Dual Transfer Learning for Event-based End-task Prediction via Pluggable Event to Image Translation –Supplementary Material–

Lin Wang, Yujeong Chae, and Kuk-Jin Yoon
Visual Intelligence Lab., KAIST, Korea
{wanglin, yujeong, kjyoon}@kaist.ac.kr

Abstract

Due to the limitation of space in the main paper, we provide more detailed analysis for the proposed DTL framework and present more experimental results in this supplementary material. Specifically, in Sec.1, we describe more detailed mathematical formulation of the event representation. Sec.2 provides more details for the proposed feature-level transfer loss where we illustrate the feature transformation and matching using graphs. In Sec.3, we provide more experimental results on semantic segmentation, together with the the results with HDR scenes. Lastly, in Sec.4, we present the implementation details of depth estimation and present more experimental results.

1. Event Representation

As DNNs are designed for image-/tensor-like inputs, we first describe the way of event embedding. An event e is interpreted as a tuple $(\mathbf{u}, t, p)$, where $\mathbf{u} = (x, y)$ is the pixel coordinate, t is the timestamp, and p is the polarity indicating the sign of brightness change. An event occurs whenever a change in log-scale intensity exceeds a threshold C. To process event streams using DNNs, it is required to stack sparse events into image-like or a fixed tensor-like representations [6, 13, 17]. An event camera interprets the intensity changes as asynchronous event streams.

$$L(x, y, t) - L(x, y, t - \Delta t) \geq pC \tag{1}$$

where $p \in \{-1, 1\}$, and Δt is the time interval since the last event at pixel $\mathbf{u} = (x, y)$. A number of events are triggered in a given time interval Δt, which can be denoted as:

$$\mathcal{E} = e_{i=1}^{N} = \{\mathbf{u}_k, t_k, p_k\}_{i=1}^{N} \tag{2}$$

A natural choice is to encode events in a spatial-temporal 3D volume to a voxel grid [13] or event frame [6] or multi-channel image [1, 17, 21]. In this work, consider representing events in a multi-channel representation. As show in Table 1, we compare the dimensions of the commonly used event representation, and describe its characteristics in keeping temporal and polarity information. In comparison, although they have different intuitions regarding how event information is extracted, they all share similar characteristics as they aim to convert sparse event streams to an event tensor with certain channel dimension. There is no absolute criteria determining which is better and which is worse as their performance on the vision tasks varies. In the paper, we consider to represent events to multi-channel image as the inputs to the DNNs, as done in [16, 17, 21]. The event volume can be described as:

$$\mathcal{E} = n \sum_{i=1}^{N} p_i \delta(x - x_i, y - y_i) \tag{3}$$

where n is the number of channels, and each pixel $\mathbf{u}$ sums the values of p_i that fall within it. In the ablation study on the semantic segmentation task as shown in Table 5 of main paper, we find a multi-channel representation shows better results than the others.

2. Details of Feature-level Transfer

As the feature representations of the decoder D_2 for the EIT branch deliver fine-grained visual structural information of scenes, we leverage these visual knowledge to guide the feature representation of the decoder D of the EEL branch. To this end, we propose a novel approach to transfer the instance-level similarity along the spatial locations between EIT branch and EEL branch based on affinity graphs, as shown in Fig. 1(a). We extract the feature maps at the penultimate layer of the EEL decoder and EIT decoder, respectively. Note that it is approachable to extract multiple features from the different positions of the encoders; however, we find that the penultimate layer is informative than the layers in other positions. We thus only leverage the penultimate layer as the feature output layer. As the channel dimension of the feature maps from both

Table 1: A comparison of different event representation approaches. H and W represent the spatial resolution of events. B and C represents the number of bins and channels, respectively.

Method	Dimensions of representation	Temporal information	Description
Event frame [6, 20]	4×H×W	Lost	Sum of each polarity
Voxel grid [13, 21]	B×H×W	Retained in B bins	3D voxel volume by summing events
Multi-channel [16, 17, 21]	C×H×W	Retained in C channels	Accumulated events with fixed numbers

