Supplementary Material – End-to-End Learning of Representations for Asynchronous Event-Based Data

1. Appendix

We encourage the reader to watch the supplementary video at https://www.youtube.com/watch?v=bQtSx59GXRY for an introduction to the event camera and qualitative results of our approach. In this section, we provide additional details about the network architecture used for our experiments, as well as supplementary results for object recognition and optical flow prediction.

1.1. Network Architecture

For all our classification experiments, we used an off-the-shelf ResNet-34 [1] architecture for inference with weights pretrained on RGB image-based ImageNet [5]. We then substitute the first and last layer of the pre-trained network with new weights (randomly initialized) to accommodate the difference in input channels (from the difference in representation) and output channels (for the difference in task).

For the optical flow experiments, we use the off-the-shelf U-Net architecture [4] for inference, adapting its input layer to the number of channels of each representation.

Learned Kernel Functions As discussed in Sec. 3.3 in the main manuscript, we used a two-layer multi-layer perceptron (MLP) to learn the kernel function to convolve the event measurement field, defined in (4). The two hidden layer have both 30 nodes, with Leaky ReLU as activation function (leak $= 0.1$) to encourage better gradient flow. To give all image locations the same importance, we designed the kernel to be translation invariant. Thus, for an event occurring at time t_k the MLP has a one-dimensional input $\delta t = t^*_k - t_n$ and a single output $k(t^*_k - t_n)$ with normalized time $t^*_k = \frac{t_k}{\Delta t}$ and Δt denoting the time window of the events. The contribution of a single event to the sum in (6) is computed for every grid position t_n for $n = 0, 1, ..., B-1$ where B is the number of temporal discretization bins. The weights of the MLP were initialized with the trilinear voting kernel $k(x, y, t) = \delta(x, y) \max(0, 1 - |\frac{t}{\Delta t}|)$ [2], since this proved to facilitate convergence in our experiments. Fig. 1 shows an illustration of the learned kernels as a function of time. Interestingly, the learned kernels show some interesting behavior, when compared against the trilinear voting kernel, on which they were initialized. For classification (Fig. 1, left), the kernel seems to increase the event influence to the past, in a causual fashion: indeed, enough evidence has to be accumulated to produce a classification label. In contrast, for optical flow prediction (Fig. 1, right), the learned kernel increases in magnitude, but not significantly in the time range, with respect to the trilinear kernel. This is probably due to the fact that optical flow is a more 'local' task with respect to classification, and therefore less temporal information is required.

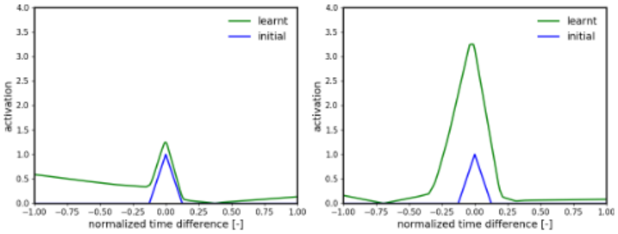


Figure 1. Kernel function learned for classification in the N-Cars dataset (left) and for optical flow prediction (right).

1.2. Ablation Studies and Qualitative Results

1.2.1 Classification

For the classification task, we investigated the relation between the number of temporal discretization bins, B, *i.e.*, channels, of the event spike tensor (EST) and the network performance. We quantativively evaluated this effect on the N-Cars [6] and N-Caltech101 [3] datasets. More specifically, we trained four networks, each using the learned EST with $B = 2, 4, 9, 16$ and timestamp measurements, since this representation achieved the highest classification scores. The final input representations have 4, 8, 18, and 32 channels since we stack the polarity dimension along the temporal dimension. The results for this experiment are summarized in Tab. 1 and example classifications for the N-Cars and N-Caltech101 dataset are provided in Figs. 2 and 3.

