Event-Based Cameras for Autonomous Systems

Daniel Dauner

Abstract

Event cameras are novel sensors that output a stream of asynchronous per-pixel brightness changes called 'events' rather than capturing brightness images. They offer outstanding performance in capturing high-speed motion and high dynamic range scenarios where traditional cameras are prone to fail. Particularly autonomous systems can benefit from event cameras by acquiring more robust visual information. Although the events theoretically encode a complete visual signal, event streams are incompatible with conventional computer vision techniques. Recent work has demonstrated the qualitative reconstruction of intensity images from event streams. This approach acts as a bridge between event-based vision and conventional computer vision. This report aims to introduce the field of event-vision, present state-of-the-art image reconstruction techniques, and examine their application for autonomous systems.

1. Introduction

Event cameras are bio-inspired vision sensors that operate fundamentally differently from conventional cameras. Instead of acquiring a sequence of images at a fixed frame rate, event cameras record pixel-wise *intensity changes* (called *"events"*) asynchronously at the time they occur [8]. The output is a stream of events, each encoding the time, location, and polarity of the brightness change (as depicted in Figure 1). Event cameras offer several advantages over traditional cameras: high temporal resolution, high dynamic range, and low power and bandwidth requirements.

Therefore, event sensors are fast (in the order of µs), lightweight, and robust alternatives for acquiring visual information. Event cameras are advantageous where traditional cameras have shortcomings, e.g., in fast motion scenarios and under challenging illumination conditions. Conventional cameras constitute a primary sensor for many autonomous systems, together with lidar and radar sensors [6]. However, systems like drones and self-driving cars require reliable vision to operate safely in their environment. Thus, event cameras could serve as an alternative or complementary vision sensor for autonomous systems.

Recent work showed that event cameras improve end-to-end steering prediction of autonomous systems, such as self-driving cars [4, 11, 19] and robots [20]. While the results indicate the value of event cameras, more advanced paradigms for autonomous cars and robots depend on various perception tasks from computer vision [12, 44]. Since the output of event cameras is an asynchronous stream of events (instead of conventional image frames), established methods from computer vision are not directly applicable. When using specialized algorithms, event cameras have shown remarkable performance in tasks such as optical flow [1, 3, 48], feature tracking [16, 46], and visual odometry [15, 28, 29]. However, such methods are highly task-specific and cannot offer a broadly applicable framework for processing event data in diverse tasks.

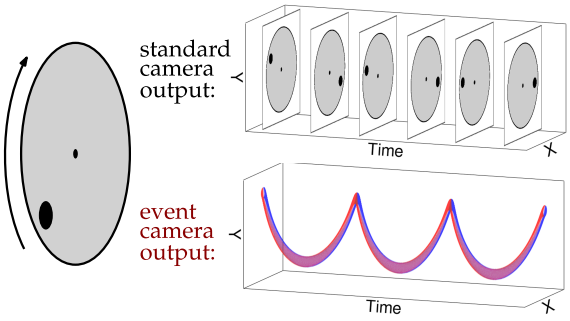


Figure 1. The illustration compares standard cameras to event cameras when recording a rotating circle with a black disk. A conventional camera captures frames at a constant rate, whereas the event camera continuously reports pixel-wise brightness changes (red: positive events, blue: negative events). Figure from [30].

Alternatively, conventional images can be reconstructed from event streams. Intensity image reconstruction is a common task in event-vision literature. The reconstructions allow to visualize events and to apply off-the-shelf computer vision techniques to event cameras [30]. Early approaches focused on reconstructing a single image based on moving event cameras through a static scene [7, 14]. Based on the brightness constancy assumption, they interpret events as temporal brightness gradients and jointly estimate multiple quantities. Cook *et al.* [7] simultaneously estimate rotation (3 DoF), optical flow, image gradient, and the intensity reconstruction, using interconnected networks. Kim *et al.* [14] used a pixel-wise Extended Kalman Filter to jointly estimate rotation (3 DoF), image gradient, and

the absolute intensity. This approach was later extended to 3D scenes together with translation and rotation tracking (6 DoF) [15]. Bardow *et al.* [1] introduced the first video reconstruction approach that works even in dynamic scenes. They jointly estimate intensity images and optical flow by formulating an energy minimization problem with a sliding window approach.