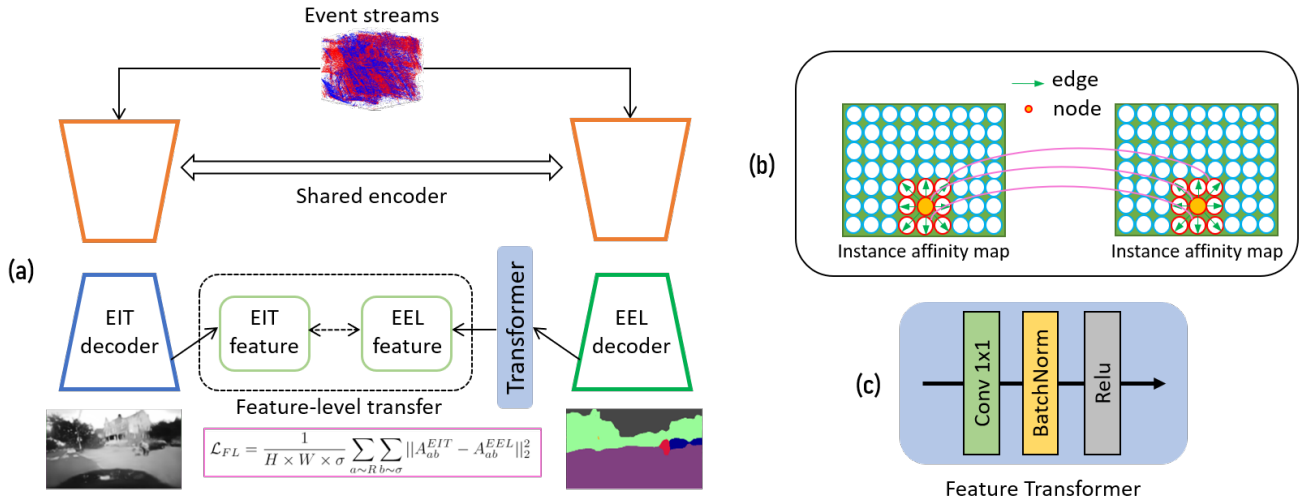

Figure 1: An illustration of the proposed feature-level transfer loss. (a) The overall framework of calculating the feature affinity vectors from the EIT and EEL decoders. (b) The details of representing instances as nodes and edges, which are then represented as the instance feature affinity maps. (c) Detailed structure of the feature transformer.

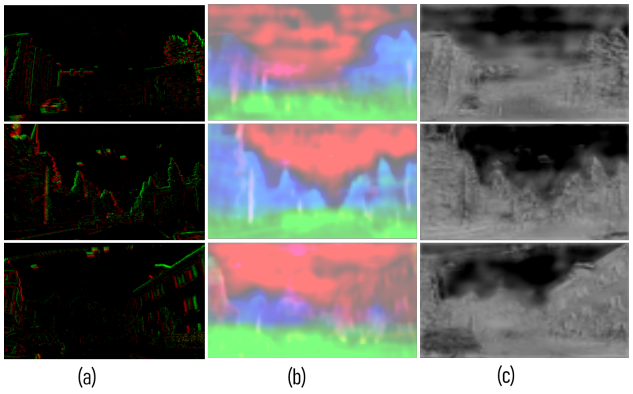

Figure 2: Additional feature visualizations of EEL and EIT branches.

branches may be different in some cases, we then design a feature transformer, which transforms the feature dimensions of EEL branch to the same dimension as the feature map of EIT branch, as shown in Fig. 1(c). The feature transformer consists of one 1x1 convolution layer, followed by a batch normalization layer and a Relu activation function. To calculate the spatial pair-wise relations between the feature maps, we represent the features in instance affinity graphs, as shown in Fig. 1(b). Specifically, the node represents a spatial location of an instance (*e.g.*, car), and the edges connected between two nodes represent the similarity of pixels. For events, if we denote the connection range (neighborhood size) as σ, then nearby events within σ (9 nodes in Fig. 1(b)) are considered for computing affinity contiguity. It is possible to adjust each node's granularity to control the size of the affinity graph; however, as events are sparse, we do not consider this factor. In such a way, we can aggregate top-σ nodes according to the spatial distances and represent the affinity feature of a certain node. For a feature map $F \sim \mathbb{R}^{C \times H \times W}$ ($H \times W$ is the spatial resolution and C is the number of channels), the affinity graph contains nodes with $H \times W \times \sigma$ connections. We denote A_{ab}^{EIT} and A_{ab}^{EEL} are the affinity graph between the a-th node and the b-th node obtained from the EIT and EEL branch, respectively, which is formulated as:

$$\mathcal{L}_{FL} = \frac{1}{H \times W \times \sigma} \sum_{a \sim R} \sum_{b \sim \sigma} ||A_{ab}^{EIT} - A_{ab}^{EEL}||_2^2 \quad (4)$$

where $R = \{1, 2, \cdots, H \times W\}$ indicates all the nodes in the graph. The similarity between two nodes, depicted as the pink lines in Fig. 1(b), is calculated from the aggregated features F_a and F_b as:

$$A_{ab} = \frac{F_a^\intercal F_b}{||F_a^\intercal||_2 ||F_b||_2} \quad (5)$$

where $F_a^\intercal$ is the transposed feature vector of F_b.

