

# Ev-3DOD: Pushing the Temporal Boundaries of 3D Object Detection with Event Cameras

## Supplementary Material

In this supplementary material, we offer more details of our work, Ev-3DOD. Specifically, we provide

- Implementation details in Section A;
- Runtime analysis of our framework, Ev-3DOD, in Section B;
- Detailed processing and annotation processes about event-based 3D detection datasets in Section C;
- Additional qualitative results and video demo in Section D;
- Hyper-parameter analysis in Section E;
- Visualization of event and voxel features in Section F;

### A. Implementation Details

The model is trained using a two-stage strategy inspired by previous works [9, 16] that leverages pre-trained encoders. In the first stage, 10 FPS LiDAR and images are utilized to train the active timestamp RGB-LiDAR Region Proposal Network. In the second stage, all sensor data are incorporated to train the blind time modules with 100 FPS ground truth bounding boxes. In the first stage, the Region Proposal Network is trained for 15 epochs with a batch size of 4 and a learning rate of 0.001, using the Adam optimizer [8] with the scheduling strategy from [12]. In the second stage, the blind time modules are trained for 15 epochs with a batch size of 3, maintaining the same learning rate of 0.001. The loss function incorporates weights  $\lambda_1 = 1.0$  and  $\lambda_2 = 1.0$ .

Since only the front camera is used, we followed the KITTI [6] methodology, utilizing only LiDAR point clouds and ground truth that fall within the camera’s field of view. The point cloud spans  $[0.0, 75.2m]$  along the  $X$  axis,  $[-75.2m, 75.2m]$  along the  $Y$  axis, and  $[-2m, 4m]$  along the  $Z$  axis, with a voxel size of  $(0.1m, 0.1m, 0.15m)$ . Ev-Waymo uses a resolution of  $960 \times 640$ , while DSEC-3DOD

adopts  $320 \times 240$  for both images and events using a down-sample. The event stream is converted into a voxel grid with 5 bins.

Waymo provides dense LiDAR data with 64 channels, whereas DSEC has only 16 channels, making it significantly sparser. Due to this sparsity, we attempted to accumulate LiDAR frames, but previous methods still struggled to train stably. Consequently, following prior works [14, 19], we used disparity maps to generate 3D points instead of directly utilizing raw LiDAR data. We acknowledge that these disparity-based 3D points are obtained through offline processing. However, the key focus of this work is not to achieve state-of-the-art performance using LiDAR but to demonstrate the feasibility of blind time object detection using event cameras. Therefore, using these 3D points does not pose an issue for our study. Nevertheless, to enhance the usability of DSEC-3DOD dataset for future research, we have conducted additional experiments using raw LiDAR data and have shared the results at the following link<sup>1</sup>.

The voxel data is encoded using a 3D backbone [20], while the image and event data are processed using a common image encoder [10]. The small version of our model is discussed in Section B. In the Virtual 3D Event Fusion module, each box proposal of size  $S \times S \times S$  is set to  $6 \times 6 \times 6$  sub-voxels.

### B. Inference Time

To measure the inference time of our method and other approaches, we followed the speed measurement protocol from conventional event-based object detection [4], using the code provided at the given link<sup>2</sup>. We also performed

<sup>1</sup><https://github.com/mickeykang16/Ev3DOD/tree/main/Benchmark>

<sup>2</sup><https://github.com/uzh-rpg/RVT>

Table A. Performance and runtime comparison on the Ev-Waymo dataset. Evaluated at 100 FPS for  $t = 0, 0.1, \dots, 0.9$ . Offline results, which rely on sensor data from timestamp 1, future information, and additional interpolation algorithms, are excluded from evaluation.

Methods	3D Detection Modality	ALL (mAP/mAPH)	VEH (AP/APH)		PED (AP/APH)		CYC (AP/APH)		FPS
		L2	L1	L2	L1	L2	L1	L2	
VoxelNeXt [2]	L	33.32/31.70	44.40/44.10	41.78/41.49	40.52/36.23	36.93/32.96	24.40/23.73	21.24/20.66	17.34
HEDNet [17]	L	31.57/29.90	42.03/41.71	39.32/39.02	38.86/34.43	35.67/31.53	22.64/21.99	19.72/19.14	12.84
Focals Conv [1]	L + I	26.27/25.01	37.31/37.01	36.60/36.31	29.20/26.16	28.41/25.44	14.30/13.78	13.79/13.29	6.08
LoGoNet [9]	L + I	33.27/31.75	44.14/43.87	41.73/41.47	39.98/35.84	36.48/32.67	24.71/24.15	21.59/21.10	10.68
Ev-3DOD (Ours)	L + I + E	<b>48.06/45.60</b>	<b>60.30/59.95</b>	<b>59.19/58.85</b>	<b>57.40/50.78</b>	<b>55.30/48.93</b>	<b>31.08/30.38</b>	<b>29.69/29.03</b>	<b>27.09</b>
Ev-3DOD- <i>Small</i> (Ours)	L + I + E	44.21/42.01	57.95/57.62	56.89/56.57	51.87/45.91	49.94/44.21	27.01/26.44	25.80/25.25	<b>54.14</b>