However, these approaches have not been able to reconstruct images that are qualitatively on par with conventional cameras in terms of visual appearance. This report analyzes more recent reconstruction methods that define current state-of-the-art and investigates their application for autonomous systems. The rest of the report is structured as follows. Section 2 provides a general background for event cameras. Section 3 presents two approaches to reconstruct qualitative images. More specifically, Section 3.1 deals with model-based approaches that use direct integration in combination with filtering for efficient reconstruction. Section 3.2 presents learning-based approaches that estimate reconstructions from events using deep learning methods trained on large datasets. The application of event-cameras and image reconstruction for autonomous systems are discussed in Section 4, and conclusions are drawn in Section 5.

2. Background

Event cameras pose a paradigm shift in computer vision. This section formally presents how event cameras operate and introduces established data representations of events.

2.1. Event Generation Model

Event sensors report an *asynchronous* stream of events that are *independently* triggered by pixels for logarithmic changes in brightness $L = \log I$. In a noise-free scenario, this can be formalized by the ideal event generation model [8, 21]. Here an event is formulated as a tuple $e_k = (\mathbf{x}_k, t_k, p_k)$ that occurs at time t_k and at a pixel $\mathbf{x}_k = (x_k, y_k)^\top$, if the brightness increment exceeds a contrast threshold C. The condition can be formulated as

$$p_k(L(\mathbf{x}_k, t_k) - L(\mathbf{x}_k, t - \Delta t_k)) > C, \qquad (1)$$

where Δt_k is the elapsed time since the last event at pixel $\mathbf{x}_k$, and $p_k \in \{-1, 1\}$ is the polarity (or sign) of the brightness change. The model can be extended for colored events [17, 37]. The threshold C depends on bias currents on the sensor and can be fixed by the user for the given conditions [23]. However, setting the brightness threshold can constitute a trade-off, as small values capture more noise, whereas large thresholds capture fewer details of brightness changes. Furthermore, even when fixed manually, the threshold highly varies depending on factors such as the temperature [23], manufacturing imperfections, and circuit noise [43], but also the polarity of the events [40]. Therefore, it must be assumed that C is neither constant nor uniform for each pixel.

Event sensors are inspired by the functionality of the human retina. Similarly, increments and decrements in light stimulate photoreceptor cells that subsequently send signals (called *"spikes"*) to the brain [26]. Therefore, event-vision is closely related to topics in neuromorphic engineering and computing.

An event-vision sensor has several benefits in design. The pixel circuit is fast in detecting an event, and the camera can timestamp the event with microsecond resolution. Therefore, event cameras have a *high temporal resolution* without motion blur like conventional cameras. The pixels operate independently without having to wait for global exposures, which results in a *low latency* (sub-millisecond). Each pixel can adapt to bright and dark stimuli due to the independence and logarithmic operation. Thus, the event-sensor can acquire visual information with a *high dynamic range*.

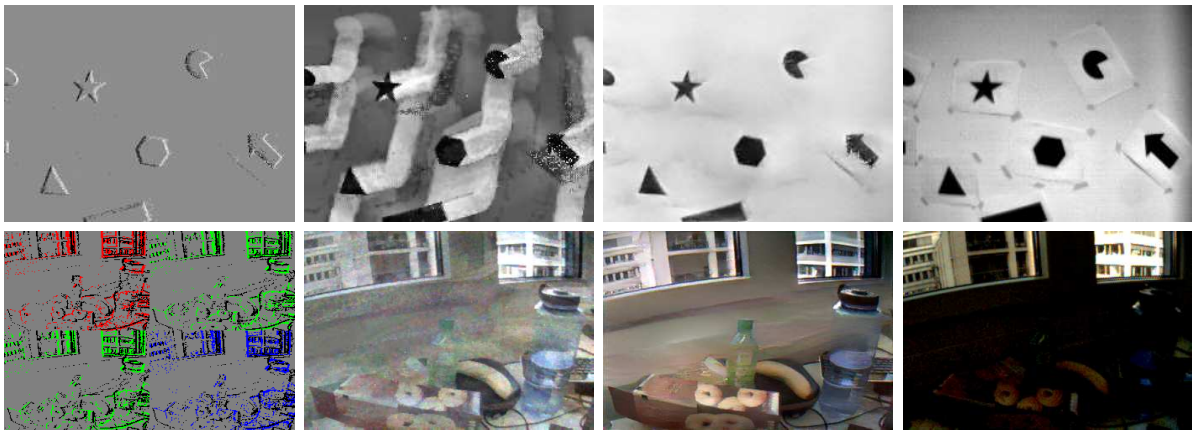
2.2. Event Representation & Processing

Due to the novelty of event cameras, there is no consensus collection of algorithms to extract features from events. The following introduces common data representations of events and corresponding processing methods.

