

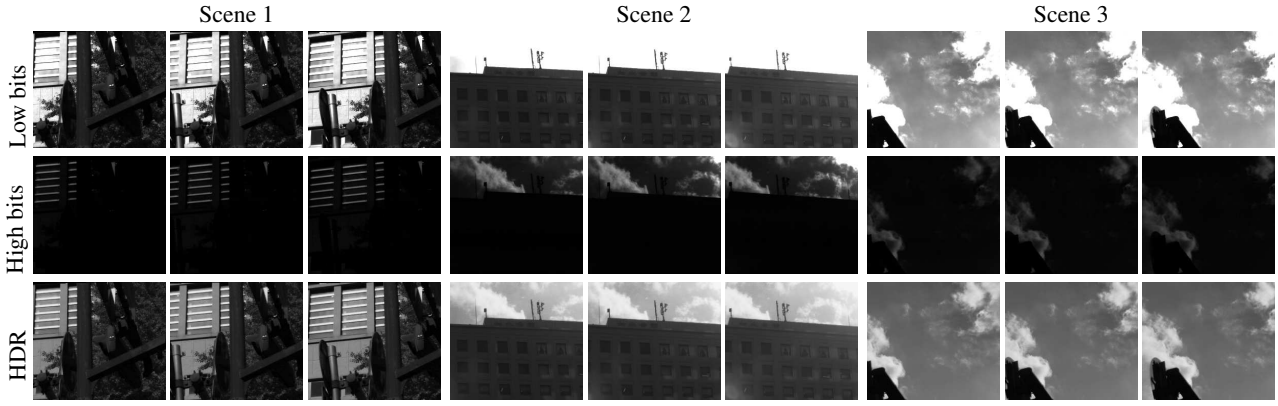


Figure 5. Three representative scenes of our captured real dataset. In order to recognize the scene motions, the shown two consecutive frames are chosen at the interval of 100 real frames.

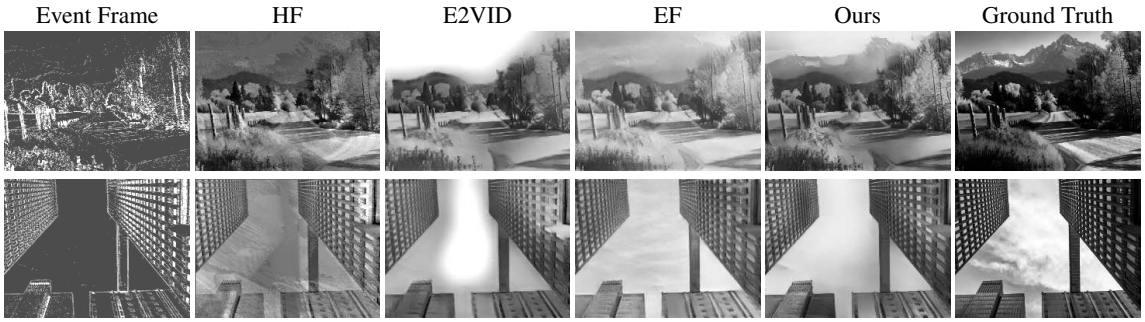


Figure 6. Qualitative reconstruction results on simulated data.

Figure 7. Qualitative reconstruction results on real-world data.

recurrent module or temporal consistency loss in most metrics. It verifies that our deep recurrent reconstruction model can effectively improve the spatial and temporal fidelity.

5. Conclusion

In this paper, we present a novel convolutional recurrent neural network to reconstruct standard HDR videos from event streams. Specifically, our model consists of a shared feature extractor, a deformable convolution based alignment module, and a convolutional recurrent fusion and reconstruction network with attention mechanism. In addition, we employ a temporal consistency loss to minimize the gap between reconstructions and real scenes. Furthermore, we collect the first dataset with paired high speed HDR and event data of real dynamic scenes, which will be accessible publicly to help other researchers who are interested in this field. Experimental results have verified the effectiveness of our proposed high speed HDR video reconstruction method and our collected paired dataset.

Acknowledgments This work was supported by the National Natural Science Foundation of China under Grants No. 61827901, No. 62088101, and No. 61936011.

References

[1] Matthew Cook, Luca Gugelmann, Florian Jug, Christoph Krautz, and Angelika Steger. Interacting maps for fast visual interpretation. In *Proc. of International Joint Conference on Neural Networks*, pages 770–776. IEEE, 2011. 2

[2] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 764–773, 2017. 4

[3] Paul E Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes*, pages 1–10. 2008. 3, 6

[4] Jan Froehlich, Stefan Grandinetti, Bernd Eberhardt, Simon Walter, Andreas Schilling, and Harald Brendel. Creating cinematic wide gamut hdr-video for the evaluation of tone mapping operators and hdr-displays, 2014. 6

[5] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3146–3154, 2019. 5

[6] G. Gallego, T. Delbruck, G. M. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza. Event-based vision: A survey. *IEEE Trans. Pattern Analysis and Machine Intelligence* , pages 1–1, 2020. 1

[7] Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3867–3876, 2018. 1

[8] Jin Han, Chu Zhou, Peiqi Duan, Yehui Tang, Chang Xu, Chao Xu, Tiejun Huang, and Boxin Shi. Neuromorphic camera guided high dynamic range imaging. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1730–1739, 2020. 2

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015. 5

[10] S. Mohammad Mostafavi I., Jonghyun Choi, and Kuk-Jin Yoon. Learning to super resolve intensity images from events. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 3, 5, 6

