This paper has been accepted for publication in the
IEEE Winter Conference on Applications of Computer Vision (WACV), 2020.

Fast Image Reconstruction with an Event Camera

Cedric Scheerlinck
Australian National University
cedric.scheerlinck@anu.edu.au

Henri Rebecq
University of Zurich
rebecq@ifi.uzh.ch

Daniel Gehrig
University of Zurich
dgehrig@ifi.uzh.ch

Nick Barnes
Australian National University
nick.barnes@anu.edu.au

Robert E. Mahony
Australian National University
robert.mahony@anu.edu.au

Davide Scaramuzza
University of Zurich
sdavide@ifi.uzh.ch

Abstract

*Event cameras are powerful new sensors able to capture high dynamic range with microsecond temporal resolution and no motion blur. Their strength is detecting brightness changes (called events) rather than capturing direct brightness images; however, algorithms can be used to convert events into usable image representations for applications such as classification. Previous works rely on hand-crafted spatial and temporal smoothing techniques to reconstruct images from events. State-of-the-art video reconstruction has recently been achieved using neural networks that are large (10M parameters) and computationally expensive, requiring 30ms for a forward-pass at 640 × 480 resolution on a modern GPU. We propose a novel neural network architecture for video reconstruction from events that is smaller (**38k** vs. 10M parameters) and faster (**10ms** vs. 30ms) than state-of-the-art with minimal impact to performance.*

Videos and Datasets:
https://cedric-scheerlinck.github.io/firenet

1. Introduction

Event cameras [19] are lightweight, fast sensors that have distinct advantages over conventional cameras: high temporal resolution, high dynamic range and no motion blur, making them ideal for robotic applications. Their raw output is a sequence of asynchronous events (discrete pixel-wise changes in brightness) corresponding to changes in scene illumination. To extract useful information from events, *e.g.* optic flow or classification, they are typically converted to an intermediate representation such as a time-surface [6, 42], event image [37], 3D voxel-grid [13] or brightness image [33, 38, 35]. Brightness images, or video, are a useful representation that act as an interface between event cameras and conventional frame-based computer vision. For example, Rebecq *et al*. [33] show that conventional frame-based methods achieve state-of-the-art performance on event reconstructed images for classification and visual inertial odometry compared to dedicated event-based algorithms. Additionally, image reconstruction allows human visualization and interpretation of events, giving us an intuition of the rich information encoded by events.

The introduction of machine learning to event cameras has caused a proliferation of works, achieving state-of-the-art results in optical flow [49, 50], 6-DOF pose relocalization [27], steering prediction [22], classification [13], segmentation [1], image reconstruction [33] and more. These methods typically convert raw events into time-surfaces, event images or voxel-grids to be passed to a convolutional neural network (CNN). Large CNN models can be memory and computationally intensive, consuming power and hampering the low latency of event cameras. This makes it harder to deploy large models on embedded platforms or IoT applications with power and memory constraints, where event cameras are ideal candidates due to their low power and bandwidth consumption. Reducing the model size can improve performance by reducing (i) memory footprint, (ii) FLOPs and power consumption and (iii) latency.

In this work, we introduce FireNet (Fig. 1): a novel neural network architecture that performs fast image reconstruction from events. FireNet is significantly smaller than state-of-the-art (E2VID) [33], requiring fewer parameters (**38k** vs 10M), less memory (**0.16Mb** vs 43Mb) and fewer FLOPs (**12.6G** vs 147.2G), and runs three times faster than E2VID [33] on a modern GPU. FireNet is a fully convolutional network that relies on recurrent connections to build a state over time, allowing a much smaller network that reuses previous computed results, showing exciting potential for very small recurrent networks that run fast.

2. Related Works

Early research into image reconstruction from events did not use machine learning, instead applying SLAM [11, 16, 17, 32], optimization [3], regularization [35, 26], temporal filtering [38, 39] and combining frames with events