Individual Events $e_k = (\mathbf{x}_k, t_k, p_k)$ are used for processing methods that are applied for each incoming event. This allows minimal latency but requires heavy processing, especially at high event rates. Methods include Spiking Neural Networks (SNNs), along with probabilistic and deterministic filters (see Section 3.1). Examples in literature include: [10, 14, 25, 35].

Event Packets: Packets $\varepsilon_k = \{e_k\}_{k=1}^N$ are aggregated groups of subsequent events that are processed together. The packet size N is either fixed or varies to capture a constant time interval Δt. Packets introduce latency but require fewer processing steps which can be crucial for real-time applications. Possible processing methods depend on the downstream representation of the package (see Section 3.2), such as event frames or voxel grids. Examples in literature: [21, 30, 32].

Event Frames: Event packets are converted into an image structure (2D grid) by a simple method, e.g., by pixel-wise summation of the events or accumulating the polarity. Thereby, the images represent a 2D pixel histogram of the events. This approach removes a large portion of temporal information. Nevertheless, event frames are often used in literature because they allow the application of well-studied computer vision algorithms, even though event frames do not share the statistics of natural images. It enables the usage of successful deep learning architectures of vision tasks, such as Convolutional Neural Networks (CNNs). Event



(a) Events (b) High-Pass Filter (c) E2VID (d) DAVIS Frame

Figure 2. Comparison of (b) high-pass filter [35] and (b) E2VID [30, 31]. Examples were recorded with a DAVIS camera [5, 17] that can capture (a) events and (d) conventional images. The first row shows conventional event streams. The high-pass filter suffers from "bleeding edges" and artifacts, while E2VID recovers smooth images. The second row shows colored event data from CED [37], where the color channels are visualized separately. Positive events are in the corresponding color of the filter, and negative events are black. The reconstructions capture the scene even in low light due to the high dynamic range of the event data. Images from [31].

frames are depicted in Figure 2 (a) and Figure 4 (a). Examples in literature are: [7, 11, 19, 29].

Voxel Grids: Similarly to frames, an event package is converted into a 3D grid of voxels representing a spatial-temporal histogram (see Section 3.2). Events are accumulated in voxels, where the voxel represents a pixel in a defined time interval. Although time discretization is needed, the voxel grid preserves more temporal information than event frames. Voxel grids are compatible with CNNs and other common vision techniques. However, the 3D grid comes with greater memory and computation requirements. Examples in literature: [1, 30, 42].

Reconstructed Images are regular brightness images reconstructed from events. In contrast to event frames, reconstructed images share the characteristics of natural images, thus becoming easy to interpret for the user. The images act as an intermediate representation for event data. Reconstructed images ideally capture the benefits of event cameras, such as a high dynamic range or no motion blur, while being compatible with conventional vision algorithms. Thus, event data can be easily applied to any downstream task [30, 31].

3. Image Reconstruction

This report presents two paradigms to reconstruct images from events. First, this section introduces model-based approaches and the concept of direct integration of events. The second part examines learning-based methods that leverage advances from deep learning with supervised techniques.

3.1. Model-Based Approaches

Under ideal circumstances (noise-free, ideal sensor response), event sensors can capture real brightness images when integrating the events over time. The reconstructed log-intensity $\hat{L}(\mathbf{x}; t)$ is reconstructed by accumulating the events $e_k = (\mathbf{x}_k, t_k, p_k)$ as follows,

$$\hat{L}(\mathbf{x}; t) = L(\mathbf{x}; 0) + \sum_{0<t_k\leq t} p_k C \delta_{\mathrm{K}}(\mathbf{x} - \mathbf{x}_k)\delta_{\mathrm{D}}(t - t_k),$$

where $L(\mathbf{x}; 0)$ is the true log-intensity offset at $t = 0$, δ_{K} and δ_{D} are the Kronecker and Dirac delta functions, respectively (which select the pixel to update) [22]. A step function approximates the continuous change of intensity. This approach has two apparent limitations. The threshold C is noisy and non-uniform, resulting in accumulated noise over time. Furthermore, the log-intensity offset is mostly unknown, and the reconstructed image rather shows the intensity change relative to $t = 0$. In a more recent approach, a *high-pass* filter is applied to the event stream [35]. The filter operates pixel-wise and suppresses events with a low temporal frequency, whereas high-frequency events are passed. This procedure assumes that events with a low-frequency are likely noise of a static pixel, while high-frequency events show actual brightness changes (e.g., an object moving through the scene). The filter noticeable improves the reconstructions only from events. The method separately suggests fusing the events with frames from conventional cameras. A *low-pass* filter operates on the conventional frames in-order to extract static pixel intensities (e.g., the background of the scene). The low- and high-pass outputs are merged and constantly updated with a *comple-*

