Journal of Electronic Imaging

JElectronicImaging.org

Photorealistic image reconstruction from hybrid intensity and event-based sensor

Prasan Shedligeri
Kaushik Mitra

Prasan Shedligeri, Kaushik Mitra, "Photorealistic image reconstruction from hybrid intensity and event-based sensor," *J. Electron. Imaging* **28**(6), 063012 (2019), doi: 10.1117/1.JEI.28.6.063012.

Photorealistic image reconstruction from hybrid intensity and event-based sensor

Prasan Shedligeri* **and Kaushik Mitra**
Indian Institute of Technology Madras, Computational Imaging Laboratory, Department of Electrical Engineering, Chennai, Tamil Nadu, India

Abstract. Event sensors output a stream of asynchronous brightness changes (called "events") at a very high temporal rate. Previous works on recovering the lost intensity information from the event sensor data have heavily relied on the event stream, which makes the reconstructed images nonphotorealistic and also susceptible to noise in the event stream. A method is proposed to reconstruct photorealistic intensity images from a hybrid sensor consisting of a low-frame-rate conventional camera and the event sensor. The texture-rich information from a traditional image sensor with the motion-rich information from the event sensor is exploited, producing photorealistic high-frame-rate videos. To accomplish the task, the low-frame-rate intensity images are warped to temporally dense locations of the event data. The results obtained from the proposed algorithm are more photorealistic compared to any of the previous state-of-the-art algorithms. The algorithm's robustness to abrupt camera motion and noise in the event sensor data is also demonstrated. © *2019 SPIE and IS&T* [DOI: 10.1117/1.JEI.28.6.063012]

Keywords: event sensors; image reconstruction; temporal super-resolution.

Paper 190493 received May 24, 2019; accepted for publication Dec. 4, 2019; published online Dec. 20, 2019.

1 Introduction

Event sensors[1,2] are a new generation of asynchronous or neuromorphic sensors that capture changes in brightness values at individual pixels at the time instant they occur. Event sensors output either a positive or a negative event at a pixel, based on whether the pixel sees an increase or a decrease in the intensity value. Event sensors can sense these brightness changes at a temporal resolution of 1 μs. This is unlike traditional image sensors, which simultaneously capture the absolute intensity values at all the pixels at a fixed frame rate. Due to the asynchronous nature of event sensors, they have several advantages over traditional sensors, such as low power requirement, low latency, and low bandwidth operation. These advantages of the event sensors over the traditional image sensor make them especially suitable for applications such as autonomous navigation and augmented/virtual reality. Recently, there has been increased interest in visual odometry and SLAM,[3–7] ego-motion estimation,[8–10] and three-dimensional (3-D) reconstruction[3,6,10–13] with the help of event sensors. There have also been attempts at solving other high level computer vision tasks with event sensors, such as object recognition,[14] gesture recognition,[15] and semantic segmentation.[16]

Event-based sensors provide a power efficient way of converting the megabytes of per-pixel intensity data into a stream of spatially sparse but temporally dense events. However, the event stream cannot be directly visualized like a normal video with which we as human beings are familiar. This calls for an algorithm that can convert this stream of event data to a more familiar version of image frames. These reconstructed intensity frames could also be used as an input for traditional frame-based computer vision algorithms, such as multiview stereo, object detection, etc. Previous attempts[17–20] at converting the event stream into images have heavily relied on event data. Although these methods do a good job of recovering the intensity frames, they suffer from two major disadvantages: (a) the intensity frames do not look photorealistic and (b) some of the objects in the scene can go missing in the recovered frames because they are not producing any events (edges parallel to the sensor motion do not trigger any events).

In this paper, a method is proposed to reconstruct photorealistic intensity images at a high frame rate. As the absolute intensity and fine texture information is lost during the encoding of the events, the information from the frames of the conventional image sensor is used to reconstruct the photorealistic intensity images. The conventional image sensor will compensate for the lost spatial information due to encoding of the events. The event sensor will compensate for the motion information lost due to the fixed-frame-rate sampling of the conventional image sensor. There exists a commercially available hybrid sensor consisting of a colocated low-frame-rate intensity sensor and an event-based sensor called dynamic and active pixel vision sensor (DAVIS).[21] Figure 1 summarizes the overall approach to reconstruct the temporally dense photorealistic intensity images using the hybrid sensor. The proposed method has four main steps. In the first step, a dense depth map is estimated using successive intensity frames obtained from the traditional image sensor. For estimating depth, a traditional iterative optimization scheme is utilized, which is initialized by a depth map obtained from a deep learning-based optical flow estimation algorithm. In the second step, the event data between successive intensity frames are mapped to multiple pseudointensity frames using Ref. 17. Next, the pseudointensity frames and the dense depth maps obtained from the first step are used to estimate temporally dense camera ego motion by direct visual odometry. Finally, in the fourth step, the successive intensity frames are warped to intermediate

