Event-based Monocular Depth Prediction in Night Driving

Javier Hidalgo-Carrió, Daniel Gehrig, Davide Scaramuzza

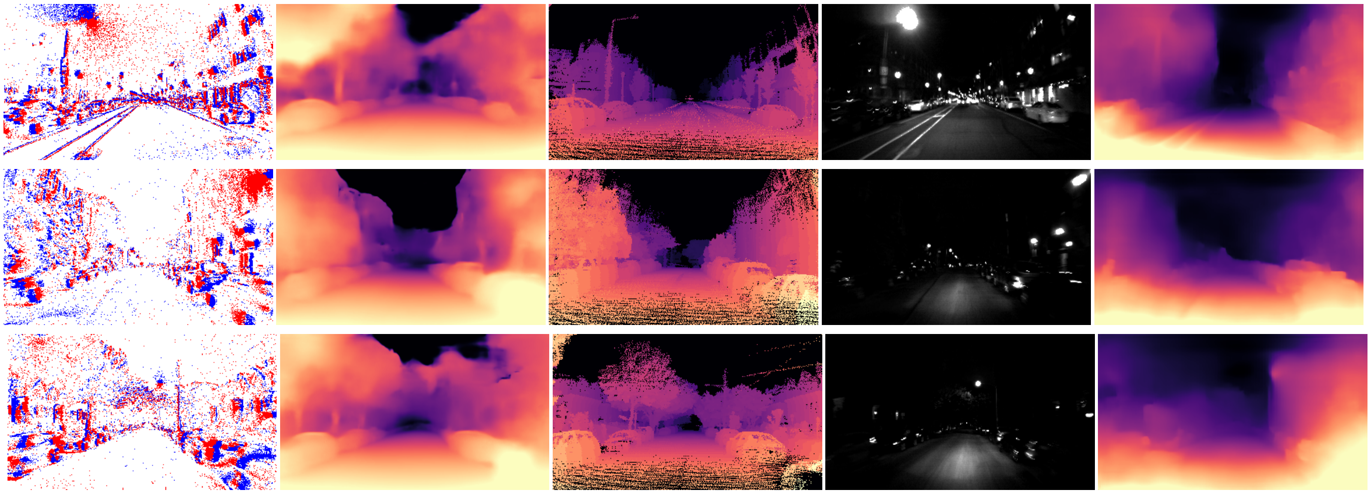

Fig. 1: Qualitative comparison of the night sequences in the MVSEC dataset. The first column shows the events. Second column depicts e2depth (our) dense depth map predictions. Third column is the ground truth depth. The fourth column shows DAVIS grayscale frames and the fifth column the MegaDepth prediction using the grayscale frames.

***Abstract*— Event cameras are novel sensors of interest for robust perception in challenging environments. They output brightness changes in the form of a stream of asynchronous ”events” instead of intensity frames. Compared to conventional image sensors, they offer significant advantages: high temporal resolution, high dynamic range, no motion blur, and much lower bandwidth. Recently, learning-based approaches have been applied to event-based data, thus unlocking their potential and making significant progress in a variety of tasks, such as monocular depth prediction. We propose a recurrent architecture to solve monocular depth and show significant improvement over standard frame-based methods at night. We also present an event camera sensor for the CARLA simulator in order to boost event-based deep learning tasks for autonomous driving and robotics.**

I. Introduction

Event cameras, such as the Dynamic Vision Sensor (DVS) [13] or the ATIS [15], are bio-inspired vision sensors with radically different working principles compared to conventional cameras. While standard cameras capture intensity images at a fixed rate, event cameras only report changes of intensity at the pixel level and do this asynchronously at the time they occur. The resulting stream of events encodes the time, location, and sign of the change in brightness. Event cameras possess outstanding properties when compared to standard cameras. They have a very high dynamic range (140 dB versus 60 dB), no motion blur, and high temporal resolution (in the order of microseconds). Event cameras are thus sensors that can provide high-quality visual information even in challenging high-speed scenarios and high dynamic range environments, enabling new application domains for vision-based algorithms. Recently, these sensors have received great interest in various computer vision fields, ranging from computational photography [20], [19], [23], [24] to visual odometry [22], [18], [17], [28], [30], [8] and depth prediction [9], [18], [16], [27], [29], [26], [30]. The survey in [3] gives a good overview of the applications for event cameras.