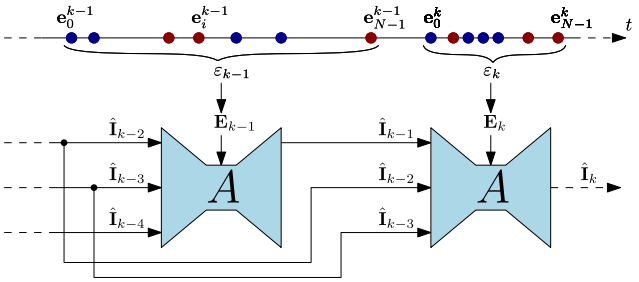
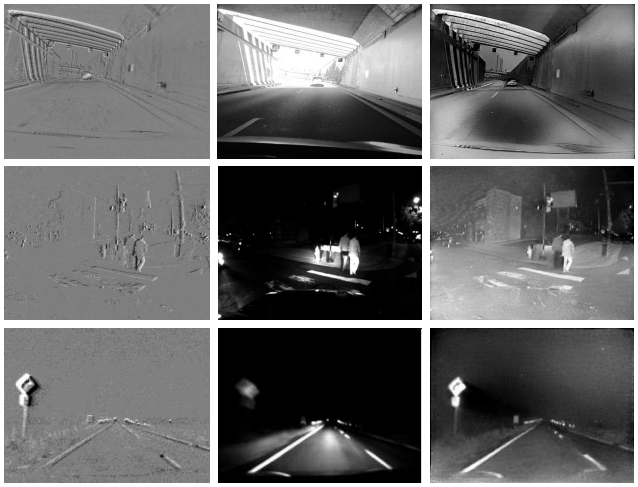


Figure 3. Overview of E2VID [30]. The incoming events (visualized as red/blue dots) are collected in packets ε_k with a fixed number of N events. The packets are converted into a 3D spatio-temporal tensor $\mathbf{E}_k$ that forms the input with the last K reconstructed images. The recurrent network outputs a new image reconstruction $\hat{\mathcal{I}}_k$ for each event packet. Figure from [30].



(a) Events (b) Frame (c) Reconstruction

Figure 4. Image reconstructions of E2VID in driving scenarios under challenging lighting [30, 31]. First row: sequence while driving out a tunnel [31]. Second row: inner-city night drive from [47]. Third row: night driving sequence from [35]. Frames of conventional cameras (b) suffer from over- and under-exposure, while the events (a) are more robust in given scenarios. The reconstructions (c) incorporate the high dynamic range and suffer less from motion blur. Images from [31].

mentary filter. Images can be reconstructed for individual events and maintain a high-temporal resolution. In contrast to the conventional frames, the reconstructions largely remove motion blur while greatly improving the high dynamic range.

Nonetheless, direct integration suffers from artifacts, such as "bleeding edges" [31], as shown in Figure 2. More importantly, reconstructing based on events only does not recover the initial image.

3.2. Learning-Based Approaches

In recent years, learning-based methods have been applied to image reconstruction [2, 42]. Rebecq *et al.* [30, 31] introduced E2VID, a recurrent neural network to reconstruct video frames from an event stream. The network is trained supervised on simulated data. This method replaced the high-pass filter described in Section 3.1 as state-of-the-art.

The event stream is represented as event packets $\varepsilon_k = \{e_k\}_{k=1}^N$ with a fixed number of N events. The goal is to learn a mapping to reconstruct an image $\hat{\mathcal{I}}_k \in [0,1]^{W \times H}$ for every incoming event package. Here, the mapping is a recurrent neural network that is based on UNet [33]. The network extracts features from the input and reconstructs the image using regular convolutions. In order to apply convolutions, each ε_k is converted into a tensor $\mathbf{E}_k$ that represents a spatio-temporal voxel-grid. The events are discretized into B temporal bins, resulting in a $B \times W \times H$ shape of $\mathbf{E}_k$. The input is attained by concatenating the event tensor $\mathbf{E}_k$ with the the K previous reconstructed images $\{\hat{I}_{k-K}, ..., \hat{I}_{k-1}\}$. This results in a recurrent architecture with a $(B+K) \times W \times H$ sized input, as shown in Figure 3.