*Address all correspondence to Prasan Shedligeri, E-mail: ee16d409@ee.iitm.ac.in

1017-9909/2019/$28.00 © 2019 SPIE and IS&T

Fig. 1 Overview of the approach: The main blocks of the algorithm are (a) an iterative depth and camera pose estimation technique for successive intensity frames, (b) mapping event data into pseudointensity frames using Ref. 17, (c) direct visual odometry-based sensor ego-motion estimation for intermediate event frame locations, and (d) a warping module for warping intensity images to intermediate locations.

temporal locations of the pseudointensity frames to obtain the photorealistic reconstruction. With extensive experiments, it is shown that our proposed method is able to reconstruct photorealistic intensity images at a high frame rate and is also robust to noisy events in the event stream. To summarize, the contributions of this work are:

- A pipeline is proposed, which uses a hybrid event and low-frame-rate intensity sensor that can reconstruct temporally dense photorealistic intensity images. This would be difficult to obtain with only the conventional image sensor or the event sensor.
- The event sensor is used for estimating temporally dense sensor ego motion and the low-frame-rate intensity images for obtaining a spatially dense depth map.
- A high quality temporally dense photorealistic reconstruction is demonstrated using the proposed method on real data captured from DAVIS.
- The algorithm's robustness to abrupt camera motion and noisy events in the event sensor data is also demonstrated.

2 Related Work

There has been increased interest in visual odometry and SLAM,[3–7] ego-motion estimation,[8–10] and 3-D reconstruction[3,6,10–13] with the help of event sensors. Event sensors have also been shown to perform well in mainstream vision tasks, such as image classification, corner detection, etc., after reconstructing intensity images.[22] We refer the reader to Ref. 23 for further research interest in event-based vision algorithms.

2.1 *Intensity Image Reconstruction from Events*

The proposed work is very closely related to other previous works, which reconstruct intensity images from events.[17–20] Most previous works[17–19] cannot recover the true intensity information of the scene as they use only the events to estimate the intensity images. Some works[3,4] reconstruct intensity images as a by-product of sensor tracking from event data over 3-D scenes but are not able to recover the true intensity information. Recently, Ref. 20 demonstrated that event data and the intensity image data can be used in a complementary filter (CF) to reconstruct intensity frames at a higher frame rate. Although Ref. 20 makes use of the intensity images, the reconstructed images tend to be blurry and are adversely affected by noisy events due to the lack of any regularization in their proposed method. Another unpublished manuscript[24] proposes to use a learning-based denoising algorithm to fuse event sensor and image sensor data to reconstruct images at a high frame rate. Although the paper shows promising results on the synthetic data, it fails to show

similar quality results on real data captured using DAVIS. Another recent work on reconstructing intensity images from event sensors is events-to-video (E2Vid).[22]

3 Background on Event Sensors

An event sensor triggers events asynchronously whenever there is a brightness change in the scene. The output of an event sensor is a four-tuple (x, y, p, t), where x and y represent the location of the pixel in the sensor, $p \in [-1, +1]$ is the polarity of the triggered event, and t is the microsecond precise timestamp at which the event was triggered. An event sensor triggers a positive or a negative event whenever the log brightness change is more than or less than a threshold τ, respectively. The polarity p of the triggered event is given as

$$p = \begin{cases} +1, & \log(I_{t+\delta t}) - \log(I_t) \geq \tau \\ -1, & \log(I_{t+\delta t}) - \log(I_t) \leq -\tau \end{cases},$$

where $\delta t > 0$ and is of the order of microseconds. In cases where the log brightness change at a particular pixel is within the threshold, the dynamic vision sensor does not trigger any event and hence saves power and bandwidth. As event sensors output only the brightness changes, it is impossible to recover the absolute scene intensity information. Further, the noise introduced due to the nonideal hardware and the event quantization makes it more challenging to recover intensity image information.