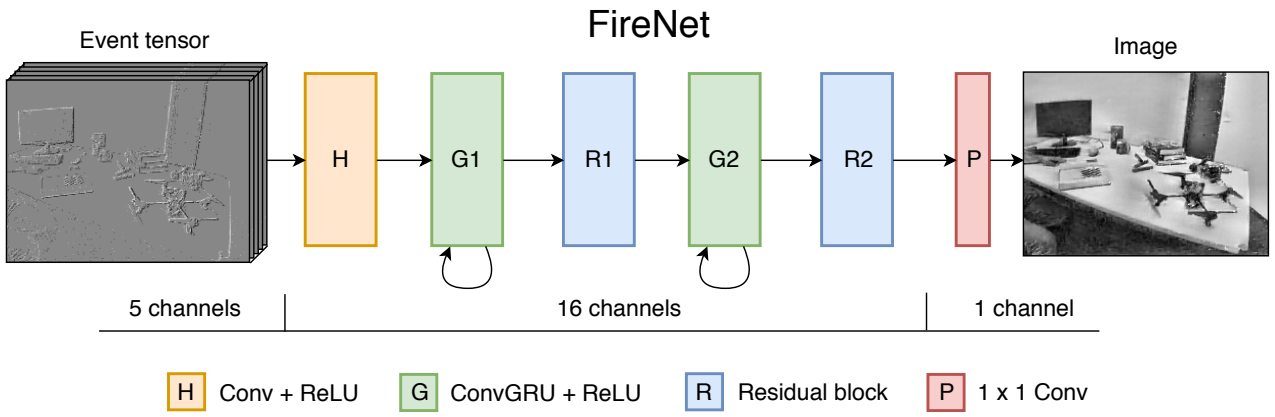


Figure 1. FireNet architecture. The input is an event tensor with 5 temporal bins. The network consists of convolutional layers (H, P), convolutional gated recurrent units (G1, G2) and residual blocks (R1, R2). Every layer uses ReLU activation except the final layer (P).

[8, 29, 28, 38, 41, 21]. SLAM-like approaches aim to simultaneously estimate ego-motion and a brightness gradient map [11, 16, 17, 32], that can be upgraded to intensity via Poisson integration. Another approach is to simultaneously optimize optical flow and intensity [3]. Since events represent brightness changes, they can be integrated as long as noise is filtered out *e.g.* via spatial smoothing based on a time-surface [35, 26] or temporal smoothing [38, 39, 5]. Finally, events can also be considered in conjunction with image frames, either using events to warp images [21, 41], or combining events and frames directly [8, 38, 29, 28].

The first learning approach to image reconstruction from events was proposed by Barua *et al*. [4], who used a simulator to learn a sparse patch-based dictionary to match event patches to gradient patches. They performed image reconstruction via Poisson integration and showed that their approach could be used for face detection. Sparse dictionary learning was also used in [47] to reconstruct images from retinal event trains. Generative adversarial networks were used in [30, 24] to generate realistic images from events. [46] proposed fusing events and frames with a CNN. Rebecq *et al*. [33] showed that a large amount of simulated data could be used to train a network (E2VID) end-to-end to reconstruct high speed, high dynamic range video from events, achieving state-of-the-art results, later improved in [34]. E2VID is a fully convolutional, recurrent UNet architecture inspired by [36, 49].

One potential downside to conventional neural networks is computational cost, putting them at odds with event cameras' natural low-power and low-latency. Spiking neural networks are theoretically more efficient in terms of power consumption and compute time, however, realizing these gains requires specialized algorithms and hardware such as IBM's TrueNorth chip [23]. Other works have proposed modifications to standard network architectures aimed at exploiting the asynchronous/serial nature of event cameras such as asynchronous convolutional neural networks [9] or recursive max pooling [40]. Hardware accelerators [12] and sparse convolutions [14] exploit sparsity in the input and hidden layers of networks, reducing the number of FLOPS and memory footprint by skipping zero (or small) tensor elements.

In this work, we aim to improve computational efficiency by proposing a novel architecture drastically smaller than E2VID[1] [33, 34]. We achieve a $280\times$ reduction in parameters, $10\times$ reduction in FLOPs and $3\times$ speedup compared to E2VID while incurring only minor drop in performance.

3. Method

3.1. Input Representation

The input to our network is a $H \times W \times B$ event tensor $E(x, y, t)$ proposed by [50], where H, W are the sensor height and width and B is the number of temporal bins. The event tensor is populated using trilinear voting (interpolation) where each event (x_i, y_i, t_i, p_i) contributes its polarity to its two closest temporal bins according to:

$$E(x, y, t_n) = \sum_i p_i \max(0, 1 - |t_n - t_i^*|), \qquad (1)$$

$$t_i^* = \frac{(t_i - t_{\min})}{(t_{\max} - t_{\min})}(B - 1)$$

where n is the temporal bin index, p_i is the polarity and t_i^* is the normalized timestamp of the i^{th} event. We use $B = 5$.