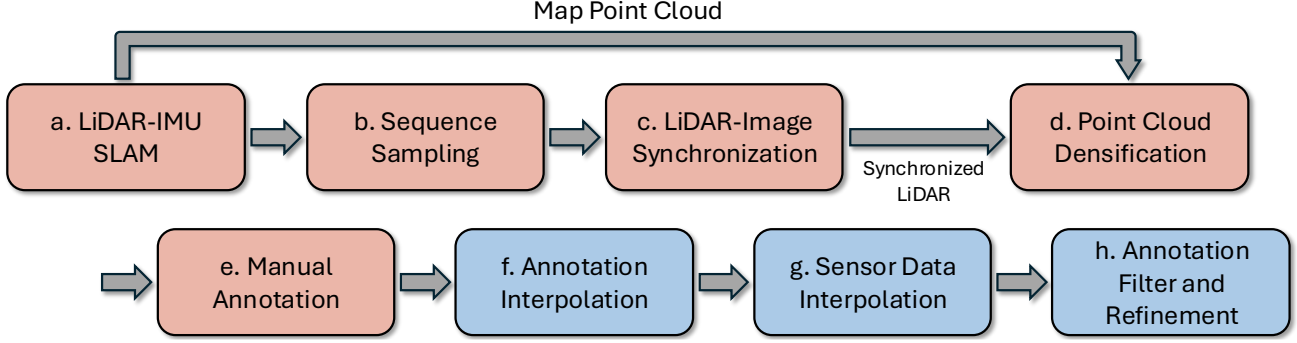


Figure A. The overall pipeline for annotation. To enhance data quality, we perform software-based alignment and generate fine-grained 100 FPS ground truths through additional annotation and post-processing. The **red** process is applied only to the DSEC-3DOD dataset, while the **blue** process is applied to both the DSEC-3DOD and Ev-Waymo datasets.

GPU warm-up and ensured GPU-CPU synchronization using ‘torch.cuda.synchronize()’ to accurately measure inference time. We measured inference time on a single NVIDIA A6000 GPU with a batch size of 1 and additionally designed a lightweight model, *Ev-3DOD-Small*, to evaluate both performance and speed. Specifically, *Ev-3DOD-Small* retains the overall structure of *Ev-3DOD* but replaces the event feature encoder with three simple convolution layers, reduces the number of pooling layers, and decreases the grid size in the Virtual 3D Event Fusion module.

Table A compares performance and inference time among online methods. Looking at the performance metrics, our method, leveraging the event camera to infer during the blind time, achieves the best and second-best results across both approaches, with a significant margin over other methods. In terms of inference time, measured in FPS, even our full model (*Ev-3DOD*) achieves the fastest speed compared to other methods. This efficiency is attributed to our approach, which avoids recalculating point clouds and images during the blind time interval by explicitly leveraging events to update and reuse data at the present moment, making it highly cost-effective. Notably, when parameters are reduced, there is almost a twofold improvement in FPS with minimal performance degradation. This demonstrates that our method can effectively estimate 3D motion using event information without relying on a large number of parameters. We believe that the proposed *Ev-3DOD*, with its fast runtime using the high-frequency properties of an event camera, provides a promising direction for advancing future research in 3D object detection using event cameras.

### C. Event-based 3D Object Detection Datasets

In this section, we provide additional details about the dataset that may not have been fully covered in the main paper. Specifically, we delve into its structure, pre-processing steps, and unique characteristics critical for understanding

the context and experimental results. By offering this comprehensive view, we aim to enhance the clarity and reproducibility of our work.

#### C.1. DSEC-3DOD Dataset

The DSEC [5] dataset provides LiDAR, stereo RGB images, and stereo events from diverse driving scenarios. To date, the DSEC dataset has been extensively studied for 2D perception tasks (*e.g.* 2D object detection, semantic segmentation). In this study, we utilized this dataset for 3D perception for the first time and established the process of Fig. A to provide fine-grained 100 FPS 3D detection ground truth.