The network is trained in a supervised fashion, meaning a large dataset of event streams with corresponding ground truth image sequences is needed. The reconstructed images should introduce the benefits of event cameras, such as high dynamic range or the absence of motion blur. Thus, images of conventional cameras would provide poor ground truth data. Therefore, E2VID is trained exclusively on synthetic data that was generated with the event simulator ESIM [27], and tested on real event data.

The network is trained using a calibrated perceptual loss (LPIPS) [45], where the target image $\mathcal{I}_k$ and reconstructed image $\hat{\mathcal{I}}_k$ are passed into a VGG-Net [39], pre-trained on Image-Net [34]. The loss corresponds to the distance of the VGG feature maps on multiple layers. The network learns to reconstruct images based on the statistics of natural images when minimizing the perceptual loss.

The authors further improve E2VID in a subsequent publication [30] by modifying the architecture and loss. The recurrent network adds ConvLSTM layers [38], and carries an internal state $\mathbf{s}_k$ instead of predicting based on the last K reconstructed images. Furthermore, a temporal consistency term is added to the loss, which removes some temporal artifacts by enforcing consistency between frames. However, this loss needs optical flow maps during training for warping in-between frames.

In comparison to model-based methods, E2VID suffers less from artifacts and outperforms previous methods across several metrics, including mean square error, structural similarity, and LPIPS (see Table 1). Moreover, the reconstructions notably surpass conventional cameras in low-light and

Table 1. Comparison of the high-temporal filter (HF) [35], improved version of E2VID [31], and FireNet [36]. Methods were evaluated on sequences of the Event Camera Dataset [22], based on mean square error (MSE), structural similarity (SSIM), and LPIPS. Best in bold. Values from [36].

	MSE			SSIM			LPIPS		
Sequence	HF	E2VID	FireNet	HF	E2VID	FireNet	HF	E2VID	FireNet
dynamic_6dof	0.10	0.14	**0.12**	0.39	0.46	**0.47**	0.54	0.46	**0.44**
boxes_6dof	0.08	**0.04**	0.04	0.49	0.62	**0.64**	0.50	0.38	**0.37**
poster_6dof	0.07	0.06	**0.04**	0.49	0.62	**0.65**	0.45	0.35	**0.34**
shapes_6dof	0.09	0.04	**0.02**	0.50	**0.80**	0.79	0.61	0.47	**0.46**
office_6zigzag	0.09	**0.03**	0.04	0.38	**0.54**	0.54	0.54	0.41	**0.40**
slider_6depth	0.06	0.05	**0.05**	0.50	0.58	**0.59**	0.50	0.44	**0.41**
calibration	0.09	**0.02**	0.04	0.48	**0.70**	0.66	0.48	**0.36**	0.37
Mean	0.08	0.05	**0.05**	0.46	0.62	**0.62**	0.52	0.41	**0.40**

Table 2. Comparison of E2VID [31], and E2VID trained to reduce the sim-to-real gap (S2RG) [40]. Both approaches are evaluated based on the mean square error (MSE), and LPIPS. Stoffregen *et al.* evaluate their method on selected sequence cuts of the Event Camera Dataset [22]. Thus, the values are not directly comparable to Table 1. Best in bold. Values from [40].

	MSE		LPIPS	
Sequence	E2VID	S2RG	E2VID	S2RG
dynamic_6dof_cut	0.17	**0.05**	0.38	**0.27**
boxes_6dof_cut	**0.04**	0.04	0.29	**0.25**
poster_6dof_cut	0.07	**0.03**	0.26	**0.19**
shapes_6dof_cut	0.03	**0.02**	0.26	**0.22**
office_6zigzag_cut	0.07	**0.04**	0.31	**0.26**
slider_6depth_cut	0.08	**0.03**	0.35	**0.24**
calibration_cut	0.07	**0.03**	0.22	**0.18**
Mean	0.07	**0.03**	0.28	**0.22**

high-speed scenarios, as shown in Figure 4. Due to the temporal resolution of the events, high frame rate videos can be reconstructed (in the range of thousands of FPS). Furthermore, the reconstructed videos allow for processing the event data with well-studied algorithms for conventional cameras directly. The authors demonstrate this by applying standard vision methods to reconstructions in tasks such as object classification and visual-inertial odometry, thereby outperforming previous methods specifically designed for event data.