Monocular depth prediction has focused primarily on standard cameras, which work synchronously, i.e., at a fixed frame rate. State-of-the-art approaches are usually trained and evaluated in common datasets such as KITTI [5], Make3D [1] and NYUv2 [14]. Depth prediction using event cameras has experienced a surge in popularity in recent years [22], [16], [27], [18], [17], [9], [30], [27], [29], [26], due to its potential in robotics and the automotive industry. Event-based depth prediction is the task of predicting the depth of the scene at each pixel in the image plane, and is important for a wide range of applications, such as robotic grasping [11] and autonomous driving, with low-latency obstacle avoidance and high-speed path planning.

However, while event-cameras have appealing properties they also present unique challenges. Due to the working principles of the event camera, they respond predominantly to edges in the scene, making event-based data inherently sparse and asynchronous. This makes dense depth estimation with an event camera challenging, especially in low contrast regions, which do not trigger events and, thus, need to be filled in. Prior work in event-based depth estimation has made significant progress in this direction, especially since the advent of deep learning. However, most existing works are limited: they can reliably only predict sparse or semi-dense depth maps [22], [16], [27], [18], [17], [9], [30], [27], [29] or rely on a stereo setup to generate dense depth predictions [26].

In this work, we focus on dense, monocular depth using an event camera, which addresses the aforementioned limitations. We show that our approach reliably generates dense depth maps overcoming the sparsity in a stream of events. Our contributions are the following:

- A recurrent network that predicts dense per-pixel depth from a monocular event camera.
- An event camera sensor for the CARLA [2] simulator.

II. Methodology

Events cameras output events at independent pixels and do this asynchronously. Specifically, their pixels respond to changes in the spatio-temporal log irradiance $L(\mathbf{u}, t)$ that produces a stream of asynchronous events. For an ideal sensor, an event $e_i = (\mathbf{u}_i, t_i, p_i)$ is triggered at time t_i if the brightness change at the pixel $\mathbf{u}_i = (x_i, y_i)^\intercal$ exceeds a threshold of $\pm C$. The event polarity p_i denotes the sign of this change.

The goal is to predict dense monocular depth from a continuous stream of events. The method works by processing subsequent non-overlapping windows of events $\epsilon_k = \{e_i\}_{i=0}^{N-1}$ each spanning a fixed interval $\Delta T = t_{N-1}^k - t_0^k$. For each window, we predict log depth maps $\{\hat{\mathcal{D}}_k\}$, with $\hat{\mathcal{D}}_k \in [0,1]^{W \times H}$. We implement log depth prediction as a recurrent convolutional neural network with an internal state $\mathbf{s}_k$. We train our network in a supervised manner, using ground truth depth maps. The network is first trained in simulation using perfect ground truth and synthetic events and finetuned in a real sequence.

A. Event Representation

Due to the sparse and asynchronous nature of event data, batches of events ϵ_k need to be converted to tensor-like representations $\mathbf{E}_k$. One way to encode these events is by representing them as a spatio-temporal voxel grid [30], [4] with dimensions $B \times H \times W$. Events within the time window ΔT are collected into B temporal bins according to

$$\mathbf{E}_k(\mathbf{u}_k, t_n) = \sum_{e_i} p_i \delta(\mathbf{u}_i - \mathbf{u}_k) \max(0, 1 - |t_n - t_i^*|) \quad (1)$$

where $t_i^* = \frac{B-1}{\Delta T}(t_i - t_0)$ is the normalized event timestamp. In our experiments, we used $\Delta T = 50ms$ of events and $B = 5$ temporal bins. To facilitate learning, we further normalize the non-zero values in the voxel grid to have zero mean and unit variance.

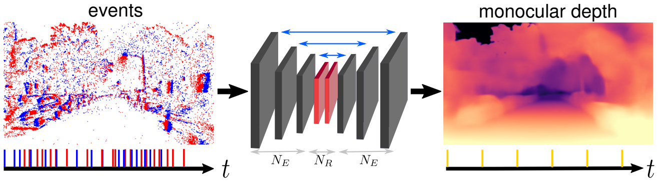

Fig. 2: Method overview, the network receives asynchronous events inputs and predicts normalized log depth $\hat{\mathcal{D}}_k$. Our method uses N_R recurrent blocks to leverage the temporal consistency in the events input.