At runtime we pass consecutive, non-overlapping event tensors with a fixed number of N events. Thus, $t_{\min}$ and $t_{\max}$ may be different for each event tensor, depending on the timing of events. Our input representation can be interpreted as adaptively rescaling the temporal dimension,

[1] We compare against the improved version of E2VID [34]. Code: https://github.com/uzh-rpg/rpg_e2vid

i.e. the network never sees absolute timestamps, only relative event timings. This scheme can be relaxed *e.g.* having a variable number of events per event tensor. In addition, we normalize the non-zero entries in $E(x, y, t)$ to have zero mean and unit norm, mitigating the impact of unbalanced ON/OFF contrast thresholds, and making the network robust against different magnitudes of contrast threshold.

3.2. Architecture

We use a fully convolutional recurrent neural network (Fig. 1). All layers use single-strided (no downsampling) 3×3 convolutions except the final layer which is 1×1. The head unit (H) consists of a 16-channel convolution ($y = w * x + b$) with ReLU activation. The convolutional gated recurrent units (G1, G2) consist of a 16-channel convolution with ReLU activation followed by a gated recurrent unit as described in [2]. We choose GRUs instead of LSTMs because they have been shown to exhibit similar performance [10] while having less parameters (two gates instead of three). The residual blocks (R1, R2) use 16-channel convolutions with ReLU activation and skip connections as described in [15]. The final prediction layer (P) is a 1×1 single-channel convolution. The output is one image per input event tensor. Table 1 shows key differences between our network and E2VID [33].

3.3. Training

To make a fair comparison to E2VID, the exact same training and validation data was used. The data was generated by the event simulator *ESIM* [31], and consists of 1,000 sequences of 2 seconds each (950 training, 50 validation). MS-COCO images [20] were mapped to a 3D plane and random 6-DOF (simulated) camera motions were used to trigger events. To simulate contrast threshold mismatch and refractory, contrast threshold values (ON/OFF) for each sequence were drawn from a normal distribution with mean $\mu = 0.18$ and standard deviation $\sigma = 0.03$ and a refractory period of 1ms was applied after each event.

As in [34], we used both (i) a *reconstruction loss* that measures the difference between the reconstruction and groundtruth image, and (ii) a *temporal loss* that penalizes differences between consecutive reconstructed images. We used Perceptual Similarity (LPIPS) [48] to a groundtruth image as the reconstruction loss $\mathcal{L}_k^R = d(\hat{\mathcal{I}}_k, \mathcal{I}_k)$, where d is the LPIPS distance function, $\hat{\mathcal{I}}_k$ is the k^{th} reconstructed image and $\mathcal{I}_k$ is the groundtruth image. The groundtruth image was selected by matching its timestamp to the latest event in our event tensor, thus, discouraging the network from predicting images in the past. We used the *temporal consistency loss* described in [34] that aligns two successive reconstructed images based on the optical flow between them and measures a photometric error $\mathcal{L}_k^{TC} = c(\hat{\mathcal{I}}_{k-1}, \hat{\mathcal{I}}_k)$, where c is the temporal consistency function.

Table 1. Network overview. Compared to E2VID [33, 34], our network has $280\times$ fewer parameters, consuming only 0.37% of the memory.

	E2VID	Ours
No. parameters (k)	10700	38
Memory (Mb)	43	0.16
Downsampling	yes	no
Recurrent units	LSTM	GRU
Max. kernel size	5×5	3×3

The final loss is a weighted sum of reconstruction and temporal losses over L consecutive images

$$\mathcal{L} = \sum_{k=0}^{L} \mathcal{L}_k^R + \lambda_T C \sum_{k=L_0}^{L} \mathcal{L}_k^{TC}, \tag{2}$$

where $L = 20$, $\lambda_{TC} = 2$ and $L_0 = 10$. We used the ADAM optimizer [18] with default parameters, learning rate 1e-4, and trained for 1000 epochs.

4. Results

4.1. Overview

Our network is $280\times$ smaller than E2VID [33, 34] (Tab. 1) with only 38k parameters (0.36%) and consuming only 160kb of memory (0.37%). This gives us a $3\times$ speedup on GPU, $4\times$ on CPU and $10\times$ reduction in the number of FLOPs (Tab. 2). Our accuracy on the event camera dataset [25] is comparable to E2VID (Tab. 3). Qualitative comparison confirms that our reconstructed images are of a similar quality to E2VID (Fig. 3), though in some challenging scenarios we perform slightly worse than E2VID (Fig. 6). We compare against improved E2VID [34] for all experiments.