##### a. LiDAR-IMU SLAM

For 3D detection annotation, a LiDAR sensor providing accurate depth information was designated as the reference sensor. The odometry of the reference sensor was estimated to synchronize LiDAR data with image timestamps, enabling accurate inter-modality alignment. Manual labeling was performed on a dense point cloud generated through pose-based LiDAR accumulation. To ensure precise LiDAR pose estimation, the LIO-Mapping [15] method was employed, consistent with the approach utilized in the DSEC dataset. The poses for the 10 FPS LiDAR data were subsequently obtained.

##### b. Sequence Sampling

As mentioned in the main paper, we provide annotations for the “zurich\_city” sequence. The DSEC provides images at 20 Hz and LiDAR data at 10 Hz. Images are sampled at 10 Hz following LiDAR. Although both the images and LiDAR data are sampled at the same 10 Hz rate, the lack of hardware time synchronization introduces temporal misalignment. To solve this problem, we utilize the sequence sampling strategy. Fig. B illustrates the absolute time difference between the nearest image and LiDAR frames. Due to the periodic discrepancy between the two sensors, this misalignment repeats approximately every 237 frames, with a

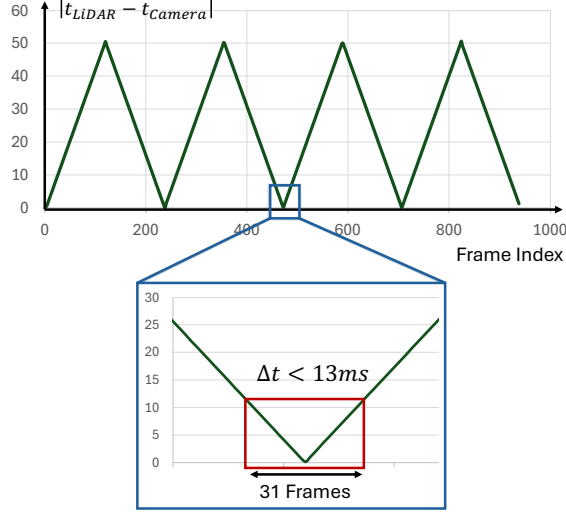


Figure B. The time difference between images and closest LiDAR. To minimize temporal misalignment, 31 frames were sampled around the frame with the minimum time offset.

maximum time offset of half a sensor period (*i.e.*, 50ms). Therefore, we sampled 31 frames centered around the point of minimal time offset, reducing the maximum misalignment of 13ms. As a result of sampling, the DSEC-3DOD dataset consists of 178 sequence chunks, each comprising 31 frames. Adjacent chunks are separated by a time gap of over 20 seconds, ensuring entirely different distributions in driving scenes. Table C and Table D show the train and test splits.

### c. LiDAR-Image Synchronization

Although the sequence sampling was designed to minimize time misalignment, synchronization errors still persist. Therefore, we further aligned the LiDAR data to the image timestamps using pose-based adjustments.

For an arbitrary RGB image  $I_t$  at time  $t$ , the two closest-time LiDAR point clouds,  $P_{t_0}$  and  $P_{t_1}$ , are identified, where  $t_0 < t < t_1$ . Assume the corresponding poses  $X_{t_0}$  and  $X_{t_1}$  are available from the mapping of Process a. Each pose is represented as a 3D coordinate and quaternion, denoted as  $X = (x, y, z, Q)$ , where  $Q \in \mathbb{R}^4$ . The image-aligned LiDAR pose  $X_t$  is computed by interpolating  $X_{t_0}$  and  $X_{t_1}$ . The position is interpolated linearly, while spherical linear interpolation (SLERP) [11] is applied for the quaternions.

