Spatio-temporal Event Feature Extractor for Event-based Stereo

Hyeokjun Kweon, Jaeseok Jeong, Sung-hoon Yoon, Yujeong Chae, Kuk-Jin Yoon
Visual Intelligence Laboratory, Department of Mechanical Engineering, KAIST, Korea
{0327june, jason.jeong, yoon307, yujeong, kjyoon}@kaist.ac.kr

1. Introduction

This technical report is submitted for the DSEC [2] competition in CVPR 2021 Workshop on Event-based Vision. The goal of the competition is to estimate dense disparity maps from event stream data obtained by event cameras in a stereo manner. The event stream data and the provided ground truth disparity maps are temporally synchronized but not spatially aligned. We use the rectification map provided in the dataset to rectify and align the event stream data. We would like to further note that the DSEC dataset does provide RGB images for the scene; however, our method is an event-only method that does not need to use these RGB images.

In this report, we primarily focus on the event embedding method and design a ConvLSTM [3]-based event feature extractor. The ConvLSTM module extracts not only spatial but also temporal information while processing the stacked event stream. We experimentally show that the designed event feature extractor can provide more valuable features to the stereo matching model than either the voxel-based [6] or queue-based method [4]. Furthermore, we employ a PSMNet as the backbone stereo matching model to improve performance. Ultimately, with only event stream data, we achieve 0.59 MAE in our train/validation split.

2. Methods

Raw event stream data is well-known for its complexity and its noise. Since its modality is quite different from that of the conventional image, proper pre-processing is required to enjoy the powerful performance of existing stereo matching models which were originally designed to handle conventional images.

To handle raw event stream data, we initially use the voxel grid representation that contains events within a certain duration as in [6]. We then design a ConvLSTM-based embedding method to extract spatio-temporal event features from the event voxel in a sequential manner, which can be helpful for the followed stereo matching network. Our approach will be explained in detail throughout this section.

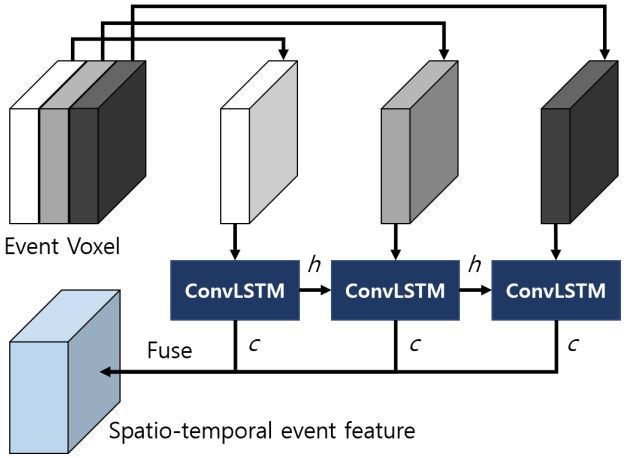

Figure 1: Diagram of ConvLSTM-based spatio-temporal event feature extractor. An event voxel grid is divided into three bins and they are fed to ConvLSTM in sequential manner. Processed cell states are fused into a spatio-temporal event feature, which will be used as an input to the followed stereo matching network. h and c denotes hidden state and cell state, respectively.

2.1. Event Data Embedding Method

The voxel grid representation [6] is an efficient and effective method to embed the sequence-like event stream into an image-like tensor; however, it cannot fully take into account the temporal nature of events. To estimate the disparity map at a certain timestamp, the model should pay more attention to the events obtained near that time.

In this regard, the voxel representation is able to provide an implicit way to process such temporal nature since each grid contains scaled timestamps. As an attempt to better process the temporal information, we also try a different representation. We generate a queue-like structure by gathering events in a First-In-First-Out (FIFO) manner for each pixel as similar with [4]. However, this queue-based approach requires a much longer computation time while the performance is worse than the aforementioned voxel embedding method based on our findings.

In this report, to efficiently gather the "valuable events" from the stacked voxel, we design a ConvLSTM-based