4.2. Computational Performance

Table 2 compares the computational cost of our method against E2VID. We used an NVIDIA Titan Xp GPU and an Intel 3.20 GHz i7-6900K CPU for all experiments. To evaluate computational cost, we measured the compute time of a forward-pass through the network at various image sensor resolutions on both GPU and CPU. We selected resolutions of common event cameras such as the DAVIS240 [7], DAVIS346 [44], Samsung, Prophesee and CeleX sensors. We also report the number of floating point operations (FLOPs) at each resolution, which is related to power consumption. Note that E2VID and our method are agnostic to the number of events per forward-pass, that is, a forward-pass will take the same amount of time if the input event tensor contains zero or one million events. Our method performs three times faster than E2VID on GPU, and up to

Table 2. Computational cost. We report inference time on GPU and CPU, and the number of FLOPs for a single forward-pass at common sensor resolutions.

Resolution	GPU (ms)		CPU (ms)		FLOPs (G)	
	E2VID	Ours	E2VID	Ours	E2VID	Ours
240×180	5.52	**1.89**	84.98	**22.86**	21.2	**1.8**
346×260	10.17	**3.22**	183.79	**40.96**	44.5	**3.7**
640×480	30.88	**10.15**	687.10	**264.39**	147.2	**12.6**
1280×720	93.34	**31.01**	2235.60	**1039.49**	441.7	**37.8**

four times faster on CPU, requiring less than one tenth the number of FLOPs.

4.3. Accuracy

We evaluated the accuracy of our method against DAVIS240C [7] frames in the event camera dataset [25] (Tab. 3, Fig. 3), and compared against several competitive methods: high-pass filter (HF) [38], manifold regularization (MR) [35] and E2VID [33, 34]. We discarded sections with poor frame quality, leaving seven sequences with 1,670 groundtruth frames. Given a pair of successive image frames ($\sim$20Hz for the event camera dataset [25]), we took all events in between the frames and constructed an event tensor as specified in 3.1, creating a sequence of event tensors for each dataset sequence. We reconstructed a video for each sequence and compared against groundtruth frames. For all methods, we matched the timestamp of the latest event used in each reconstructed image to the nearest groundtruth frame with a tolerance of 1ms. For HF and MR, we used code provided by the authors and manually tuned the parameters to get best results possible. For HF we additionally applied a 5×5 bilateral filter with $\sigma = 25$ to smooth high-frequency noise, which improved results of HF in all metrics. To ensure the intensity values lay within a similar range, we applied local histogram normalization to both the output and groundtruth frames. We compared reconstructed images against groundtruth frames using the metrics: mean squared error (MSE), structural similarity (SSIM) [45] and perceptual similarity (LPIPS) [48].

Table 3 shows that our method performs favorably compared to hand-crafted methods HF and MR. We achieve a 40% decrease in mean squared error, 20% increase in structural similarity and 20% improvement in perceptual similarity. At the same time, we quantitatively match performance of E2VID for all metrics on the event camera dataset [25], though there are minor qualitative defects for some challenging scenarios (see Fig. 6).

4.4. Qualitative Evaluation

Figure 2 shows qualitative comparison to [24]. [24] used a generative adversarial network (GAN), trained on a mixture of real and synthetic data. While our network was trained exclusively on synthetic data, GAN was trained on data from the event camera dataset [25] (rows 2, 3) (*i.e*. the network has seen these sequences at train time), thus, we did not include quantitative comparison out of fairness. Images reconstructed with GAN appear less sharp, and contain artefacts in textureless regions of the scene (where there are no events). In challenging sequences, such as Sun (from [38]), there are many noise events that translate into artefacts for both GAN and ours, however, our method appears less impacted. Note that both methods appear to contain more information than the oversaturated DAVIS frame.

GAN Ours DAVIS frame

Figure 2. Left: GAN [24] appears less sharp and exhibits artefacts in textureless regions (note: rows 2, 3 were used in training). Middle: our method looks cleaner but still suffers from very noisy events *e.g*. Sun (top). Right: the DAVIS frame has lower dynamic range than events, evident in the reconstructions of both methods.

Figure 4 shows results on the *High Speed and HDR Dataset*[2] [34], demonstrating that our network can generalize to a different sensor (Samsung DVS Gen3 [43]). Each sequence was reconstructed using 50k events per input tensor (higher due to the higher sensor resolution: 640×480 vs. 240×180). Local histogram equalization was applied to improve the visual contrast of the images.

[2] Available at: http://rpg.ifi.uzh.ch/E2VID.html

Table 3. Comparison to state-of-the-art image reconstruction methods on the Event Camera Dataset [25].