The synchronized LiDAR point cloud  $P_t$  is obtained using the transformation  $P_t = T_t^{-1}T_{t'}P_{t'}$ , where  $T_t$  and  $T_{t'}$  are transformation matrices corresponding to the poses  $X_t$  and  $X_{t'}$ , respectively. Here,  $t'$  is the nearest time to  $t$  as,

$$t' = \begin{cases} t_0 & \text{if } |t - t_0| < |t - t_1|, \\ t_1 & \text{otherwise.} \end{cases} \quad (1)$$

The effect of pose-based LiDAR synchronization is demonstrated in Fig. C. LiDAR-image misalignment due to

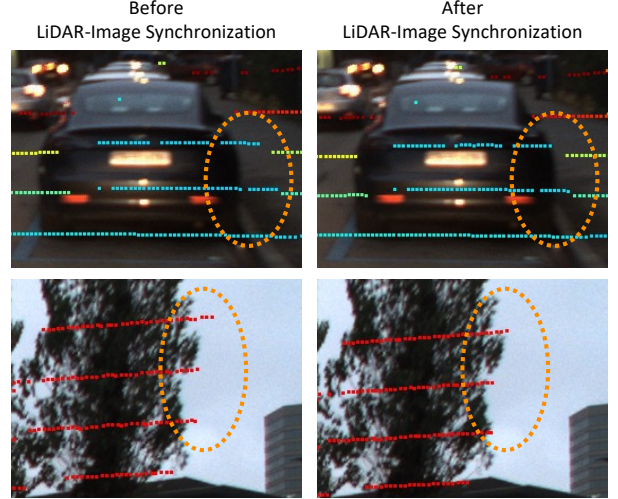


Figure C. LiDAR projection on images before pose-based LiDAR synchronization (left) and after synchronization (right).

time offsets is most evident in scenes with large motions or significant rotations. By transforming the nearest LiDAR point cloud to the pose of the image timestamp, projection errors between sensors caused by time misalignment were minimized.

### d. Point Cloud Densification

Raw LiDAR data is inherently sparse, which can lead to reduced accuracy in ground truth bounding boxes if used directly for annotation. To address this, we utilized an accumulated point cloud created by combining multiple LiDAR scans with their relative poses. As noted in the DSEC [5], LiDAR accumulation does not effectively handle occlusions or moving objects, both of which are critical considerations in the labeling process. To mitigate these issues, we employed filtered results (*i.e.*, disparity) during the annotation process.

### e. Manual Annotation

Annotation experts labeled 3D bounding boxes on the densified 10 FPS LiDAR data, ensuring accuracy by considering both the LiDAR data and images. The ground truth consists of three classes: vehicle, pedestrian, and cyclist. Detailed annotation rules were derived from the guidelines provided by the Waymo Open Dataset [13]. To maintain high-quality annotations, bounding boxes containing fewer than two points or positioned beyond 50 meters were excluded from the ground truth.

### f. Annotation Interpolation

To generate 100 FPS annotations from the manually labeled 10 FPS annotations, bounding boxes were interpolated to create ground truth for blind times where neither LiDAR nor images were available. Linear interpolation was applied to the bounding box pose and size, while SLERP interpolation was applied for rotations. The interpolated annotations were subsequently refined through the following process.

#### g. Sensor Data Interpolation

In process **f**, the automatically interpolated annotations are generally accurate but may not fully capture the dynamics of real-world motion. To enhance the quality of these annotations, sensor data was generated for the blind times. For images, realistic video frames were produced using a recent event-based video frame interpolation method [7]. Similarly, the latest techniques [18] were utilized to generate accurate and realistic intermediate point cloud data.

#### h. Annotation Filter and Refinement

The interpolated sensor data was employed to refine the annotations for the blind times. In most cases with minimal motion, bounding box interpolation alone yielded ground truth that aligned well with the sensor data. In such instances, refinements were avoided to preserve smooth bounding box poses and ensure temporal consistency. However, if the interpolated labels were misaligned with the sensor data or if sensor data was unavailable, the affected labels were filtered out.

### C.2. Ev-Waymo Dataset

**Event Synthesis.** The Waymo Open Dataset (WOD) provides 10 FPS synchronized images, LiDAR, and 3D bounding box labels. To generate the events, we utilize a widely adopted event simulation model [3] to synthesize the events from video data. This enables us to utilize temporally dense events between image and LiDAR active timestamps for training and testing.

**Annotation Interpolation and Refinement.** Since WOD provides dense 10 FPS annotations, we can obtain 100 FPS ground truth through annotation interpolation and refinement. A process similar to the **f**, **g**, and **h** steps in DSEC-3DOD dataset processing was employed. We interpolated the 10 FPS bounding box information, including pose, dimensions, and heading, provided by WOD to generate 100 FPS data. In addition, we synthesized blind-time data of camera and LiDAR using an interpolation algorithm for refinement and filtering, ensuring higher quality.

### D. More Qualitative Results and Videos

To provide a more comprehensive understanding of the proposed model, we present additional qualitative results in Figures **E**, **F**, and **G**, showcasing its performance on the DSEC-3DOD dataset. The proposed method consistently predicts bounding boxes that closely align with the ground truth across various challenging environments.