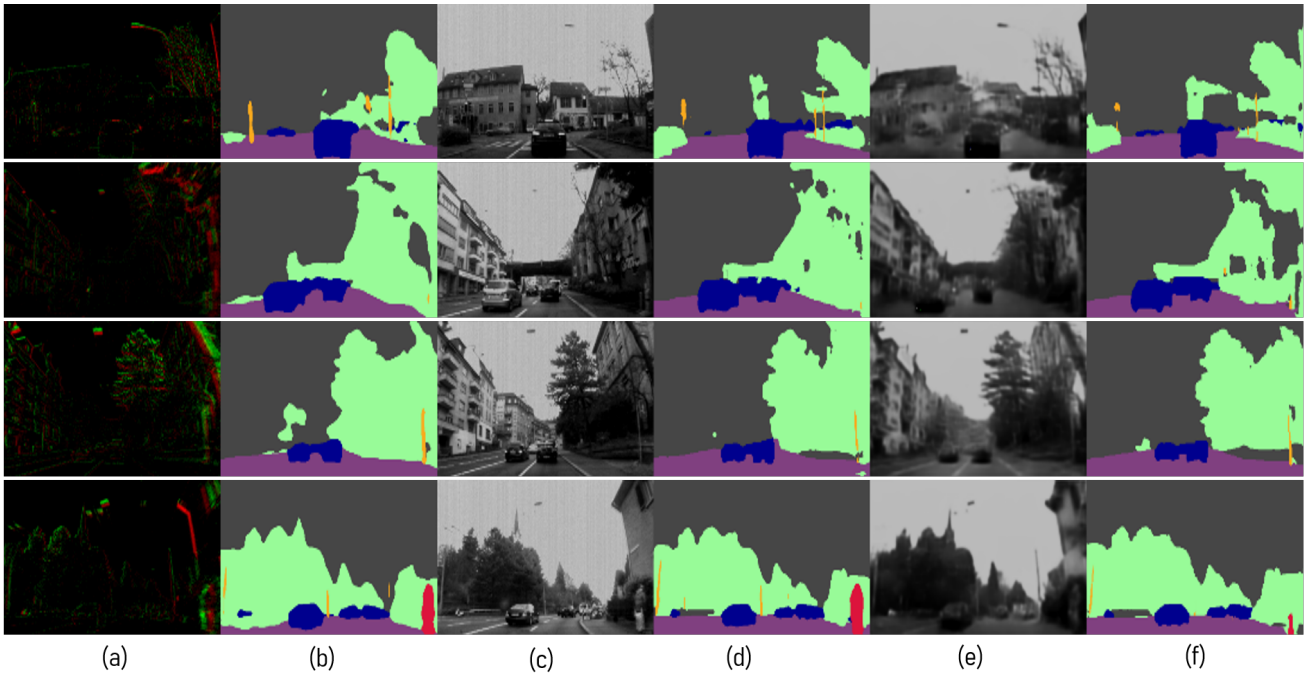

Figure 3: Qualitative results on DDD17 test sequence provided by [1]. (a) Events, (b) Segmentation results on events, (c) APS frames, (d) Segmentation results on APS frames, (e) Generated intensity images from events, (f) Pseudo GT labels.

3. Event-based Semantic Segmentation

Implementation details. We use the state-of-the-art DeepLabv3 (with ResNet101) [2] as the semantic segmentation network. The hyper-parameters λ_1, λ_2, λ_3 and λ_4 are set as 1, 1, 0.1 and 1, respectively. In the training, we set the learning rate as $1e-3$ and use the stochastic gradient descent (SGD) optimizer with weight decay rate of $5e-6$ to avoid overfiting. As the common classification accuracy is not well fit for semantic segmentation, we use the following metric to evaluate the performance, as done in the literature [2, 3]. The *intersection of union* (IoU) score is calculated as the ratio of intersection and union between the ground-truth mask and the predicted segmentation mask for each class. We use the *mean IoU* (MIoU) to measure the effectiveness of segmentation.

3.1. Evaluation on DDD17 dataset

General scene We first present the experimental results on the DDD17 dataset [1]. The visual results in Fig. 3 further verify the effectiveness of the proposed DTL framework. Overall, the segmentation results on events are comparable to those based on the APS frames, and some are even better than those based on the APS frames, *e.g.*, the 1st and 2nd rows. Meanwhile, our method generates convincing intensity images from EIT branch (5th column), The results indicate that, although events only reflect the edge information, our method successfully explores the feature-level and prediction-level knowledge to facilitate the end-task learning. The simple yet flexible approach brings a significant performance boost on the end-task learning.

Table 2: Segmentation performance of our method and the baseline on the test data DDD17 dataset, measured by MIoU. The baseline is trained using the pseudo labels made by the APS frames.

Method	Event Rep.	MIoU
Baseline-Deeplabv3	Multi-channel	50.92
Our (no GT)-Deeplabv3	Multi-channel	56.52 (+6.60)
Ours-Deeplabv3	Multi-channel	**58.80** (+ 7.88)

High dynamic range (HDR). HDR is one distinct advantage of an event camera. Even when APS frames are ill-exposed, events capture the intensity changes. We show the segmentation network shows promising performance in the extreme condition. The qualitative results are shown in Fig. 4. As can be seen, the APS frames (Fig. 4(d)) are over-exposed, making these images failed to be segmented by the segmentation network (as shown in Fig. 4(d)). However, events capture the scene details (as shown in Fig. 4(a)), which enables the segmentation network to successfully learn from these details and shows more convincing segmentation results. Meanwhile, with events, our method also reconstructs realistic intensity images with visual details that are lost in the APS frames.