Dataset	MSE				SSIM				LPIPS			
	HF	MR	E2VID	Ours	HF	MR	E2VID	Ours	HF	MR	E2VID	Ours
dynamic_6dof	0.10	**0.05**	0.14	0.12	0.39	**0.52**	0.46	0.47	0.54	0.50	0.46	**0.44**
boxes_6dof	0.08	0.10	**0.04**	0.04	0.49	0.45	0.62	**0.64**	0.50	0.53	0.38	**0.37**
poster_6dof	0.07	0.05	0.06	**0.04**	0.49	0.54	0.62	**0.65**	0.45	0.52	0.35	**0.34**
shapes_6dof	0.09	0.19	0.04	**0.02**	0.50	0.51	**0.80**	0.79	0.61	0.64	0.47	**0.46**
office_zigzag	0.09	0.09	**0.03**	0.04	0.38	0.45	**0.54**	0.54	0.54	0.50	0.41	**0.40**
slider_depth	0.06	0.07	0.05	**0.05**	0.50	0.50	0.58	**0.59**	0.50	0.55	0.44	**0.41**
calibration	0.09	0.07	**0.02**	0.04	0.48	0.54	**0.70**	0.66	0.48	0.47	**0.36**	0.37
Mean	0.08	0.09	0.05	**0.05**	0.46	0.50	0.62	**0.62**	0.52	0.53	0.41	**0.40**

(a) HF (b) MR (c) E2VID (d) **Ours** (e) Ground truth

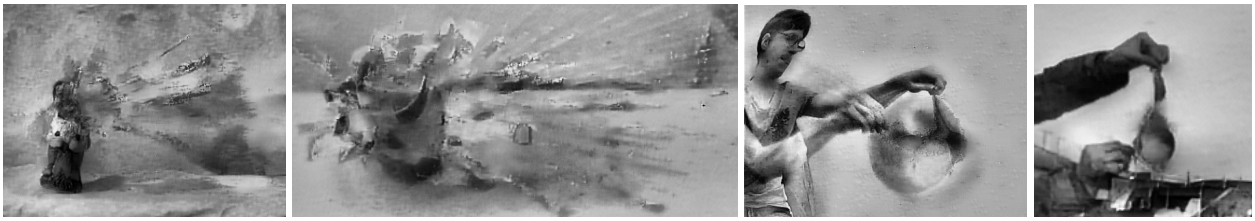
Figure 3. Qualitative comparison against state-of-the-art methods on the event camera dataset [25]. Our method performs comparably to E2VID [33, 34], and produces higher quality images than HF [38] and MR [35].

4.5. Recurrent Connection Ablation

Recurrent connections in a network give it the ability to build a hidden state $h(t)$ that can be maintained and improved over time. Given temporal sequences of training data, the network learns some form of temporal integration, re-using previous computed results. Figure 5 shows that E2VID can reconstruct images from batches of 10k events per input tensor without recurrent connections. We believe this is because its large size and receptive field allows it to spatially propagate information from events (edges). However, smaller models such as ours cannot reliably reconstruct images without recurrent connections because of a limited receptive field (maximum 15×15 in our case). Because our network is fully convolutional, pixels in the prediction layer can only see events within their receptive field. Thus, we can conclude that recurrent connections are the primary driver enabling a smaller network.

4.6. Limitations

In challenging scenarios such as very fast motions, and initialization, FireNet exhibits defects such as smearing, or incomplete reconstruction in places with no events. Figure 6 shows selected challenging scenes where FireNet artefacts are apparent, while E2VID [34] typically does a better job. To highlight smearing artefacts we used a fixed time-window of 50ms per input event tensor for both methods. Using a smaller time-window or fixed number of events per input tensor may decrease smearing for fast motions.



(a) Gnome (b) Mug (c) Air balloon (d) Water balloon

Figure 4. High speed phenomenon from the *High Speed and HDR Dataset* [34]. (a) and (b) are moments after a bullet impact from a gun. (c) and (d) are moments after an air and water balloon are popped. The water (d) initially retains the original shape of the balloon as it falls.