Nevertheless, E2VID comes with limitations that are partially addressed by further research. Firstly, the reconstruction is computationally expensive, which limits the real-time applicability. For example, a 1280x720 image takes 93.34 ms on an NVIDIA Titan Xp GPU, which would result in a 10 FPS video [36]. Scheerlinck *et al.* proposed a smaller network, called FireNet, which reduces the number of parameters by 99% with minor trade-offs regarding the reconstruction quality [36]. Consequently, FireNet runs three times faster than E2VID on the same GPU (31.01 ms vs. 93.34 ms).

A second limitation of E2VID originates from training solely on simulated data. Thereby, inference with actual events is carried out exclusively on out-of-distribution data. Stoffregen *et at.* proposed a new strategy for generating training data by emphasizing the contrast threshold C when synthesizing data [40]. The E2VID network was trained on generated data with a wide range of contrast thresholds and noise augmentation to resemble actual event data closer. The training approach improved generalizability and outperformed the conventionally trained E2VID across several benchmarks (see Table 2). The results highlight the vital role of synthetic data in learning-based reconstruction. The training data must share the statistics of the actual use case.

4. Discussion

Autonomous systems can become more robust by introducing event cameras as visual sensors. However, image reconstruction as event representation also introduces other flaws. First, learning-based approaches have to be trained exclusively on simulated data. Thus, the simulation must be tailored to the use case and the camera parameters. The estimation is prone to errors if the training distribution mismatches the real-world application despite precautions. Dissimilarities of simulated and real-world data are especially concerns for functional safety in autonomous systems. Furthermore, image reconstruction with learned models is computationally demanding, resulting in slow processing even with advanced hardware. Smaller networks can reduce computational cost [36] but still require hardware accelerators (i.e., GPUs) for processing that are not generally available in autonomous systems (e.g., drones, or

mobile robots). Moreover, downstream applications further require time to process the images for the actual task. The application in an autonomous system becomes unfeasible when real-time requirements cannot be satisfied. Model-based approaches would offer a faster reconstruction but come with artifacts and reduced image quality.

Most importantly, image reconstruction is an expensive intermediate step when applied for a downstream target task. Although the images become applicable for conventional vision techniques, the reconstructions only transform the provided input and also discard positive aspects of event cameras. Theoretically, the event stream contains the same visual information but in a fast and highly compressed format. Event vision needs a generally applicable framework to process event streams directly. Spiking Neural Networks (SNNs) seem promising due to fitting input modality and the same bio-inspired nature. SNNs have shown promising results for specific tasks, such as optical flow estimation [25], or angular velocity regression [10]. Especially a combination of SNNs with neuromorphic processors would offer a low memory and low power vision approach that is desirable for autonomous systems [9].

Furthermore, conventional cameras and event cameras are not contrary technologies but complementary. Autonomous systems also need visual information in static or slowly varying scenes. Traditional cameras are ideally suited for such scenarios. It is most likely beneficial to consider both cameras as input data, either separately or merged. In fact, event an frame-based cameras have been fused in literature to remove motion blur [13, 18, 24], generate high-speed videos [41], and to increase dynamic range [35]. Enhancing videos with event data has potential beyond autonomous vision, e.g., in film-making or smartphone applications.

Overall, image reconstruction remains important in current event-vision research. Image reconstruction functions as a data representation of events. The representation allows for establishing a baseline for a task with conventional algorithms. Using off-the-shelf algorithms enables to validate that event data offers a valuable contribution before creating more sophisticated methods tailored to a task. Furthermore, the reconstructions remain the most familiar and interpretable visualization for humans of event data.

5. Conclusion

Event cameras introduce a novel approach to how machines perceive and represent visual information. They offer several advantages compared to conventional cameras, such as high-speed capabilities, high dynamic range, low power, and low latency. Consequently, event cameras have much potential for autonomous systems and robotics, mainly to increase robustness in challenging scenarios. The field of event-based vision has several challenges ahead, especially the development of algorithms that unlock the unique properties of events. Intensity image reconstruction can reduce the development gap by making traditional computer vision accessible to event cameras. This work presented model and learning-based approaches for image reconstruction. Model-based approaches offer fast reconstructions that are qualitatively inferior in terms of visual appearance. Learning-based approaches define current state-of-the-art by training neural networks to reconstruct images. However, such networks have high computational costs and can only be trained in simulation, restricting their application for autonomous systems. After all, image reconstruction from events remains a vital discipline in the current state of research. The reconstructions enable a bridge to traditional computer vision, are valuable for prototyping, and visualize the outstanding capabilities of event cameras.