Segmentation without using GT labels. With the EIT branch empowered by the teacher model, we show that

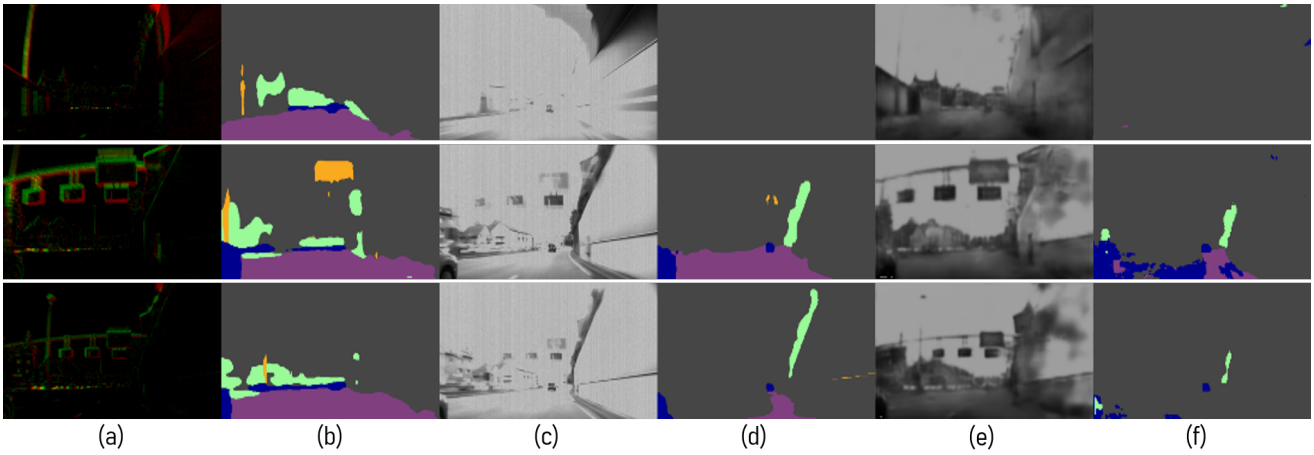

Figure 4: Semantic segmentation results **on the HDR scenes** of the DDD17 dataset. (a) Events, (b) Segmentation results on events, (c) APS frames, (d) Segmentation results on APS frames, (e) Generated intensity images from events, (f) Pseudo GT labels.

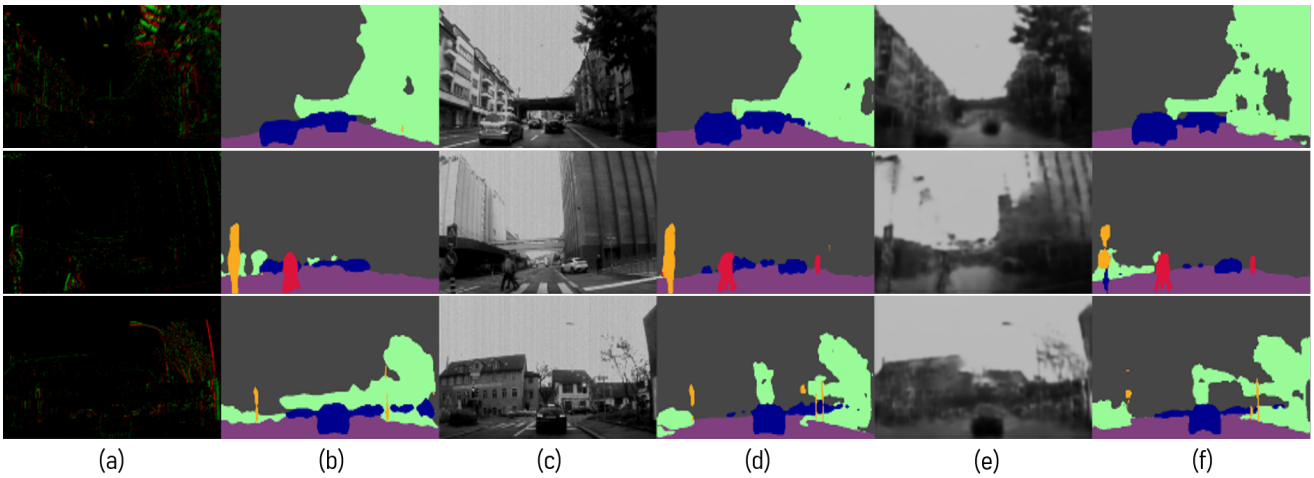

Figure 5: Semantic segmentation results **without using the GT labels** on the DDD17 dataset. (a) Events, (b) Segmentation results on events, (c) APS frames, (d) Segmentation results on APS frames, (e) Generated intensity images from events, (f) Pseudo GT labels.

our DTL framework can learn to segment events without using the semantic labels. The quantitative evaluation is given in Table 2 and the qualitative results are in Fig. 5. Numerically, even without using the ground truth labels, our method achieves 56.52% MIoU, which significantly enhances the semantic segmentation performance by 6.60% MIoU than the baseline. Compared with the SoTA methods [1, 5], our method still surpasses them with around 2% MIoU. The qualitative results also validate the numerical results.