B. Network Architecture

It consists of a recurrent, fully convolutional neural network, based on the UNet architecture [21]. The network input is first processed by a head layer $\mathcal{H}$ and then N_E recurrent encoder layers ($\mathcal{E}^i$) followed by N_R residual blocks ($\mathcal{R}^j$) and N_E decoder layers $\mathcal{D}^l$. A final depth prediction layer $\mathcal{P}$ produces the output of our network. The head layer produces an output with N_b channels, which is doubled at each encoder layer, resulting in a feature map with $N_b \times 2^{N_E}$ output channels. $\mathcal{P}$ performs a depth-wise convolution with one output channel and kernel size 1. We use skip connections between symmetric encoder and decoder layers (see Fig. 2). At the final layer, the activations are squashed by a sigmoid activation function. Each encoder layer is composed of a downsampling convolution with kernel size 5 and stride 2 and a ConvLSTM [25] module with kernel size 3. The encoding layers maintain a state c_k^i which is at 0 for $k = 0$. The residual blocks use a kernel size of 3 and apply summation over the skip connection. Finally, each decoder layer is composed of a bilinear upsampling operation followed by convolution with kernel size 5. We use ReLU except for the prediction layer, and batch normalization [7]. In this work we use $N_E = 3$, $N_R = 2$ and $N_b = 32$ and we unroll the network for $L = 40$ steps.

C. Depth Map Post-processing

As usual, in recent work on depth prediction, we train our network to predict a normalized log depth map. Log depth maps have the advantage of representing large depth variations in a compact range, facilitating learning. If $\hat{\mathcal{D}}_k$ is the depth predicted by our model, the metric depth can be recovered by performing the following operations:

$$\hat{\mathcal{D}}_{m,k} = \mathcal{D}_{\max} \exp(-\alpha(1 - \hat{\mathcal{D}}_k)) \quad (2)$$

Where $\mathcal{D}_{\max}$ is the maximum expected depth and α is a parameter chosen, such that a depth value of 0 maps to minimum observed depth. In our case, $\mathcal{D}_{\max} = 80$ meters and $\alpha = 3.7$ corresponding to a minimum depth of 2 meters.

D. Training Details

We train our network in a supervised fashion, by minimizing the scale-invariant and multi-scale scale-invariant gradient matching losses at each time step. Given a sequence of ground truth depth maps $\{\mathcal{D}_k\}$, denote the residual $\mathcal{R}_k = \hat{\mathcal{D}}_k - \mathcal{D}_k$. Then the scale-invariant loss is defined as

$$\mathcal{L}_{k,\text{si}} = \frac{1}{n}\sum_{\mathbf{u}}(\mathcal{R}_k(\mathbf{u}))^2 - \frac{1}{n^2}\left(\sum_{\mathbf{u}}\mathcal{R}_k(\mathbf{u})\right)^2, \quad (3)$$

where n is the number of valid ground truth pixels $\mathbf{u}$. The multi-scale scale-invariant gradient matching loss encourages smooth depth changes and enforces sharp depth discontinuities in the depth map prediction. It is computed as follows:

$$\mathcal{L}_{k,\text{grad}} = \frac{1}{n}\sum_{s}\sum_{\mathbf{u}}|\nabla_x\mathcal{R}_k^s(\mathbf{u})| + |\nabla_y\mathcal{R}_k^s(\mathbf{u})|. \quad (4)$$

Here $\mathcal{R}_k^s(\mathbf{x})$ refers to the residual at scale s and the L_1 norm is used to enforce sharp depth discontinuities in the prediction. In this work, we consider four scales, similar to [12]. The resulting loss for a sequence of L depth maps is thus

$$\mathcal{L}_{\text{tot}} = \sum_{k=0}^{L-1}\mathcal{L}_{k,\text{si}} + \lambda\mathcal{L}_{k,\text{grad}}. \quad (5)$$

The hyper-parameter $\lambda = 0.5$ was chosen by cross-validation. We train with a batch size of 20 and a learning rate of 10^{-4} and use the Adam [10] optimizer.

Our network requires training data in the form of events sequences with corresponding depth maps. However, it is difficult to get perfect dense ground truth depth maps in real datasets. For this reason, we propose to first train the network using synthetic data generated in CARLA and get the final metric scale by finetuning the network using real events from the MVSEC dataset.

III. Experiments

Our method is compared against the state of the art monocular depth: two image-based techniques, MonoDepth [2] [6] and MegaDepth [12], and the event-based approach from [30] (see Table I). MegaDepth is further applied to frames reconstructed from events using E2VID [20]. The evaluation is done using the average mean error at depths of 10m, 20m, and 30m since these are the available metrics reported until now in the MVSEC dataset. The values for MonoDepth are directly taken from the evaluation in [30]. Our work gives more accurate depth prediction at all distances with an average improvement overall sequence of 26.25% at 10m, 25.25% at 20m and 21.0% at 30m with respect to values reported in [30]. Our method produces dense depth results up to 50.0% improvement with respect to previous methods in *outdoor night3* sequence of MVSEC. Image-based methods have difficulties to predict depth in low light conditions. MegaDepth applied to reconstructed frames performs more accurately in night sequences. However, the direct use of events (i.e.: end to end without parsing through image reconstruction) in our method gives a better estimate since the events capture increments in contrast at a higher temporal resolution.