In all the experiments, a commercially available hybrid sensor called "DAVIS 240C"[25] supplied by Inivation is used. This particular sensor consists of a colocated event sensor and an image sensor, each with a resolution of 180×240 pixels. We use a 6-mm lens, which approximately gives horizontal and vertical field of views of 40.5 deg and 49.4 deg, respectively. The image sensor gives a raw grayscale intensity image with 8 bits and the frame rate is 24 fps. The event sensor uses an address event representation to record the data from the environment. The data are logged in an ".aedat," which is a binary file containing 8 bytes of data per line. The 8 bytes of data consist of the timestamp, the spatial location of the event, and the event polarity. The event sensor is capable of sending out 12M events per second. More information about the data format and other specifications can be found in Ref. 25.

4 Photorealistic Image Reconstruction

Here, a method is proposed to reconstruct photorealistic intensity images using the event stream obtained from an event sensor. The conventional image sensor will compensate for the fine texture and the absolute intensity information, which is lost in the event stream. As can be seen from Fig. 1, the proposed algorithm has four major steps to reconstruct the temporally dense photorealistic image reconstruction: (a) estimate dense depth maps d_k and d_{k+1} corresponding to the successive intensity frames I_k and I_{k+1} and the relative pose ξ between them (Sec. 4.1); (b) reconstruct pseudointensity frames E_k^j at uniformly spaced temporally dense locations $j = 1, 2, \ldots N$ between every successive intensity frame I_k and I_{k+1}; (c) estimate temporally dense sensor ego-motion estimates ξ_k^j and ξ_{k+1}^j for each intermediate pseudointensity frame with respect to the intensity frames I_k and I_{k+1} (Sec. 4.2); and (d) forward warp the intensity frames I_k and I_{k+1} to the intermediate location of each of the pseudointensity frames E_k^j and blend them (Sec. 4.3).

4.1 *Depth Estimation from Two Successive Intensity Images*

One of the important steps in the proposed algorithm is forward warping the intensity images to multiple intermediate temporal locations between successive intensity frames. However, warping can introduce undesired holes in the final reconstructed images at regions of disocclusion. This can be solved by warping both the successive intensity frames, I_k and I_{k+1}, to the intermediate locations. This requires the estimation of two dense depth maps d_k and d_{k+1} corresponding to images I_k and I_{k+1}, respectively. Figure 2 shows the overall scheme of estimating dense depth maps from successive intensity frames. We initialize the depth estimates d_k and d_{k+1} from the optical flow and the 6 degrees of freedom (DOF) camera pose ξ with zero rotation and translation. Here, ξ is the 6 DOF relative camera pose at I_{k+1} with respect to I_k. The intensity image I_{k+1} is warped to the location of I_k with the current estimate of d_k and ξ to obtain $\hat{I}_k$. Similarly, image I_k is warped to the location of I_{k+1} to obtain $\hat{I}_{k+1}$. The photometric reconstruction loss $\mathcal{L}_{\text{ph}}$ is defined as

$$\mathcal{L}_{\text{ph}}(d_k, d_{k+1}, \xi) = \|(\hat{I}_k - I_k)\|_1 + \|(\hat{I}_{k+1} - I_{k+1})\|_1. \quad (1)$$

By minimizing the above reconstruction loss, $\mathcal{L}_{\text{ph}}$, it is possible to estimate the depth maps d_k and d_{k+1} and 6 DOF relative pose ξ. An edge aware Laplacian smoothness prior is enforced on the estimated depth maps d_k and d_{k+1} by taking inspiration from Ref. 27. The smoothness loss $\mathcal{L}_{\text{sm}}$ is defined as

$$\mathcal{L}_{\text{sm}}(d) = \sum \|\nabla_x d\| e^{-\beta\|\nabla_x I\|} + \|\nabla_y d\| e^{-\beta\|\nabla_y I\|}, \quad (2)$$

Fig. 2 Estimating dense depth maps and relative pose of two successive intensity images: The optical flow estimated from PWC-Net[26] is used to obtain an initial depth estimate and initialize the relative pose to zero rotation and translation. The photometric error is iteratively minimized over the depth maps d_k and d_{k+1} and the relative pose ξ.

where I is the intensity image, d is the corresponding dense depth map, and ∇_x and ∇_y are the x- and y-gradient operators, respectively. Overall, the dense depth estimate d_k, d_{k+1}, and the relative pose ξ are estimated as

$$\xi, d_k, d_{k+1} = \arg\min_{\xi, d_k, d_{k+1}} \mathcal{L}_{\text{ph}}(d_k, d_{k+1}, \xi) + \lambda_{\text{sm}}[\mathcal{L}_{\text{sm}}(d_k) + \mathcal{L}_{\text{sm}}(d_{k+1})]. \quad (3)$$