3.2. Evaluation on MVSEC dataset

To further validate the effectiveness of the proposed DTL framework, we utilize the MVSEC dataset [21], which contains various driving scenes for 3D scene perception. As there are no semantic segmentation labels in this dataset, to quantitatively evaluate our method, we also utilize the APS frames to generate pseudo labels based on a network [2] pre-trained on the Cityscapes dataset (grayscale) [4], similar to [1], as our comparison baseline. Due to the poor quality of APS frames in the 'day1' sequence, we mainly use 'day2' sequence and divide the data into training (around 10K paired embedded events and APS frames) and test (378 paired embedded events and APS frames) sets based on the way of splitting DDD17 dataset in [1]. For the training data, we remove the redundant sequences, such as vehicles stopping in the traffic lights, etc. We also use the night driving sequences to show the advantage of events on HDR.

The qualitative results are shown in Fig. 6 and Table 3. In Fig. 6, we mainly show the results in the general condition. Using a multi-channel event representation in Table 3, the proposed DTL framework significantly surpasses the baseline by a noticeable margin with around 10.3% in-

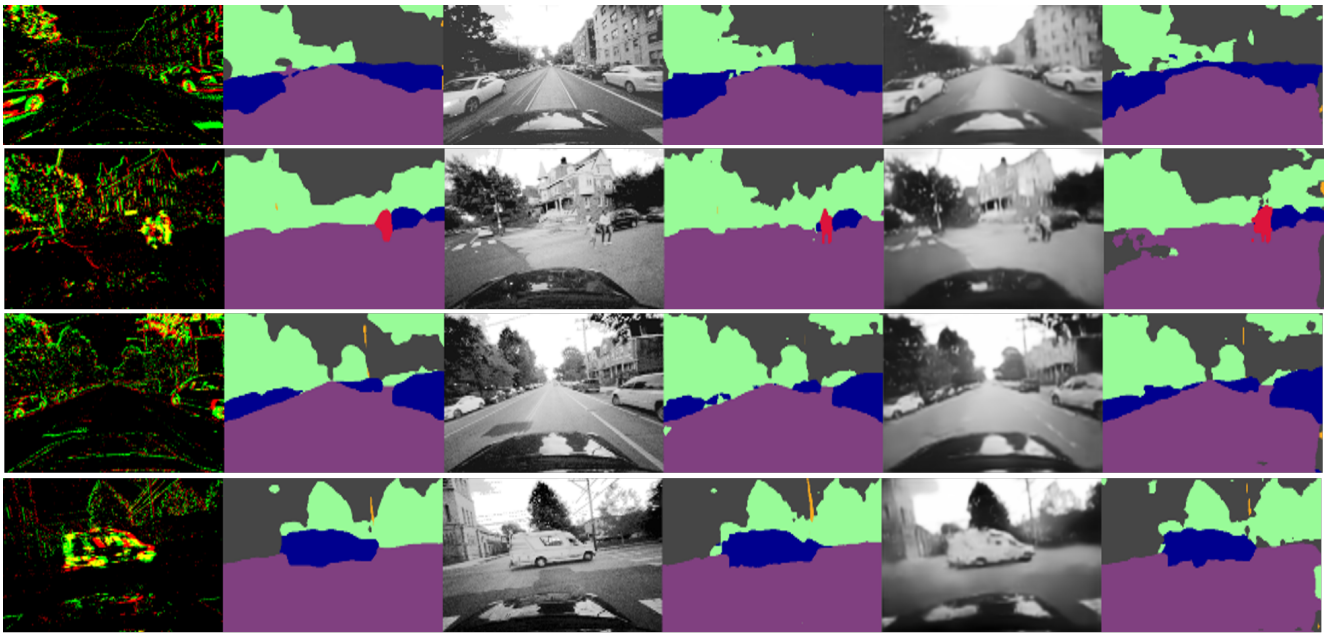

Figure 6: Qualitative results of semantic segmentation and image translation on the MVSEC dataset. (a) Events, (b) Segmentation results on events, (c) APS frames, (d) Segmentation results on APS frames, (e) Generated intensity images from events, (f) Pseudo GT labels.

Table 3: Segmentation performance of our method and the baseline on the test data [19], measured by MIoU. The baseline is trained using the pseudo labels made by the APS frames.

Method	Event Rep.	MIoU
Baseline-Deeplabv3	Multi-channel	50.53
Ours-Deeplabv3	Multi-channel	**60.82** (+ 10.29)

crease of MIoU. The results indicate a significant performance boost for semantic segmentation. The effectiveness can also be verified from visual results in Fig. 6. As can be seen, the semantic segmentation results (2nd column) are fairly convincing compared with the results on APS frames (4th column) and the pseudo GT labels (6th column). Meanwhile, our method also generates very realistic intensity images (5th column) from the EIT branch. The results on both semantic segmentation and image translation show that our the proposed DTL framework successfully exploit the knowledge from one branch to enhance the performance of the other.

Importance of Tanh function. In the main paper, we mentioned that using Tanh activation function at the last layer of the EIT decoder is very important. We now provide some visual examples to show the effectiveness. The visual comparison is shown in Fig. 7. As can be clearly verified, the generated images without using Tanh function are with noticeable artifacts, especially with black spots, as shown in Fig. 7(b). When the Tanh activation function is added, the artifacts are removed as shown in Fig. 7(c).