Figure **E** illustrates a challenging scene involving a bus, where size estimation is particularly difficult. The offline method, even with access to future information, fails to detect certain instances. Likewise, the online method demonstrates increasing errors compared to the ground truth as time progresses beyond the active timestamp. In contrast,

the proposed method shows robust performance, generating predictions that closely align with the ground truth labels.

The introduced DSEC-3DOD dataset features challenging scenarios, including night scenes, as shown in Fig. **F**. Unlike the Ev-Waymo dataset, which cannot leverage challenging illumination conditions to generate synthetic events, our proposed real event dataset enables validation in such scenarios. In the night scene, the proposed method exhibits accurate box predictions compared to both the offline and online methods.

Figure **H**, **I**, and **J** present the results on the Ev-Waymo dataset, which features numerous complex sequences with high object density. The proposed model effectively predicts complex object motions, producing bounding boxes closely aligned with the ground truth. In such scenes, even with access to future data, interpolating sensor information during blind times remains challenging, which complicates precise bounding box predictions. Consequently, the qualitative results on Ev-Waymo demonstrate that the proposed method outperforms others by delivering more accurate bounding boxes.

We provide a short video to showcase the datasets used in the experiments and the results on sequential data. The proposed method demonstrates robust performance across various environments in both the DSEC-3DOD dataset and the Ev-Waymo dataset. Notably, it accurately estimates the motion of object bounding boxes even in challenging night scenes within the DSEC-3DOD dataset.

The proposed model performs 3D detection during a single blind time using data from a single active timestamp and an event stream. When new LiDAR and RGB data become available, the model relies on the most recent data. Consequently, discontinuities in 3D detection may occur at each active timestamp. In this paper, we lay the foundation for a methodology that combines conventional sensors and events for blind time 3D detection. However, in further research, incorporating past information could improve both accuracy and temporal consistency.

### E. Hyper-parameter Analysis

As shown in Table **B**, we conduct an ablation study on the loss weights. The box regression loss and confidence prediction loss weights were set to 0.1, 1.0, and 10.0, respectively, during model training. The results demonstrated that the model remained robust, producing consistent outcomes despite changes in the loss magnitudes.

### F. Visualization of Event and Voxel Features.

To analyze the role of each modality visually, we visualized event and voxel features in Fig. **D**. When visualizing the event features in 2D, they primarily activate along edges, with particularly strong activations on moving objects. Ad-



Table B. The result according to hyper-paramter in Eq. (3) on Ev-Waymo LEVEL 2 (L2).  $\lambda_1$  and  $\lambda_2$  refer to the weight of box regression loss and binary cross entropy loss, respectively.

$\lambda_1 \setminus \lambda_2$	0.1		1.0		10.0	
	mAP	mAPH	mAP	mAPH	mAP	mAPH
0.1	47.32	44.87	47.70	45.22	47.14	44.70
1.0	47.72	45.25	48.06	45.60	46.98	44.53
10.0	47.46	45.02	47.20	44.74	47.32	44.86

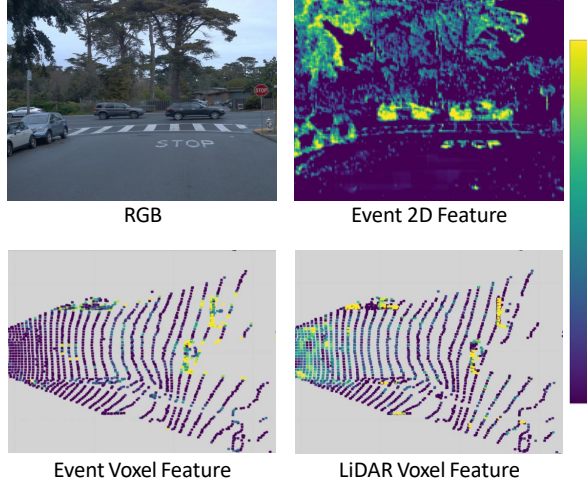


Figure D. Visualization of features generated from event and LiDAR.

ditionally, when comparing event and LiDAR features by projecting them into 3D, we observe that LiDAR features are highly activated across all regions containing 3D information, whereas 3D event features remain predominantly activated around moving objects. The event feature visualization highlights regions with frequent events around moving objects, effectively capturing dynamic areas of interest. Thus, by effectively leveraging multi-modal features from events and pointclouds, it is visually evident that 3D motion can be reliably estimated.