Equation (3) is a nonconvex optimization problem and hence a good initialization of depth and pose is essential to avoid local minima. Here, we use optical flow between the successive intensity frames obtained from PWC-Net[26] as an initial estimate of the depth.[28] For a scene that is static, the optical flow can be related to the scene depth and the camera motion as follows:[28]

$$\mathbf{u}(x, y) = p(x, y)\mathbf{A}(x, y)\mathbf{T} + \mathbf{B}(x, y)\mathbf{\Omega}, \quad (4)$$

where $p(x, y)$ is the inverse of the depth at the scene point captured by the pixel (x, y). The matrices $\mathbf{A}$ and $\mathbf{B}$ are dependent only on the image position and the focal length of the camera. The vectors $\mathbf{T} = (T_x, T_y, T_z)$ and $\mathbf{\Omega} = (\Omega_x, \Omega_y, \Omega_z)$ are the camera's instantaneous translations and rotations, respectively. Under the assumption of a static scene and a purely translating camera motion, it can be seen that optical flow is inversely proportional to the depth map of the scene. It should be noted that the inverse of the optical flow does not exactly provide us the depth of the scene for 6 DOF general motion. However, in practice, we found that it can be used as an initialization to the optimization iterations in Eq. (3). The pose is initialized with zero rotation and translation.

4.2 *6 DOF Relative Pose Estimation by Direct Matching*

To achieve the goal of photorealistic reconstruction, the successive intensity frames captured by the image sensor are warped to the intermediate temporal location of an event frame. For warping, the 6 DOF camera pose between the temporal locations of the successive intensity frames and that of the intermediate event frames are to be determined. We reconstruct pseudointensity images from events using Ref. 17 at the temporal locations of the intermediate event frames as well as the successive intensity frames. A brief explanation of the algorithm used to reconstruct pseudointensity images is provided in the following paragraph.

The pseudointensity image reconstruction from events in Ref. 17 is cast as a regularized integration of events. Each incoming event is added to a previously reconstructed log-intensity frame to produce an intermediate frame. This intermediate frame is then assumed to be generated from a Poisson likelihood model whose mean and variance is the final reconstructed image. Hence, the task is to estimate the mean and variance of a Poisson distribution given only a single observation. As this is an ill-posed estimation problem, the authors use a regularization term on the final reconstructed image. This regularization term is defined on a manifold of event timestamps called the surface of active events. This manifold ensures that the regularization between pixels with different timestamps is reduced while pixels having similar timestamps have higher regularization.

As shown in Fig. 3, the objective here is to estimate the relative camera pose between E_k^0, E_{k+1}^0, and the pseudointensity images E_k^j $(j = 1,2,\ldots N)$. This relative pose is used to warp the successive intensity frames to the intermediate locations specified by the event frames (E_k^j) and hence reconstruct photorealistic intensity images.

Fig. 3 Estimating relative pose of intermediate pseudointensity images: The 6-DOF camera pose of E_k^j with respect to E_k^0 and E_{k+1}^0 are estimated by iteratively minimizing the photometric error between the warped image $\hat{E}_k^j$ and the target image E_k^j. The photometric error over the relative poses ξ_k^j and ξ_{k+1}^j is minimized using the known depth estimates d_k and d_{k+1}.

Let ξ_k^j represent the 6 DOF camera pose of the intermediate pseudointensity image E_k^j with respect to E_k^0 and ξ_{k+1}^j be the 6 DOF camera pose of E_k^j with respect to E_{k+1}^0. The current estimate of relative camera pose ξ_k^j and the known depth estimate d_k is used to inverse warp the pseudointensity frame E_k^j to the location of E_k^0 to obtain $\hat{E}_k^0$. Similarly, the pseudointensity frame E_k^j is inverse warped to the location of E_{k+1}^0 to obtain $\hat{E}_{k+1}^0$ using the current estimate of relative pose ξ_{k+1}^j and the known depth d_{k+1}. The photometric loss $\mathcal{L}_p$ is defined as the mean absolute error between the warped intensity frame and the ground truth frame