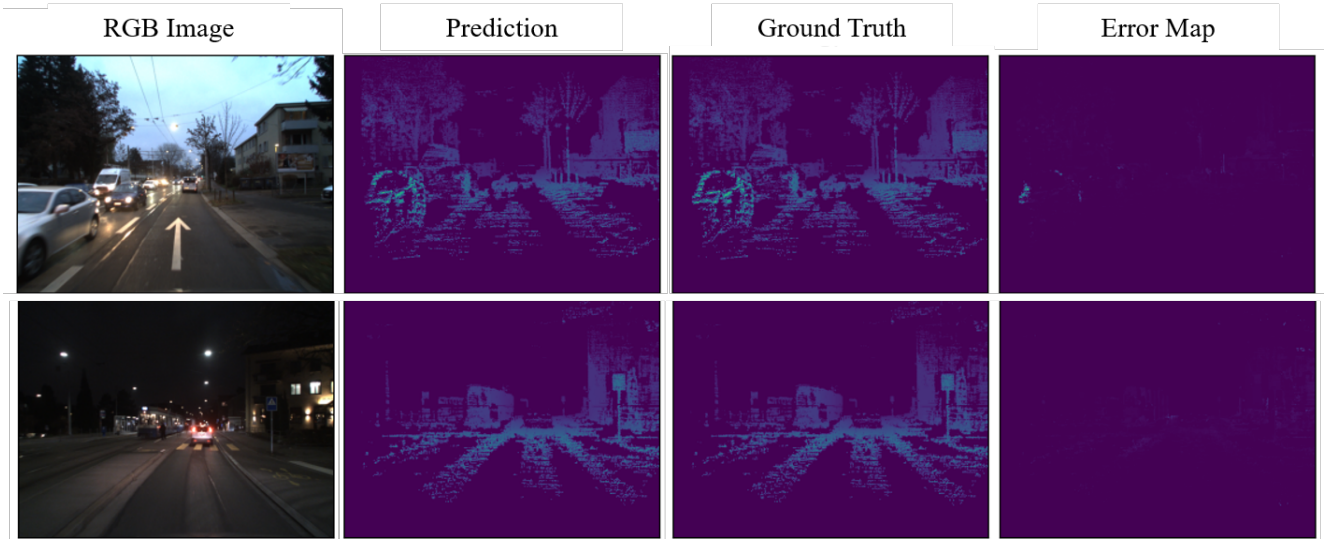

Figure 2: Qualitative comparison of the predicted and GT disparity map. From left to right: RGB images, Predictions of ours, ground-truths, and error maps. Note that our method only uses event data, and the RGB images are inserted for better explanation.

event feature extractor. In Fig. 1, we visualize the diagram of proposed ConvLSTM-based event feature extractor. We first divide the event voxel into multiple bins according to the temporal axis. The ConvLSTM then processes each bin while keeping the spatial dimension consistent. During the inference, each *hidden* state is propagated to the next ConvLSTM in a sequential manner. By fusing the cell state tensors with weighted sum, we extract an event feature that includes both spatial and temporal information.

In a stereo setting, we use separate feature extractors for the left and right event stream, since the event streams from left and right event cameras show different distributions. The resulting left/right spatio-temporal event features are fed into a stereo matching model, which will be explained in the following section.

2.2. Stereo Matching Method

For the stereo matching module, we test two networks that perform well on conventional stereo images: PSMNet [1] and AANet [5]. While using these networks as the backbone structure in conjunction with the event data embedding module, we empirically found that the performance of PSMNet is comparable to using AANet while being significantly faster to train in our system. As such, we employ PSMNet as our backbone stereo matching module.

3. Experiments and Result

3.1. Dataset and Evaluation Metrics

Since GT disparity maps for the test set are not provided, we split the provided train dataset into train/validation set. Among the 41 scenes, 36 scenes are selected as the train set and the remaining 5 scenes are selected as the validation set. Two of the validation set are scenes captured in night (low-lighting) condition. For the evaluation of the predicted disparity map, we use the MAE (Mean Absolute Error) metric. Because of the sparse nature of the Lidar sensor, we define an evaluation mask where the GT disparity is valid.

3.2. Training Details

For the training of event feature extractor and baseline network (PSMNet [1]), we use Adam optimizer for both. The learning rate is set to 0.001 and beta values are set to (0.9, 0.999). For the loss function, we use a smooth L1 loss following the work of PSMNet. Since PSMNet has a stacked hourglass architecture with three outputs and losses ($Loss_1$, $Loss_2$, $Loss_3$), we also use same weighting values ($0.5Loss_1 + 0.7Loss_2 + 1.0Loss_3$) for each loss. We trained the network with 25 epochs. For the event voxel, time duration for stacking (Δt) is set to 50ms. Before being fed into the ConvLSTM, the event voxel is split into 3 separate bins. The ConvLSTM module is composed of two convolutional layers where the channel dimension of both hidden state and cell state is set to 32. Note that the spatial resolution remains fixed to preserve the structural details, which are important for stereo matching.

3.3. Result

With the default setting (delta=50ms, bin=15), we test the performance of PSMNet. As shown in Table 1, MAE of the baseline network is 0.71. As aforementioned, we also test using AANet and it achieves almost similar performance to PSMNet. By using ConvLSTM for event embedding, we are able to achieve a lower MAE value (0.59)

Table 1: Performance (MAE, pixel) comparison of different event embedding methods and stereo matching models. Every experiment is conducted on our randomly split train/validation set.

Event embedding method	Stereo matching model	MAE
Voxel	AANet	0.72
Voxel	PSMNet	0.71
Queue	PSMNet	0.80
ConvLSTM (Ours)	PSMNet	**0.59**

than directly using event voxel grid as the stereo network input. We also show a qualitative comparison of the resulting disparity map in Fig. 2.

4. Conclusion

In this report, we discuss the event-based disparity map estimation method for the DSEC [2] competition. We design the ConvLSTM-based event feature extractor that considers both spatial and temporal nature of event stream data in a more efficient way than the voxel-based or queue-based method. While employing PSMNet as the backbone stereo matching model, we achieve 0.59 MAE in our train/validation split. We also show that the designed event-based stereo method can estimate a pretty accurate disparity map even under low-lighting conditions.

Though we simply focus on event embedding method in this report, there is still plenty of room to improve the stereo matching model for event-based stereo. It would be valuable and interesting for future works to design event-adaptive stereo matching model or frame-supported event-stereo method.

References

[1] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018. 2

[2] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. Dsec: A stereo event camera dataset for driving scenarios. *IEEE Robotics and Automation Letters*, 6(3):4947–4954, 2021. 1, 3

[3] 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. *arXiv preprint arXiv:1506.04214*, 2015. 1

[4] Stepan Tulyakov, Francois Fleuret, Martin Kiefel, Peter Gehler, and Michael Hirsch. Learning an event sequence embedding for dense event-based deep stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1527–1537, 2019. 1

[5] Haofei Xu and Juyong Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1959–1968, 2020. 2

[6] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 989–997, 2019. 1