For both datasets, we observe a very similar trend in the dependency of classification accuracy to temporal discretization: performance appears to increase with finer discretization, *i.e.*, with a larger number of channels. However,

Temporal Bins	N-Cars	N-Caltech101
2	0.908	0.792
4	0.912	0.816
9	**0.925**	0.817
16	0.923	**0.837**

Table 1. Classification accuracy on N-Cars [6] and N-Caltech101 [3] for input representations based on the event spike tensor (EST). Four variations of the EST were tested, varying the number of temporal bins between $2, 4, 9$ and 16. The best representations are highlighted in bold.

for the N-Cars dataset performance plateaus after $B = 9$ channels, while for the N-Caltech dataset performance continues to increase with a larger number of channels. This difference can be explained by the different qualities of the datasets. While the N-Cars dataset features samples taken in an outdoor environment (Fig. 2), the N-Caltech101 samples were taken in controlled, constant lighting conditions and with consistent camera motion. This leads to higher quality samples in the N-Caltech101 dataset (Fig. 3), while the samples in N-Cars are frequently corrupted by noise (Fig. 2 (a-d)). In low noise conditions (Fig. 2 (a)) classification accuracy is very high (99%). However, as the signal decreases due to the lack of motions (Fig. 2 (b-d)) the classification accuracy decreases rapidly. Increasing the number of temporal bins further dilutes the signal present in the event stream, resulting in noisy channels (Fig. 2 (c)), which impacts performance negatively. In addition, more input channels results in higher the memory and computational costs of the network. Therefore to trading-off performance for computational accuracy, we use $B = 9$ in all our classification experiments.

1.2.2 Optical Flow

In this section, we ablate two features of the representations used for optical flow prediction: (i) the measurement function f (defined in (5)), and (ii) the number of temporal discretization bins, B. We use the Multi Vehicle Stereo Event Camera (MVSEC) dataset [7] for quantitative evaluation.

Tab. 2 shows the performance of our candidate measurement functions, *i.e.*, *polarity*, *event count*, and event *timestamp*, for the generation of the representations (see (5)). While it would be possible to learn the measurement function together with the kernel, in our experiments we have considered this function to be fixed. This heuristic proved to speed-up convergence of our models, while decreasing the computational costs at training and inference time.

In Tab. 2 it can be observed that the event timestamp yields the highest accuracy among the measurement functions. This is indeed very intuitive since, while polarity and event count information is contained in the EST, the timestamp information is partially lost due to discretization. Adding it back in the measurements gives the EST the least amount of information lost with respect to the original event point set, therefore maximizing the performance of end-to-end learning.

To understand the role that the number of temporal bins plays, we choose the best event representation for this task, the EST with timestamp measurements, and vary the number of temporal bins from $B = 2, 4, 9, 16$. The average endpoint errors and outlier ratios are reported in Tab. 3.

As with the classification task (Sec. 1.2.1), we observe a trade-off between using too few channels and too many. Since MVSEC records natural outdoor scenes, event measurements are corrupted by significant noise. As we increase the number of channels, the signal-to-noise ratio in the individual channels drops, leading to less accurate optical flow estimates. In contrast, decreasing the number of channels also has adverse effects, as this removes valuable information from the event stream due to temporal aliasing effects. Therefore, a compromise must be made between high and low channel numbers. In the experiments reported in the paper we chose a channel number of nine, as this presents a good compromise.

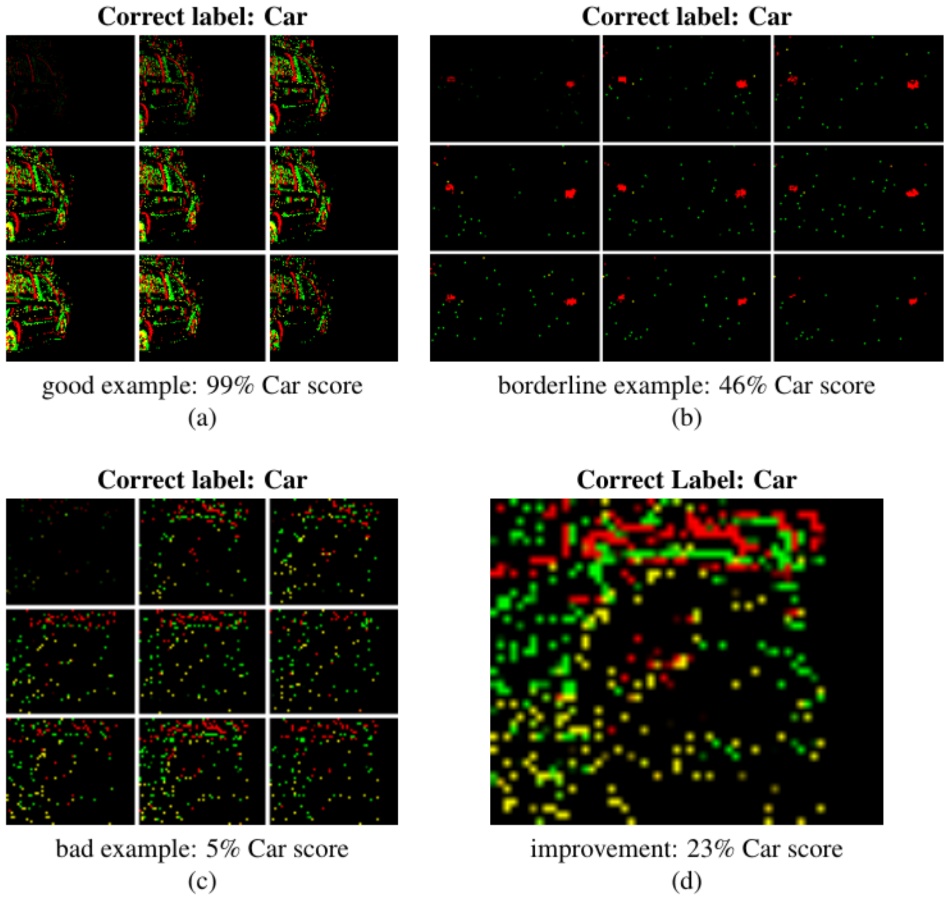
In conclusion, we encourage the reader to watch the supplementary video to see the qualitative results of our method on optical flow prediction. We have observed that, despite the application environment and illumination conditions, our method generates predictions which are not only accurate, but also temporally consistent without any post-processing.