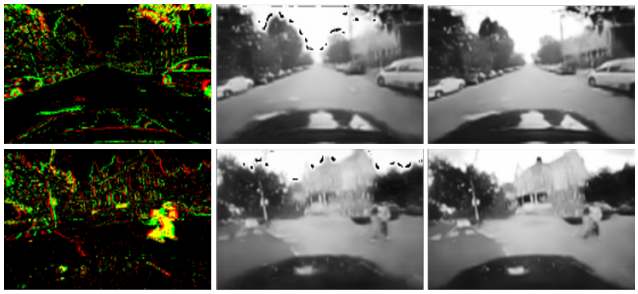

Figure 7: Visual comparison of the generated images **with and without using Tanh activation function** in the EIT decoder. (a) Events, (b) Generated images without Tanh function. (c) Generated images with Tanh activation functions.

4. Monocular Dense Depth Estimation

Event-based depth estimation is the task of predicting the depth of scene at each pixel in the image plane, and is important for various applications, *e.g.*, autonomous driving [19]. Previous works for event-based depth estimation have most focused on sparse or semi-dense depth estimation [10, 11, 12, 15, 18]. Recently, DNN has been applied to stereo events to generate dense depth predictions [14] and to estimate monocular semi-dense depth [21]. Some other works have focused on the dense depth estimation with only events [9] or with additional inputs [7]. We show

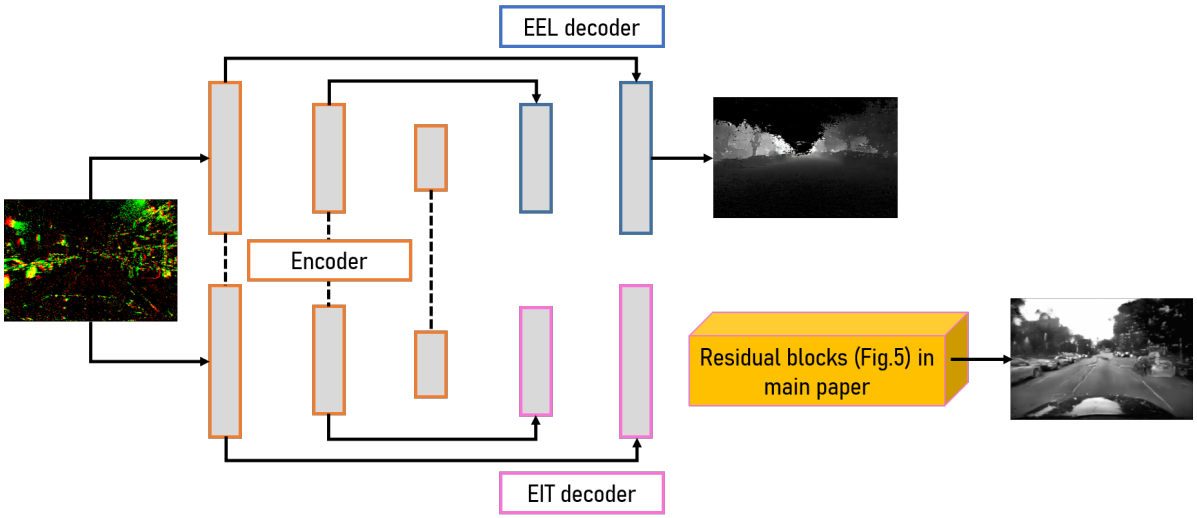

Figure 8: The proposed network structure for monocular dense depth estimation.

that the proposed DTL framework is also capable of predicting monocular dense depth from sparse event data.

Implementation details. We use Unet network structure, inspired by [8], as the depth estimation network. We then extend it by adding the event to image translation branch based on the depth estimation network, as shown in Fig. 8. The object function is exactly similar to the one (Eq. 6 in the main paper) for semantic segmentation except the supervision loss, which is changed to the regression loss (*e.g.*, L1 loss) instead of multi-classification cross-entropy loss. The hyper-parameters λ_1, λ_2, λ_3 and λ_4 are set as 100, 100, 1 and 20, respectively. In the training, we set the learning rate as $2e-4$ and use the Adam optimizer with weight decay rate of $5e-6$ to avoid overfiting. We use the following metric to evaluate the performance, as done in the literature [8, 21]. To evaluate the scale-invariant depth, we use the absolute relative error (Abs. Rel.), logarithmic mean squared error (RMSELog), scale invariant logarithmic error (SILog) and accuracy (Acc.). The mathematical formulations are as follows:

$$Acc. = \%\ of\ d_i\ s.t.\ \max(\frac{d_i}{d_i'}, \frac{d_i'}{d_i}) = \sigma < th, \quad (6)$$

$$SILog = \frac{1}{n}\Sigma a_i^2 - \frac{1}{n^2}(\Sigma a_i)^2, a_i = \log d_i - \log d_i', \quad (7)$$

$$Abs.Rel. = \frac{1}{n}\Sigma\frac{\|d - d'\|}{d'}, \quad (8)$$

$$RMSELog = \sqrt{\frac{1}{n}\Sigma\|\log d - \log d'\|^2}. \quad (9)$$

We present quantitative and qualitative results and compare with the baseline settings and prior methods [14, 21] with sparse event data as inputs on the MVSEC dataset [19]. We read the online available ROS bag data and represent the events with the representation method in Sec. 1, which are formed by two consecutive frames and ground truth depth labels. We use *outdoor_day2* sequence of the MVSEC dataset and select around 10K embedded event image and APS image pairs with their synchronized depth GT images to train the proposed DTL framework, similar to [14, 21]. We then utilize the *outdoor_day1* sequence (normal driving condition), *outdoor_night1*, *outdoor_night2* and *outdoor_night3* sequences (night driving condition) as the test sets. We train the DTL framework for 200 epochs and perform data augmentation by randomly scaling, cropping and flipping the training examples.