[11] Zhe Jiang, Yu Zhang, Dongqing Zou, Jimmy Ren, Jiancheng Lv, and Yebin Liu. Learning event-based motion deblurring. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*. 2

[12] Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High dynamic range video. *ACM Trans. on Graphics*, 22(3):319–325, 2003. 3

[13] H Kim, A Handa, R Benosman, SH Ieng, and AJ Davison. Simultaneous mosaicing and tracking with an event camera. In *Proc. of Conference on British Machine Vision Conference (BMVC)*, 2014. 2, 3

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. of International Conference on Learning representations(ICLR)*, 2015. 5

[15] Joel Kronander, Stefan Gustavson, Gerhard Bonnet, and Jonas Unger. Unified hdr reconstruction from raw cfa data. In *Proc. of International Conference on Computational Photography (ICCP)*, pages 1–9, 2013. 6

[16] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 170–185, 2018. 5

[17] Jun Haeng Lee, Kyoobin Lee, Hyunsurk Ryu, Paul KJ Park, Chang-Woo Shin, Jooyeon Woo, and Jun-Seok Kim. Real-time motion estimation based on event-based vision sensor. In *Proc. of IEEE International Conference on Image Processing (ICIP)*, pages 204–208. IEEE, 2014. 1

[18] Min Liu and Tobias Delbrück. Adaptive time-slice block-matching optical flow algorithm for dynamic vision sensors. In *Proc. of Conference on British Machine Vision Conference (BMVC)*, page 280, 2018. 1

[19] Liyuan Pan, Cedric Scheerlinck, Xin Yu, Richard Hartley, Miaomiao Liu, and Yuchao Dai. Bringing a blurry frame alive at high frame-rate with an event camera. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6820–6829, 2019. 2, 5

[20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Proc. of Conference on Neural Information Processing Systems (NeurIPS)*, pages 8026–8037, 2019. 6

[21] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. Esim: an open event camera simulator. In *Proc. of Conference on Robot Learning (CoRL)*, pages 969–982, 2018. 1, 2, 5, 6

[22] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE Trans. Pattern Analysis and Machine Intelligence* , 2019. 1, 2, 3, 4, 5, 6, 7

[23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proc. of International Conference on Medical image computing and computer-assisted intervention*, pages 234–241, 2015. 5

[24] Daniel Saner, Oliver Wang, Simon Heinzle, Yael Pritch, Aljoscha Smolic, Alexander Sorkine-Hornung, and Markus H Gross. High-speed object tracking using an asynchronous temporal contrast sensor. In *VMV*, pages 87–94. Citeseer, 2014. 1

[25] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. Continuous-time intensity estimation using event cameras. In *Proc. of Asian Conference on Computer Vision (ACCV)*, pages 308–324. Springer, 2018. 7

[26] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Proc. of Conference on Neural Information Processing Systems (NeurIPS)*, 28:802–810, 2015. 5

[27] Li Song, Yankai Liu, Xiaokang Yang, Guangtao Zhai, Rong Xie, and Wenjun Zhang. The sjtu hdr video sequence dataset. In *Proc. of International Conference on Quality of Multimedia Experience (QoMEX)*, page 100, 2016. 6

[28] Timo Stoffregen, Cedric Scheerlinck, Davide Scaramuzza, Tom Drummond, Nick Barnes, Lindsay Kleeman, and Robert Mahony. Reducing the sim-to-real gap for event cameras. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 1, 3, 7

[29] Neuromorphic Vision Systems. https://inivation.com/. 2

[30] Matias Tassano, Julie Delon, and Thomas Veit. Fastdvdnet: Towards real-time deep video denoising without flow estimation. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1354–1363, 2020. 4

[31] Michael D Tocci, Chris Kiser, Nora Tocci, and Pradeep Sen. A versatile hdr video production system. *ACM Trans. on Graphics*, 30(4):1–10, 2011. 3

[32] Bishan Wang, Jingwei He, Lei Yu, Gui-Song Xia, and Wen Yang. Event enhanced high-quality image recovery. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 5

[33] Lin Wang, Yo-Sung Ho, Kuk-Jin Yoon, et al. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10081–10090, 2019. 1, 2, 3, 6, 7

[34] Lin Wang, Tae-Kyun Kim, and Kuk-Jin Yoon. Eventsr: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8315–8325, 2020. 2, 6

[35] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. EDVR: Video restoration with enhanced deformable convolutional networks. In *Proc. of Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, pages 0–0, 2019. 3, 4

[36] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Processing*, 13(4):600–612, 2004. 7

[37] Nicolai Waniek, Johannes Biedermann, and Jorg Conradt. Cooperative slam on small mobile robots. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1810–1815, 2015. 1

[38] Yi Xu, Longwen Gao, Kai Tian, Shuigeng Zhou, and Huyang Sun. Non-local convlstm for video compression artifact reduction. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 7043–7052, 2019. 3

[39] Huanjing Yue, Cong Cao, Lei Liao, Ronghe Chu, and Jingyu Yang. Supervised raw video denoising with a benchmark dataset on dynamic scenes. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2301–2310, 2020. 3, 4

[40] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–595, 2018. 5, 7

[41] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 989–997, 2019. 1

[42] Dongqing Zou, Feng Shi, Weiheng Liu, Jia Li, Qiang Wang, Paul-KJ Park, Chang-Woo Shi, Yohan J Roh, and Hyunsurk Eric Ryu. Robust dense depth map estimation from sparse dvs stereos. In *Proc. of Conference on British Machine Vision Conference (BMVC)*, volume 1, 2017. 1