$$\mathcal{L}_p(\xi_k^j) = \|E_k^0 - \hat{E}_k^0\|_1, \tag{5}$$

$$\mathcal{L}_p(\xi_{k+1}^j) = \|E_{k+1}^0 - \hat{E}_{k+1}^0\|_1. \tag{6}$$

By composing the relative pose estimates, ξ_k^j and $(\xi_{k+1}^j)^{-1}$, the overall pose between I_k and I_{k+1} is obtained. This knowledge is used to regularize the relative camera pose estimates ξ_k^j and ξ_{k+1}^j with $\mathcal{L}_p(\xi_k^j, \xi_{k+1}^j) = \|I_k - \hat{I}_k\|_1$. Overall,

$$\xi_k^j, \xi_{k+1}^j = \arg\min_{\xi_k^j, \xi_{k+1}^j} \mathcal{L}_p(\xi_k^j) + \mathcal{L}_p(\xi_{k+1}^j) + \lambda_r \mathcal{L}_p(\xi_k^j, \xi_{k+1}^j), \tag{7}$$

where λ_r is the regularization parameter.

4.3 *Forward Warping and Blending*

At this stage, depth maps d_k and d_{k+1} are obtained corresponding to intensity images I_k and I_{k+1}, respectively. A source-target mapping (forward warping) is done from two images I_k and I_{k+1} using the estimated relative poses ξ_k^j and ξ_{k+1}^j to the latent image I_k^j and alpha blending them. In forward warping, the pixel (x, y) of the image I_k is mapped to the pixel (x', y') as follows:

$$(x', y', 1)^T = \frac{1}{h} K[R|t]K^{-1} d_k(x, y) \times (x, y, 1)^T, \tag{8}$$

where K is the intrinsic matrix of the camera and R and t are the rotation and translation parameters given by ξ_k^j. Note that while the values (x, y) lie on a regular grid, the transformed values (x', y') need not necessarily lie on the regular rectangular grid. To avoid any holes in the resultant warped image, we splat the intensity values, which are transformed from the source image I_k to the target image at position I_k^j. Similarly, we also transform the source image I_{k+1} to the target image position I_k^j using the depth estimate d_{k+1} and the pose estimate ξ_{k+1}^j.

Now, there are two images warped from two different source images at the single target image location. In order to combine the two frames into a single frame, the simple technique of alpha blending is used. Alpha blending performs a convex combination of two images where the parameter α determines of the weight assigned to each image. The value of α is set to 0.5 for overlapping image regions and is 1.0 for regions where one of the images has a nonzero value. The value of α linearly increases in the transition between 0.5 and 1.0.

5 Experiments

For all the experiments, a DAVIS[21] sensor is used, which is commercially available and has a conventional image sensor and an event sensor bundled together. We used the recently proposed dataset of Refs. 5 and 20, which consists of several video sequences captured using DAVIS. Spatially dense depth maps at the locations of low-frame-rate intensity frames and temporally dense sensor ego motion using the event sensor data are obtained to warp the low-frame-rate intensity frames to intermediate camera locations. For estimating depth, the edges of the depth obtained from the optical flow estimate are enhanced using a fast bilateral solver.[29] The output of this bilateral solver is then used as an initialization for the iterative depth refinement scheme.

Using the event stream from each sequence in the dataset, pseudointensity estimates are generated using the algorithm proposed in Ref. 17. Nonoverlapping blocks of 2000 events are stacked into a frame to generate a corresponding pseudointensity frame using Ref. 17. These pseudointensity frames are then used for estimating the temporally dense sensor ego motion.

A hyperparameter search is conducted for different values of β, λ_{sm}, and λ_r in Eqs. (2), (3), and (7). For β, the search was done between 1 and 20, with steps of 5, essentially searching over 5 values $\{1, 5, 10, 15, 20\}$. For λ_r, we searched over $\{0.003, 0.01, 0.03, 0.1\}$ and for λ_{sm} we searched over $\{0.1, 0.3, 1.0, 3.0\}$. The values that gave us the visually best results are used. Finally, we had $\beta = 10$, $\lambda_r = 0.01$, and $\lambda_{\text{sm}} = 1.0$. For pose estimation, $\lambda_r = 0.01$ in Eq. (7) is used. We use the Adam optimizer[30] to solve Eqs. (3) and (7).

The number of intensity frames interpolated between successive intensity frames is adaptive to the event rate produced. This is because events are binned into a frame based on number of events instead of binning all the events occurring in a particular time interval. This is also known as stacking by number in Ref. 24. The typical event rates in an event sensor can range from 2×10^5 to 1×10^6 events per second. As a successive 2000 events are being binned into a single frame, the frame rate can range from 100 to 500 frames/s. Theoretically, this frame rate can be further increased by binning overlapping events into a frame. For the video sequences shown in this paper, we have mentioned the frame rate of the video wherever appropriate (Fig. 4).