References

[1] Patrick Bardow, Andrew J Davison, and Stefan Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 884–892, 2016. 1, 2, 3

[2] Souptik Barua, Yoshitaka Miyatani, and Ashok Veeraraghavan. Direct face detection and video reconstruction from event cameras. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016. 4

[3] Ryad Benosman, Charles Clercq, Xavier Lagorce, Sio-Hoi Ieng, and Chiara Bartolozzi. Event-based visual flow. *IEEE transactions on neural networks and learning systems*, 25(2):407–417, 2013. 1

[4] Jonathan Binas, Daniel Neil, Shih-Chii Liu, and Tobi Delbruck. Ddd17: End-to-end davis driving dataset. *arXiv preprint arXiv:1711.01458*, 2017. 1

[5] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014. 3

[6] Guang Chen, Hu Cao, Jorg Conradt, Huajin Tang, Florian Rohrbein, and Alois Knoll. Event-based neuromorphic vision for autonomous driving: a paradigm shift for bio-inspired visual sensing and perception. *IEEE Signal Processing Magazine*, 37(4):34–49, 2020. 1

[7] Matthew Cook, Luca Gugelmann, Florian Jug, Christoph Krautz, and Angelika Steger. Interacting maps for fast visual interpretation. In *The 2011 International Joint Conference on Neural Networks*, pages 770–776. IEEE, 2011. 1, 3

[8] Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J Davison, Jörg Conradt, Kostas Daniilidis, et al. Event-based vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(1):154–180, 2020. 1, 2

[9] Francesco Galluppi, Christian Denk, Matthias C Meiner, Terrence C Stewart, Luis A Plana, Chris Eliasmith, Steve Furber, and Jörg Conradt. Event-based neural computing on an autonomous mobile platform. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2862–2867. IEEE, 2014. 6

[10] Mathias Gehrig, Sumit Bam Shrestha, Daniel Mouritzen, and Davide Scaramuzza. Event-based angular velocity regression with spiking networks. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4195–4202. IEEE, 2020. 2, 6

[11] Yuhuang Hu, Jonathan Binas, Daniel Neil, Shih-Chii Liu, and Tobi Delbruck. Ddd20 end-to-end event camera driving dataset: Fusing frames and events with deep learning for improved steering prediction. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, 2020. 1, 3

[12] Joel Janai, Fatma Güney, Aseem Behl, Andreas Geiger, et al. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision*, 12(1–3):1–308, 2020. 1

[13] Zhe Jiang, Yu Zhang, Dongqing Zou, Jimmy Ren, Jiancheng Lv, and Yebin Liu. Learning event-based motion deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3320–3329, 2020. 6

[14] Hanme Kim, Ankur Handa, Ryad Benosman, Sio-Hoi Ieng, and Andrew Davison. Simultaneous mosaicing and tracking with an event camera. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014. 1, 2

[15] Hanme Kim, Stefan Leutenegger, and Andrew J Davison. Real-time 3d reconstruction and 6-dof tracking with an event camera. In *European Conference on Computer Vision*, pages 349–364. Springer, 2016. 1, 2

[16] Beat Kueng, Elias Mueggler, Guillermo Gallego, and Davide Scaramuzza. Low-latency visual odometry using event-based feature tracks. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 16–23. IEEE, 2016. 1

[17] Chenghan Li, Christian Brandli, Raphael Berner, Hongjie Liu, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. Design of an rgbw color vga rolling and global shutter dynamic and active-pixel vision sensor. In *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 718–721. IEEE, 2015. 2, 3

[18] Songnan Lin, Jiawei Zhang, Jinshan Pan, Zhe Jiang, Dongqing Zou, Yongtian Wang, Jing Chen, and Jimmy Ren. Learning event-driven video deblurring and interpolation. In *European Conference on Computer Vision*, pages 695–710. Springer, 2020. 6

[19] Ana I Maqueda, Antonio Loquercio, Guillermo Gallego, Narciso García, and Davide Scaramuzza. Event-based vision meets deep learning on steering prediction for self-driving cars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5419–5427, 2018. 1, 3