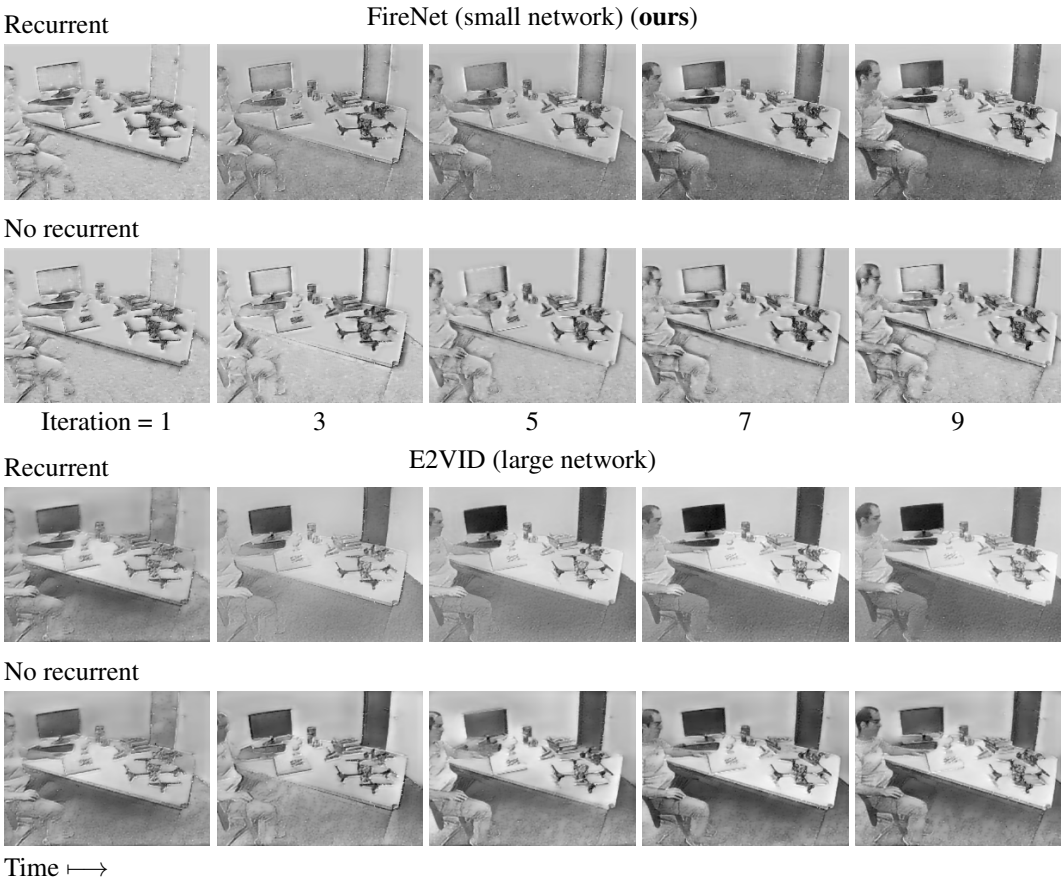


Figure 5. Recurrent connection ablation study. Image is initialized at zero. Top: small network (ours) relies on recurrent connection to build hidden state over time. When recurrent connection is disabled (second row), the network fails, indicating that recurrent connections are a key component. Bottom: While recurrent connections help stabilize video (third row), large networks (E2VID [33, 34]) are still able to reconstruct images without recurrent connection (fourth row).

5. Conclusion

We have presented FireNet, a fast, lightweight CNN that reconstructs images directly from events. Our method performs almost as well as state-of-the-art (E2VID [33, 34]) at a fraction of the computational cost, yielding a $3\times$ speedup, $10\times$ reduction in FLOPs with $280\times$ fewer parameters. We showed that recurrent connections are a key component enabling smaller networks because it allows them to build and improve a hidden state over time, re-using previous computed results. We believe FireNet shows exciting potential for fast, lightweight recurrent networks for event processing, and that the reconstructed images reveal an exciting depth of information that can be unlocked from events with a surprisingly small network.

Acknowledgments

This work was supported by (i) the Australian Government Research Training Program Scholarship (ii) the Australian Research Council through the "Australian Centre of Excellence for Robotic Vision" under Grant CE140100016 (iii) the Swiss Government