5.1 *Depth Estimation*

In Fig. 5, the effectiveness of our proposed method is demonstrated for estimating depth. An initial estimate of depth from a deep learning method is used and iteratively refined. We empirically found that using PWC-Net[26] to initialize the depth estimate for the iterative optimization scheme gave consistently good results. Initializing the depth from Refs. 31 and 32 is also experimented with and comparisons are provided in the accompanying multimedia file (Video 1). In Table 1, quantitative results are provided, which compares the accuracy of two different initialization schemes.[26,31] As there are no real event-based datasets with ground truth depth maps, an redgreenblue-depth dataset[33] captured using a Kinect sensor is used. As the algorithm requires only an

Fig. 4 Effect of two different initialization schemes on depth estimation.

Fig. 5 Estimated depth maps from our proposed method.

Table 1 Predicted depth accuracy for different depth initialization schemes. Lower values of "Abs" and "root mean squared error (RMSE)" are preferred while higher values of "$\delta < th$" are preferred.

	DeMoN[31] initialization		PWC-Net[26] initialization	
	brown_bm_1	brown_bm_2	brown_bm_1	brown_bm_2
Abs	0.55	0.61	0.45	0.3
RMSE	0.71	0.8	0.58	0.45
$\delta < 1.25$	0.27	0.33	0.24	0.58
$\delta < 1.25^2$	0.46	0.56	0.58	0.7
$\delta < 1.25^3$	0.62	0.66	0.73	0.81

RGB image sequence to compute the depth map, a real dataset is used instead of a synthetic one containing events. These datasets contain multiple video sequences with an RGB video and corresponding depth maps obtained from the Kinect time-of-flight sensor. Two sequences "brown_bm_1" and "brown_bm_2" are used here for quantitative evaluation. The depth maps from this dataset contain holes, pixels where depth values are not acquired. These invalid pixels are masked while computing the error metrics. The different metrics used for quantifying depth accuracies are

- Abs: $\frac{1}{n}\Sigma|d_i - \hat{d}_i|$
- RMSE: $\sqrt{\frac{1}{n}\Sigma(d_i - \hat{d}_i)^2}$
- Accuracy: % of d_i s.t. $\max(\frac{d_i}{\hat{d}_i}, \frac{\hat{d}_i}{d_i}) = \delta < th$.

From Table 1, it can be seen that the predicted depth is not as accurate as some of the relevant state-of-the art methods. However, there is clearly an advantage by using PWC-Net[26] to initialize the depth estimate over DeMoN.[31]

Fig. 6 Comparing the reconstruction from the proposed algorithm using a hybrid sensor data (such as DAVIS) with that of CF,[20] MR,[17] and E2Vid.[22] Note that MR only uses events for reconstruction. In column (a) inset, the zoomed-in version of an image region is shown. We can clearly see that our proposed reconstruction method is able to recover the image region well compared to other state-of-the-art methods.

5.2 *Photorealistic Intensity Image Reconstruction*

In Figs. 6 and 7, the intensity images reconstructed using the proposed method and the one proposed in Refs. 20 and 34 are compared. While manifold reconstruction (MR)[17] utilizes only event sensor data, CF[20] uses both event sensor data as well as information from intensity images. For fairness in comparison, intensity images are generated from Refs. 17 and 20 for every 2000 events in the sequence. In Ref. 20, the cut-off frequency was initialized to 6.28 rad/s and other parameters were dynamically updated to yield the best results. More reconstruction results obtained using the proposed technique are shown on the data captured by us in Fig. 8.

Peak-signal-to-noise-ratio (PSNR) is also computed for the reconstructed intermediate intensity frames and compared with the state-of-the-art algorithms in Table 2. Note that for real hybrid sensor data, we do not have access to the ground-truth intermediate frames. In order to overcome this, synthetic data are generated by considering every fifth frame in a video sequence[5] as the successive intensity frames. By doing this, we now have the four ground-truth intermediate frames. It is also made sure that the successive frames in the generated synthetic sequence have enough overlap between them. The four intermediate frames are interpolated using the proposed method as well as methods proposed in Refs. 17 and 20. Using the ground-truth intermediate frames, the PSNR for each of the methods is computed on three different sequences. From Table 2, it is clear that the proposed

Fig. 7 Qualitative comparisons of reconstructions. The frame upconversion rate for the sequences from top to bottom are, respectively, 150×, 30×, 7×, and 15×. (Video 1, MPEG, 15 MB [URL: https://doi.org/10.1117/1.JEI.28.6.063012.1]).).