Representation	**Measurement**	**Kernel**	indoor_flying1		indoor_flying2		indoor_flying3	
			AEE	% Outlier	AEE	% Outlier	AEE	% Outlier
Event Frame	polarity	trilinear	1.21	4.19	2.04	20.6	1.83	16.6
Two-Channel Image			1.31	4.75	2.05	23.2	1.83	11.4
Voxel Grid			**0.96**	1.47	1.65	14.6	1.45	11.4
EST (Ours)			1.01	1.59	1.79	16.7	1.57	13.8
Event Frame	count	trilinear	1.25	3.91	2.11	22.9	1.85	17.1
Two-Channel Image			1.21	4.49	2.03	22.8	1.84	17.7
Voxel Grid			0.97	1.33	1.66	14.7	1.46	12.1
EST (Ours)			1.03	2.00	1.78	16.5	1.56	13.2
Event Frame	time stamps	trilinear	1.17	2.44	1.93	18.9	1.74	15.6
Two-Channel Image			1.17	1.50	1.97	14.9	1.78	11.7
Voxel Grid			0.98	1.20	1.70	14.3	1.50	12.0
EST (Ours)			1.00	1.35	1.71	11.4	1.51	8.29
EST (Ours)	time stamps	alpha	1.03	1.34	1.52	11.7	1.41	8.32
		exponential	**0.96**	1.27	1.58	10.5	**1.40**	9.44
		learnt	0.97	**0.91**	**1.38**	**8.20**	1.43	**6.47**

Table 2. Average end-point error (AEE) and % of outliers evaluation on the MVSEC datasets. Ablation of different measurement functions for the event spike tensor. The best candidates are highlighted in bold.

Temporal Bins	*indoor_flying1*		*indoor_flying2*		*indoor_flying3*	
	AEE	% Outlier	AEE	% Outlier	AEE	% Outlier
2	0.97	0.98	1.45	8.86	1.37	6.66
4	**0.96**	1.13	1.42	8.86	1.35	**5.98**
9	0.97	**0.91**	**1.38**	**8.20**	1.43	6.47
16	0.95	1.56	1.39	8.58	**1.34**	6.82

Table 3. Average end-point error (AEE) and % of outliers for optical flow predictions on the MVSEC dataset [7]. Four event representations based on the voxel grid were tested with 2, 4, 9 and 16 temporal bins. The best representation is highlighted in bold.