The additional qualitative results for HDR scene are shown in Fig. 9. As can be noticed from Fig. 9, the proposed DTL framework not only shows convincing performance on the depth prediction task but also reconstructs intensity frames with more detailed structures. Compared with the GT depth, the predicted depth better preserves the shapes and structures of objects, such as buildings, trees, cars, etc. Although having perfect alignment of depth with events is difficult to achieve in real-world data, our method predicts depth with sharp edges and shows more convincing results. Meanwhile, the DTL framework also enhances performance on image translation, where we can see the translated images are close to the APS frames. Our method shows a distinctive advantage on the HDR scene. As shown in Fig. 9, when the APS frames all fail to predict the correct depth information (6th column), events show very convincing depth estimation results. The EIT branch successfully generates realistic intensity images (3rd column) and shows better depth estimation results (5th column) than those of APS frames.

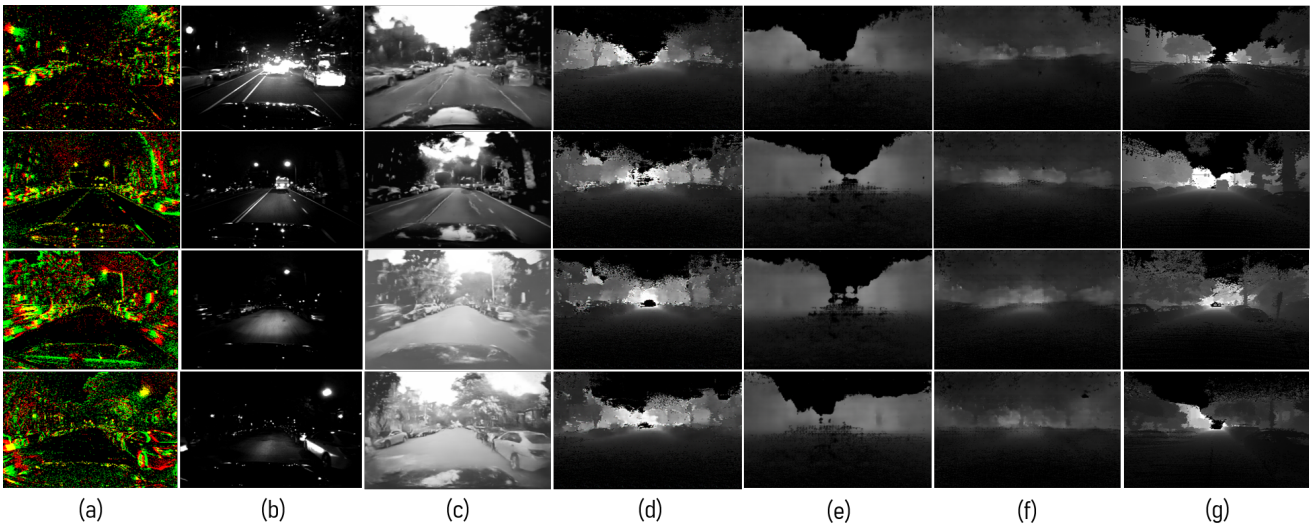

Figure 9: Qualitative results for monocular dense depth estimation **on the HDR scenes**. (a) Events, (b) Dark APS frames, (c) Generated intensity images, (d) Predicted depth on events, (e) Predicted depth on the generated intensity images, (f) Predicted depth on APS frames, (g) Depth GT.