None of the methods uses samples from night driving sequences at training time, neither image-based methods nor event-based solutions. MegaDepth is trained with the MD dataset from images available on the Internet and this achieves superior generalizability than MonoDepth which is trained with KITTI [5] and reported the values from Zhu et al [30]. Fig. 1 contains a visual comparison.

IV. Conclusion

We presented a solution to predict monocular dense solely using events. We reported results on the night drive sequences of the Multi Vehicle Stereo Event Camera Dataset (MVSEC). We showed that training on synthetic data is beneficial for several reasons: it helps the network converge faster, depth maps have a better quality due to perfect ground truth, and simulation captures a larger variety of conditions. For this reason we released the event camera sensor for CARLA.

References

[1] M. S. Ashutosh Saxena and A. Y. Ng. Make3d: Learning 3d scene structure from asingle still image. 2009.

[2] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. CARLA: An open urban driving simulator. In *Conf. on Robotics Learning (CoRL)*, 2017.

[3] G. Gallego, T. Delbruck, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza. Event-based vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[4] 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.

[5] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Research*, 32(11):1231–1237, 2013.

[6] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017.

[7] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. Int. Conf. Mach. Learning (ICML)*, 2015.

[8] 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.

[9] 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.

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

[11] I. Lenz, H. Lee, and A. Saxena. Deep learning for detecting robotic grasps. *Int. J. Robot. Research*, 34(4-5):705–724, 2015.

[12] Z. Li and N. Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018.

[13] 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.

[14] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[15] C. Posch, D. Matolin, and R. Wohlgenannt. A QVGA 143dB dynamic range asynchronous address-event PWM dynamic image sensor with lossless pixel-level video compression. In *IEEE Intl. Solid-State Circuits Conf. (ISSCC)*, pages 400–401, 2010.

[16] H. Rebecq, G. Gallego, and D. Scaramuzza. EMVS: Event-based multi-view stereo. In *British Mach. Vis. Conf. (BMVC)*, 2016.

[2] MonoDepth performs more accurately than MonoDepth2 for the MVSEC dataset

Dataset	Distance	Frame based			Event based	
		MonoDepth [6]	MegaDepth [12]	MegaDepth$^+$ [12]	Zhu et al. [30]	e2depth [1] (ours)
outdoor night1	10m	3.49	2.54	**2.40**	3.13	3.38
	20m	6.33	4.15	4.20	4.02	**3.82**
	30m	9.31	5.60	5.80	4.89	**4.46**
outdoor night2	10m	5.15	3.92	3.39	2.19	**1.67**
	20m	7.80	5.78	4.99	3.15	**2.63**
	30m	10.03	7.05	6.22	3.92	**3.58**
outdoor night3	10m	4.67	4.15	4.56	2.86	**1.42**
	20m	8.96	6.00	5.63	4.46	**2.33**
	30m	13.36	7.24	6.51	5.05	**3.18**

TABLE I: Average absolute depth errors (in meters) at different cut-off depth distances (lower is better). MegaDepth$^+$ refers to MegaDepth [12] using E2VID [20] reconstructed frames. Our results outperform state of the art image-based monocular depth prediction methods [6], [12] while outperforming state of the art event-based methods [30].

[17] H. Rebecq, T. Horstschaefer, and D. Scaramuzza. Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization. In *British Mach. Vis. Conf. (BMVC)*, 2017.

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

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

[24] C. Scheerlinck, H. Rebecq, D. Gehrig, N. Barnes, R. Mahony, and D. Scaramuzza. Fast image reconstruction with an event camera. In *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2020.

[25] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Conf. Neural Inf. Process. Syst. (NIPS)*, 2015.

[26] S. Tulyakov, F. Fleuret, M. Kiefel, P. Gehler, and M. Hirsch. Learning an event sequence embedding for dense event-based deep stereo. In *Int. Conf. Comput. Vis. (ICCV)*, 2019.

[27] Y. Zhou, G. Gallego, H. Rebecq, L. Kneip, H. Li, and D. Scaramuzza. Semi-dense 3D reconstruction with a stereo event camera. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 242–258, 2018.

[28] A. Z. Zhu, N. Atanasov, and K. Daniilidis. Event-based visual inertial odometry. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5816–5824, 2017.

[29] A. Z. Zhu, Y. Chen, and K. Daniilidis. Realtime time synchronized event-based stereo. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 438–452, 2018.

[30] 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.