Fig. 8 More results obtained using the proposed technique on our data. The frame upconversion rate for the sequences from top to bottom are respectively 32× and 4×. (Video 1).

Table 2 Comparing PSNR for different sequences.

Sequence	MR[17]	CF[20]	Ours
Slider_depth	25.23	33.8	38.4
Poster_6DOF	26.11	34.59	39.73
Boxes_6DOF	23.29	31.44	36.87

algorithm outperforms other state-of-the-art methods. MR[17] performs the worst in the metric of PSNR as it does not include the intensity image information. Although it is not fair to compare MR[17] with methods that use intensity image information such as ours and CF,[20] the results are included for completeness.

MR[17] uses only event information and is hence unable to recover the true intensity information present in the scene. CF[20] does not use any kind of spatial regularization and hence the reconstructed images are noisy and blurry even though it has access to the intensity images. Though the proposed algorithm takes more time to run compared to previous works,[17,20] it is able to produce much better results. The proposed algorithm takes about 2 min to estimate the dense depth maps and about 40 s to render each intermediate frame. However, with recent advances in stereo depth estimation methods, it is expected that in the future, the need for an iterative depth refinement scheme can be eliminated and the output of a state-of-the-art stereo depth estimation algorithm can be directly used. This will greatly reduce the computation time. It is possible to further reduce the computation time for estimating pose by using the Lucas–Kanade inverse compositional method.[35]

5.3 *Robustness to Abrupt Camera Motion*

In the case of abrupt motion of the sensor, the intensity images get blurred and the rate at which events are generated

Fig. 9 Abrupt camera motion: (a) Top row shows the two successive images blended into one where we can see the abrupt camera motion. The bottom row shows an intermediate event frame affected by noise. In (b) and (c), two intermediate reconstructed frames using CF[20] and our proposed algorithm are shown. We can clearly see that the proposed method performs much better even during abrupt camera motions.

becomes high. In this case, the intensity images are deblurred using an existing deblurring technique (in our experiments, we used the one in Ref. 36). These deblurred images are then used as an input to the reconstruction pipeline. Abrupt motion results in a high event rate and also produces many noisy events. These noisy events affect the reconstructions in Ref. 20 as their trust on events increases exponentially with the rise in the event rate. As can be seen in Fig. 9, our method is robust to such abrupt motions as can be seen from the results shown in columns (b) and (c).

6 Conclusion

The strength of texture-rich low-frame-rate intensity frames is combined with high temporal rate event data to obtain temporally dense photorealistic images. This is achieved by warping the low-frame-rate intensity frames from the conventional image sensor to intermediate locations. With extensive experiments, it has been demonstrated that the images reconstructed from the proposed algorithm are photorealistic compared to any of the previous methods. The robustness of our algorithm to abrupt camera motion has also been shown. Currently, the proposed algorithm assumes a static scene. A future direction would be to build a generalized algorithm that can reconstruct photorealistic images for dynamic scenes as well.