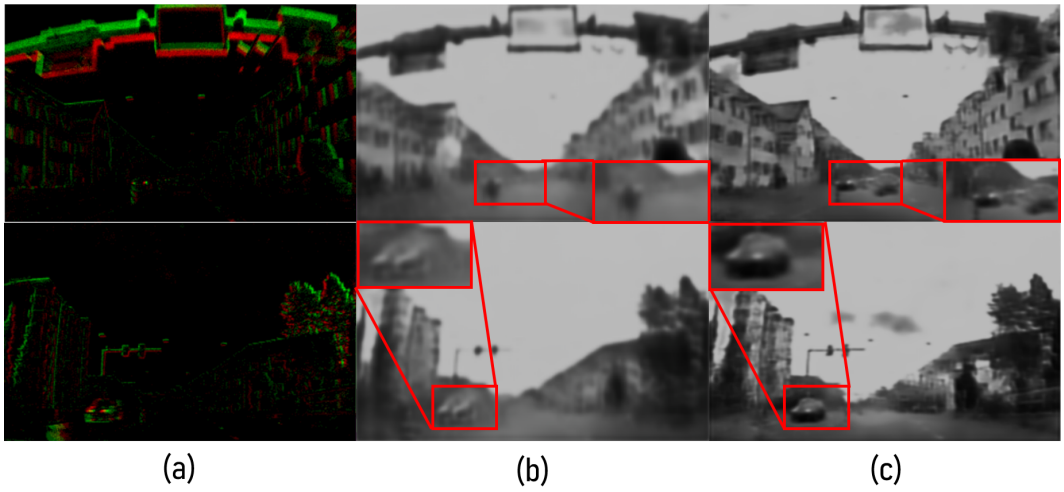

Figure 10: Impact of DTL on image translation. (a) Events, (b) Translated images without DTL, (c) Translated images with DTL.

5. Discussions

The effectiveness of TL module for EIT branch. Although the EIT branch is regarded as an *auxiliary* task in the proposed DTL framework, we show that it also benefits the EIT learning. We qualitatively compare the quality of translated images with and without using the DTL framework. Fig. 10 (enlarged one for showing better details) shows the visual results. In contrast to the generated images without DTL (2nd column), the results with DTL are shown to have more complete semantic information and better structural details, as shown in the cropped patches in the 3rd column. Interestingly, better structural details, *e.g.*, cars, buildings and trees, are restored. The experimental results show that our method works effectively on sparse events and are shown successful not only for the end-tasks but also for the image translation.

Value of this work for the community In this paper, we have demonstrated the effectiveness of the proposed DTL framework for event-based end-task learning. In the experimental results on semantic segmentation and depth estimation, the proposed DTL framework have achieved a significant performance boost than the existing methods and the baselines. Although our framework is concentrated on the event data, it is apparent that the proposed method is a general framework for other modality data, such as depth and thermal camera data. As event to image translation is way to represent events to the image domain, we take the feature-

level information as very important knowledge for learning better representation on the sparse events. This also applies to the depth and thermal camera data which are also sparse to some extent. In the future work, we plan to extend the proposed framework to other modality data, especially for demonstrating the high dynamic range imaging in the over-exposed and under-exposed illumination conditions.

References

[1] Inigo Alonso and Ana C Murillo. Ev-segnet: semantic segmentation for event-based cameras. In *CVPRW*, pages 0–0, 2019. 1, 3, 4

[2] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*. 3, 4

[3] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *ECCV*, 2018. 3

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, pages 3213–3223, 2016. 4

[5] Daniel Gehrig, Mathias Gehrig, Javier Hidalgo-Carrió, and Davide Scaramuzza. Video to events: Bringing modern computer vision closer to event cameras. *CVPR*, 2020. 4

[6] Daniel Gehrig, Antonio Loquercio, Konstantinos G Derpanis, and Davide Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *ICCV*, pages 5633–5643, 2019. 1, 2

[7] Daniel Gehrig, Michelle Rüegg, Mathias Gehrig, Javier Hidalgo Carrio, and Davide Scaramuzza. Combining events and frames using recurrent asynchronous multimodal networks for monocular depth prediction. *IEEE RA-L*, 2021. 5

[8] Rick Groenendijk, Sezer Karaoglu, Theo Gevers, and Thomas Mensink. On the benefit of adversarial training for monocular depth estimation. *CVIU*, 190:102848, 2020. 6

[9] Javier Hidalgo-Carrió, Daniel Gehrig, and Davide Scaramuzza. Learning monocular dense depth from events. *International Conference on 3D Vision*, 2020. 5

[10] Henri Rebecq, Guillermo Gallego, and Davide Scaramuzza. Emvs: Event-based multi-view stereo. 2016. 5

[11] Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization. 2017. 5

[12] 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. *RA-L*, 2(2):593–600, 2016. 5

[13] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *TPAMI*, 2019. 1, 2

[14] Stepan Tulyakov, Francois Fleuret, Martin Kiefel, Peter Gehler, and Michael Hirsch. Learning an event sequence embedding for dense event-based deep stereo. In *ICCV*, pages 1527–1537, 2019. 5, 6

[15] Antoni Rosinol Vidal, Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios. *RA-L*, 3(2):994–1001, 2018. 5

[16] Bishan Wang, Jingwei He, Lei Yu, Gui-Song Xia, and Wen Yang. Event enhanced high-quality image recovery. *ECCV*, 2020. 1, 2

[17] Lin Wang, S. Mohammad Mostafavi I., Yo-Sung Ho, and Kuk-Jin Yoon. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *CVPR*, pages 10081–10090, 2019. 1, 2

[18] Yi Zhou, Guillermo Gallego, Henri Rebecq, Laurent Kneip, Hongdong Li, and Davide Scaramuzza. Semi-dense 3d reconstruction with a stereo event camera. In *ECCV*, pages 235–251, 2018. 5

[19] 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. *RA-L*, 3(3):2032–2039, 2018. 5, 6

[20] 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. 2

[21] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *CVPR*, pages 989–997, 2019. 1, 2, 4, 5, 6