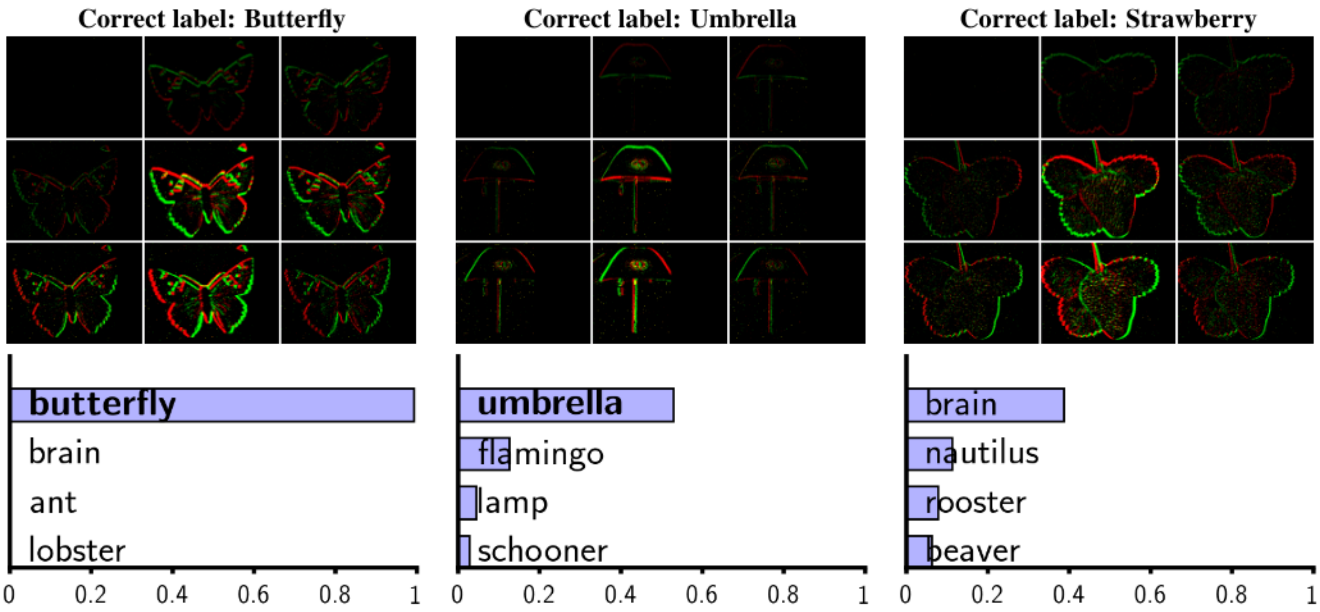
good example: 99% Car score
(a)

borderline example: 46% Car score
(b)

bad example: 5% Car score
(c)

improvement: 23% Car score
(d)

Figure 2. Visualization of input representations derived from samples from the N-Cars dataset [6] (a and b) show the event spike tensor (EST) representation with time measurements, which achieved the highest classification score on N-Cars, while (d) shows the two-channel image of sample (c) for comparison. The EST consists of 18 channels, where the first nine are filled with events of positive polarity and the last nine are filled with negative polarity. The images show the nine temporal bins of the tensor with positive events in red and negative events in green. In good conditions (a) the classifier has high confidence in the car prediction. However, when there are less events due to the lack of motion (b and c) the uncertainty rises leading to predictions close to random (50%). In (b) the classifier sees the headlights of the car (red dots) but may still be unsure. In (c) the classifier sees only noise due to the high temporal resolution, likely attributing presence of noise to no motion. When we aggregate the noise (d) into the Two-Channel Image we see a more distinct pattern emerge, leading to higher classification confidence.



(a) (b) (c)

Figure 3. Visualization of the event spike tensor (EST) representations derived from samples from the N-Caltech101 dataset [3]. The EST consists of 18 channels, where the first nine are filled with events of positive polarity and the last 9 are filled with negative polarity. The figures show the nine temporal bins of the tensor with positive events in red and negative events in green. We see that compared to N-Cars [6] the event stream of this dataset is much cleaner and with much less noise. This is because the dataset was recorded in a controlled environment, by positioning an event camera toward an image projected on a screen. (a) and (b) correspond to correct predictions and (c) an incorrect one.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 770–778, June 2016. 1

[2] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Conf. Neural Inf. Process. Syst. (NIPS)*, pages 2017–2025, 2015. 1

[3] Garrick Orchard, Ajinkya Jayawant, Gregory K. Cohen, and Nitish Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Front. Neurosci.*, 9:437, 2015. 1, 2, 5

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, 2015. 1

[5] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Fei-Fei Li. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, Apr. 2015. 1

[6] Amos Sironi, Manuele Brambilla, Nicolas Bourdis, Xavier Lagorce, and Ryad 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. 1, 2, 4, 5

[7] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. EV-FlowNet: Self-supervised optical flow estimation for event-based cameras. In *Robotics: Science and Systems (RSS)*, 2018. 2, 3