References

1. P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128 × 128 120 db 15 μs latency asynchronous temporal contrast vision sensor," *IEEE J. Solid-State Circuits* **43**(2), 566–576 (2008).
2. "Event sensor product description," https://inivation.com/dvs/videos/ (accessed 15 May 2019).
3. H. Kim, S. Leutenegger, and A. J. Davison, "Real-time 3-D reconstruction and 6-dof tracking with an event camera," *Lect. Notes Comput. Sci.* **9910**, 349–364 (2016).
4. H. Rebecq et al., "EVO: a geometric approach to event-based 6-dof parallel tracking and mapping in real time," *IEEE Rob. Autom. Lett.* **2**(2), 593–600 (2017).
5. E. Mueggler et al., "The event-camera dataset and simulator: event-based data for pose estimation, visual odometry, and slam," *Int. J. Rob. Res.* **36**(2), 142–149 (2017).
6. G. Gallego, H. Rebecq, and D. Scaramuzza, "A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 3867–3876 (2018).
7. G. Gallego, M. Gehrig, and D. Scaramuzza, "Focus is all you need: loss functions for event-based vision," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 12280–12289 (2019).
8. A. Nguyen et al., "Real-time 6 dof pose relocalization for event cameras with stacked spatial LSTM networks," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. Workshops* (2019).
9. S. Bryner et al., "Event-based, direct camera tracking from a photometric 3-D map using nonlinear optimization," in *IEEE Int. Conf. Robot. Autom.*, Vol. 2 (2019).
10. A. Z. Zhu et al., "Unsupervised event-based learning of optical flow, depth, and egomotion," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 989–997 (2019).
11. A. Z. Zhu, Y. Chen, and K. Daniilidis, "Realtime time synchronized event-based stereo," *Lect. Notes Comput. Sci.* **11210**, 438–452 (2018).
12. Y. Zhou et al., "Semi-dense 3-D reconstruction with a stereo event camera," *Lect. Notes Comput. Sci.* **11205**, 242–258 (2018).
13. A. Andreopoulos et al., "A low power, high throughput, fully event-based stereo system," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 7532–7542 (2018).
14. M. Cannici et al., "Attention mechanisms for object recognition with event-based cameras," in *IEEE Winter Conf. Appl. Comput. Vision*, IEEE, pp. 1127–1136 (2019).
15. A. Amir et al., "A low power, fully event-based gesture recognition system," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 7243–7252 (2017).
16. I. Alonso and A. C. Murillo, "Ev-SegNet: semantic segmentation for event-based cameras," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. Workshops* (2019).
17. G. Munda, C. Reinbacher, and T. Pock, "Real-time intensity-image reconstruction for event cameras using manifold regularisation," *Int. J. Comput. Vision* **126**(12), 1381–1393 (2018).
18. P. Bardow, A. J. Davison, and S. Leutenegger, "Simultaneous optical flow and intensity estimation from an event camera," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 884–892 (2016).
19. S. Barua, Y. Miyatani, and A. Veeraraghavan, "Direct face detection and video reconstruction from event cameras," in *IEEE Winter Conf. Appl. Comput. Vision*, IEEE, pp. 1–9 (2016).
20. C. Scheerlinck, N. Barnes, and R. Mahony, "Continuous-time intensity estimation using event cameras," *Lect. Notes Comput. Sci.* **11365**, 308–324 (2018).
21. R. Berner et al., "A 240×180 10 mW 12 μs latency sparse-output vision sensor for mobile applications," in *Symp. VLSI Circuits* (2013).
22. H. Rebecq et al., "Events-to-video: bringing modern computer vision to event cameras," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 3857–3866 (2019).
23. G. Gallego et al., "Event-based vision: a survey," arXiv:1904.08405 (2019).
24. Z. Wang et al., "Event-driven video frame synthesis," arXiv:1902.09680 (2019).
25. "Davis 240c hardware description," https://inivation.com/support/hardware/davis240/ (accessed 15 August 2019).
26. D. Sun et al., "PWC-net: CNNS for optical flow using pyramid, warping, and cost volume," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 8934–8943 (2018).
27. R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3-D geometric constraints," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 5667–5675 (2018).
28. D. J. Heeger, "Notes on motion estimation" (1996).
29. J. T. Barron and B. Poole, "The fast bilateral solver," *Lect. Notes Comput. Sci.* **9907**, 617–632 (2016).
30. D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," in *3rd Int. Conf. Learning Representations (ICLR)*, Conference Track Proceedings, San Diego, California (2015).
31. B. Ummenhofer et al., "Demon: depth and motion network for learning monocular stereo," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 5038–5047 (2017).
32. E. Ilg et al., "Flownet 2.0: evolution of optical flow estimation with deep networks," in *IEEE Conf. Comput. Vision and Pattern Recognit.* (2017).
33. J. Xiao, A. Owens, and A. Torralba, "SUN3D: a database of big spaces reconstructed using SfM and object labels," in *Proc. IEEE Int. Conf. Comput. Vision*, pp. 1625–1632 (2013).
34. C. Reinbacher, G. Graber, and T. Pock, "Real-time intensity-image reconstruction for event cameras using manifold regularisation," in *Br. Mach. Vision Conf.* (2016).
35. S. Baker and I. Matthews, "Lucas–Kanade 20 years on: a unifying framework," *Int. J. Comput. Vision* **56**(3), 221–255 (2004).
36. S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 3883–3891 (2017).

Prasan Shedligeri is a PhD student at the Indian Institute of Technology Madras, India. He received his BE degree in electronics and communication from the Visweswaraya Technological University in 2015. His current research interests include computational photography, image processing, and low-level computer vision.

Kaushik Mitra is an assistant professor at the Indian Institute of Technology Madras, India. He received his PhD from the University of Maryland in 2011. His current research interests include computational photography, image processing, optics, and low-level computer vision.