[20] Diederik Paul Moeys, Federico Corradi, Emmett Kerr, Philip Vance, Gautham Das, Daniel Neil, Dermot Kerr, and Tobi Delbrück. Steering a predator robot using a mixed frame/event-driven convolutional neural network. In *2016 Second International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*, pages 1–8. IEEE, 2016. 1

[21] Elias Mueggler, Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. Continuous-time visual-inertial odometry for event cameras. *IEEE Transactions on Robotics*, 34(6):1425–1440, 2018. 2

[22] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbruck, and Davide Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *The International Journal of Robotics Research*, 36(2):142–149, 2017. 3, 5

[23] Yuji Nozaki and Tobi Delbruck. Temperature and parasitic photocurrent effects in dynamic vision sensors. *IEEE Transactions on Electron Devices*, 64(8):3239–3245, 2017. 2

[24] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6820–6829, 2019. 6

[25] Federico Paredes-Vallés, Kirk YW Scheper, and Guido CHE De Croon. Unsupervised learning of a hierarchical spiking neural network for optical flow estimation: From events to global motion perception. *IEEE transactions on pattern analysis and machine intelligence*, 42(8):2051–2064, 2019. 2, 6

[26] Christoph Posch, Teresa Serrano-Gotarredona, Bernabe Linares-Barranco, and Tobi Delbruck. Retinomorphic event-based vision sensors: Bioinspired cameras with spiking output. *Proceedings of the IEEE*, 102(10):1470–1484, 2014. 2

[27] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. Esim: an open event camera simulator. In *Conference on Robot Learning*, pages 969–982. PMLR, 2018. 4

[28] Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization. 2017. 1

[29] Henri Rebecq, Timo Horstschäfer, Guillermo Gallego, and Davide Scaramuzza. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time. *IEEE Robotics and Automation Letters*, 2(2):593–600, 2016. 1, 3

[30] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3857–3866, 2019. 1, 2, 3, 4

[31] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):1964–1980, 2019. 3, 4, 5

[32] Paul Rogister, Ryad Benosman, Sio-Hoi Ieng, Patrick Lichtsteiner, and Tobi Delbruck. Asynchronous event-based binocular stereo matching. *IEEE Transactions on Neural Networks and Learning Systems*, 23(2):347–353, 2011. 2

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmen-

tation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 4

[35] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. Continuous-time intensity estimation using event cameras. In *Asian Conference on Computer Vision*, pages 308–324. Springer, 2018. 2, 3, 4, 5, 6

[36] Cedric Scheerlinck, Henri Rebecq, Daniel Gehrig, Nick Barnes, Robert Mahony, and Davide Scaramuzza. Fast image reconstruction with an event camera. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 156–163, 2020. 5

[37] Cedric Scheerlinck, Henri Rebecq, Timo Stoffregen, Nick Barnes, Robert Mahony, and Davide Scaramuzza. Ced: Color event camera dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2, 3

[38] 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. *Advances in neural information processing systems*, 28, 2015. 4

[39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[40] 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 *European Conference on Computer Vision*, pages 534–549. Springer, 2020. 2, 5

[41] Stepan Tulyakov, Daniel Gehrig, Stamatios Georgoulis, Julius Erbach, Mathias Gehrig, Yuanyou Li, and Davide Scaramuzza. Time lens: Event-based video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16155–16164, 2021. 6

[42] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10081–10090, 2019. 3, 4

[43] Ziwei Wang, Yonhon Ng, Pieter van Goor, and Robert E. Mahony. Event camera calibration of per-pixel biased contrast threshold. *CoRR*, abs/2012.09378, 2020. 2

[44] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access*, 8:58443–58469, 2020. 1

[45] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 4

[46] Alex Zihao Zhu, Nikolay Atanasov, and Kostas Daniilidis. Event-based feature tracking with probabilistic data association. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4465–4470. IEEE, 2017. 1

[47] Alex Zihao Zhu, Dinesh Thakur, Tolga Özaslan, Bernd Pfrommer, Vijay Kumar, and Kostas Daniilidis. The multivehicle stereo event camera dataset: An event camera dataset for 3d perception. *IEEE Robotics and Automation Letters*, 3(3):2032–2039, 2018. 4

[48] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Ev-flownet: Self-supervised optical flow estimation for event-based cameras. *arXiv preprint arXiv:1802.06898*, 2018. 1