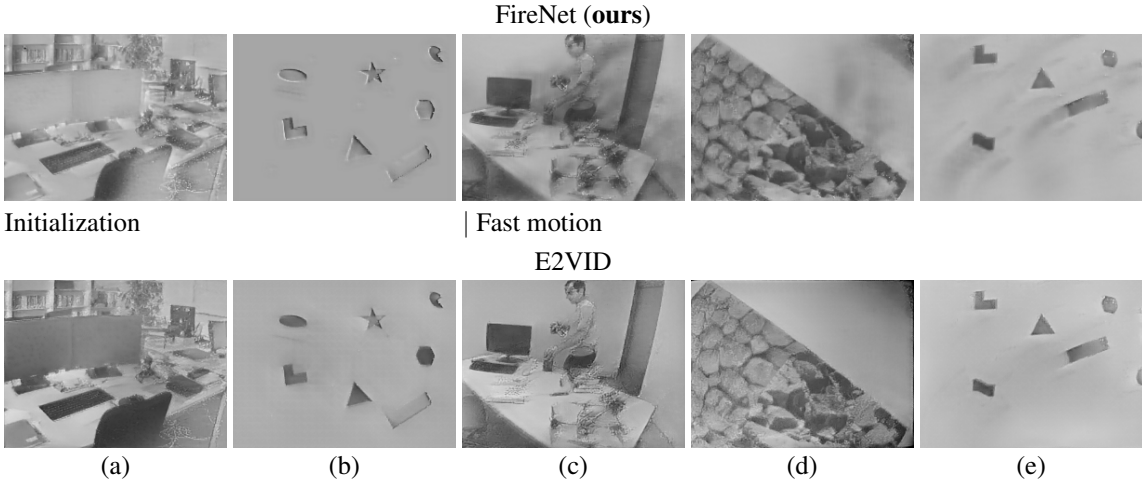


Figure 6. (a) and (b) show the network output at initialization. FireNet (ours) does not "fill in" the image as quickly as E2VID [34, 33]. (c)-(e) are captured when the camera is undergoing fast motion. FireNet exhibits more smearing artefacts than E2VID.

Excellence Scholarship (iv) the Swiss National Center of Competence Research Robotics (NCCR) (v) Qualcomm (through the Qualcomm Innovation Fellowship Award 2018) (vi) the SNSF-ERC Starting Grant.

References

[1] I. Alonso and A. C. Murillo. EV-SegNet: Semantic segmentation for event-based cameras. 2019.

[2] N. Ballas, L. Yao, C. Pal, and A. Courville. Delving deeper into convolutional networks for learning video representations. In *Int. Conf. Learn. Representations (ICLR)*, 2016.

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

[4] S. Barua, Y. Miyatani, and A. Veeraraghavan. Direct face detection and video reconstruction from event cameras. In *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2016.

[5] A. N. Belbachir, S. Schraml, M. Mayerhofer, and M. Hofstaetter. A novel HDR depth camera for real-time 3D 360-degree panoramic vision. In *IEEE Conf. Comput. Vis. Pattern Recog. Workshops (CVPRW)*, 2014.

[6] R. Benosman, C. Clercq, X. Lagorce, S.-H. Ieng, and C. Bartolozzi. Event-based visual flow. *IEEE Trans. Neural Netw. Learn. Syst.*, 25(2):407–417, 2014.

[7] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck. A 240x180 130dB 3us latency global shutter spatiotemporal vision sensor. *IEEE J. Solid-State Circuits*, 49(10):2333–2341, 2014.

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

[9] M. Cannici, M. Ciccone, A. Romanoni, and M. Matteucci. Asynchronous convolutional networks for object detection in neuromorphic cameras. In *IEEE Conf. Comput. Vis. Pattern Recog. Workshops (CVPRW)*, 2019.

[10] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NeurIPS Workshop on Deep Learning*, 2014.

[11] M. Cook, L. Gugelmann, F. Jug, C. Krautz, and A. Steger. Interacting maps for fast visual interpretation. In *Int. Joint Conf. Neural Netw. (IJCNN)*, pages 770–776, 2011.

[12] C. Gao, D. Neil, E. Ceolini, S.-C. Liu, and T. Delbruck. DeltaRNN: A power-efficient recurrent neural network accelerator. In *ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA)*, 2018.

[13] D. Gehrig, A. Loquercio, K. G. Derpanis, and D. Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *Int. Conf. Comput. Vis. (ICCV)*, 2019.

[14] B. Graham, M. Engelcke, and L. van der Maaten. 3D semantic segmentation with submanifold sparse convolutional networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 9224–9232, 2018.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 770–778, 2016.

[16] H. Kim, A. Handa, R. Benosman, S.-H. Ieng, and A. J. Davison. Simultaneous mosaicing and tracking with an event camera. In *British Mach. Vis. Conf. (BMVC)*, 2014.

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

[18] D. P. Kingma and J. L. Ba. Adam: A method for stochastic optimization. *Int. Conf. Learn. Representations (ICLR)*, 2015.

[19] 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.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 740–755. 2014.

[21] H.-C. Liu, F.-L. Zhang, D. Marshall, L. Shi, and S.-M. Hu. High-speed video generation with an event camera. *The Visual Computer*, 33(6-8):749–759, June 2017.

[22] A. I. Maqueda, A. Loquercio, G. Gallego, N. García, and D. Scaramuzza. Event-based vision meets deep learning on steering prediction for self-driving cars. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5419–5427, 2018.

[23] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197):668–673, 2014.

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

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

[26] G. Munda, C. Reinbacher, and T. Pock. Real-time intensity-image reconstruction for event cameras using manifold regularisation. *Int. J. Comput. Vis.*, 126(12):1381–1393, July 2018.

[27] A. Nguyen, T. Do, D. G. Caldwell, and N. G. Tsagarakis. Real-time 6DOF pose relocalization for event cameras with stacked spatial LSTM networks. In *IEEE Conf. Comput. Vis. Pattern Recog. Workshops (CVPRW)*, 2019.

[28] L. Pan, R. I. Hartley, C. Scheerlinck, M. Liu, X. Yu, and Y. Dai. High frame rate video reconstruction based on an event camera. *arXiv e-prints*, 2019.

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

[30] S. Pini, G. Borghi, R. Vezzani, R. C. U. of Modena, and R. Emilia. Learn to see by events: Color frame synthesis from event and RGB cameras. *Int. Joint Conf. Comput. Vis., Image and Comput. Graph. Theory and Appl.*, 2020.

[31] H. Rebecq, D. Gehrig, and D. Scaramuzza. ESIM: an open event camera simulator. In *Conf. on Robot. Learning (CoRL)*, 2018.

[32] H. Rebecq, T. Horstschäfer, G. Gallego, and D. Scaramuzza. EVO: A geometric approach to event-based 6-DOF parallel tracking and mapping in real-time. *IEEE Robot. Autom. Lett.*, 2(2):593–600, 2017.

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

[34] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.

[35] C. Reinbacher, G. Graber, and T. Pock. Real-time intensity-image reconstruction for event cameras using manifold regularisation. In *British Mach. Vis. Conf. (BMVC)*, 2016.

[36] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015.

[37] A. Rosinol Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza. Ultimate SLAM? combining events, images, and IMU for robust visual SLAM in HDR and high speed scenarios. *IEEE Robot. Autom. Lett.*, 3(2):994–1001, Apr. 2018.

[38] C. Scheerlinck, N. Barnes, and R. Mahony. Continuous-time intensity estimation using event cameras. In *Asian Conf. Comput. Vis. (ACCV)*, 2018.

[39] C. Scheerlinck, N. Barnes, and R. Mahony. Asynchronous spatial image convolutions for event cameras. *IEEE Robot. Autom. Lett.*, 4(2):816–822, Apr. 2019.

[40] Y. Sekikawa, K. Hara, and H. Saito. EventNet: Asynchronous recursive event processing. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019.

[41] P. Shedligeri and K. Mitra. Photorealistic image reconstruction from hybrid intensity and event-based sensor. *J. Electron. Imaging*, 28(06):1, Dec. 2019.

[42] A. Sironi, M. Brambilla, N. Bourdis, X. Lagorce, and R. Benosman. HATS: Histograms of averaged time surfaces for robust event-based object classification. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 1731–1740, 2018.

[43] B. Son, Y. Suh, S. Kim, H. Jung, J.-S. Kim, C. Shin, K. Park, K. Lee, J. Park, J. Woo, Y. Roh, H. Lee, Y. Wang, I. Ovsiannikov, and H. Ryu. A 640x480 dynamic vision sensor with a 9um pixel and 300Meps address-event representation. In *IEEE Intl. Solid-State Circuits Conf. (ISSCC)*, 2017.

[44] G. Taverni, D. P. Moeys, C. Li, C. Cavaco, V. Motsnyi, D. S. S. Bello, and T. Delbruck. Front and back illuminated Dynamic and Active Pixel Vision Sensors comparison. *IEEE Trans. Circuits Syst. II*, 65(5):677–681, 2018.

[45] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, Apr. 2004.

[46] Z. W. Wang, W. Jiang, A. Katsaggelos, and O. Cossairt. Event-driven video frame synthesis. In *Int. Conf. Comput. Vis. Workshops (ICCVW)*, 2019.

[47] Y. Watkins, A. Thresher, D. Mascarenas, and G. T. Kenyon. Sparse coding enables the reconstruction of high-fidelity images and video from retinal spike trains. In *Proceedings of the International Conference on Neuromorphic Systems*, 2018.

[48] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018.

[49] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis. EV-FlowNet: Self-supervised optical flow estimation for event-based cameras. In *Robotics: Science and Systems (RSS)*, 2018.

[50] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019.