## References

- [1] Yukang Chen, Yanwei Li, Xiangyu Zhang, Jian Sun, and Jiaya Jia. Focal sparse convolutional networks for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5428–5437, 2022. 1
- [2] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, and Jiaya Jia. Voxelnex: Fully sparse voxelnex for 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21674–21683, 2023. 1
- [3] Daniel Gehrig, Mathias Gehrig, Javier Hidalgo-Carri6, and Davide Scaramuzza. Video to events: Recycling video datasets for event cameras. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020. 4
- [4] Mathias Gehrig and Davide Scaramuzza. Recurrent vision transformers for object detection with event cameras. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13884–13893, 2022. 1
- [5] 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. 2, 3
- [6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. 1
- [7] Taewoo Kim, Yujeong Chae, Hyun-Kurl Jang, and Kuk-Jin Yoon. Event-based video frame interpolation with cross-modal asymmetric bidirectional motion fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18032–18042, 2023. 4, 9, 10, 11
- [8] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1
- [9] Xin Li, Tao Ma, Yuenan Hou, Botian Shi, Yuchen Yang, Youquan Liu, Xingjiao Wu, Qin Chen, Yikang Li, Yu Qiao, et al. Logonet: Towards accurate 3d object detection with local-to-global cross-modal fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17524–17534, 2023. 1
- [10] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021. 1
- [11] Ken Shoemake. Animating rotation with quaternion curves. *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, 1985. 3
- [12] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, pages 369–386. SPIE, 2019. 1
- [13] Pei Sun, Henrik Kretschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott M. Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2443–2451, 2019. 3
- [14] Zhexiong Wan, Yuxin Mao, Jing Zhang, and Yuchao Dai. Rpeflow: Multimodal fusion of rgb-pointcloud-event for joint optical flow and scene flow estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10030–10040, 2023. 1
- [15] Haoyang Ye, Yuying Chen, and Ming Liu. Tightly coupled 3d lidar inertial odometry and mapping. *2019 International Conference on Robotics and Automation (ICRA)*, pages 3144–3150, 2019. 2
- [16] Junbo Yin, Jianbing Shen, Runnan Chen, Wei Li, Ruigang Yang, Pascal Frossard, and Wenguan Wang. Is-fusion: Instance-scene collaborative fusion for multimodal 3d object detection. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14905–14915, 2024. 1
- [17] Gang Zhang, Chen Junnan, Guohuan Gao, Jianmin Li, and Xiaolin Hu. Hednet: A hierarchical encoder-decoder network for 3d object detection in point clouds. *Advances in Neural Information Processing Systems*, 36, 2024. 1
- [18] Zehan Zheng, Danni Wu, Ruizi Lu, Fan Lu, Guang Chen, and Changjun Jiang. Neuralpci: Spatio-temporal neural field for 3d point cloud multi-frame non-linear interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 909–918, 2023. 4
- [19] Hanyu Zhou, Yi Chang, and Zhiwei Shi. Bring event into rgb and lidar: Hierarchical visual-motion fusion for scene flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26477–26486, 2024. 1
- [20] Yin Zhou and Oncel Tuzel. Voxelnex: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018. 1

Table C. The train sequence splits of DSEC-3DOD dataset.

Split	Time	Sequence	# Frames	# GT Scenes
Train	Day	zurich_city_00_a	31	301
		zurich_city_00_b	155	1,505
		zurich_city_01_a	62	602
		zurich_city_01_b	155	1,505
		zurich_city_01_c	124	1,204
		zurich_city_01_d	93	903
		zurich_city_01_e	217	2,107
		zurich_city_01_f	155	1,505
		zurich_city_02_a	31	301
		zurich_city_02_b	124	1,204
		zurich_city_02_c	279	2,709
		zurich_city_02_d	155	1,505
		zurich_city_02_e	186	1,806
		zurich_city_04_a	93	903
		zurich_city_04_c	93	903
		zurich_city_04_d	93	903
		zurich_city_04_e	31	301
		zurich_city_04_f	124	1,204
		zurich_city_05_a	217	2,107
		zurich_city_05_b	124	1,204
		zurich_city_06_a	186	1,806
		zurich_city_07_a	124	1,204
		zurich_city_08_a	62	602
		zurich_city_11_a	31	301
		zurich_city_11_b	93	903
		zurich_city_11_c	155	1,505
	Day Total		3,193	31,003
	Night	zurich_city_03_a	62	602
		zurich_city_09_a	217	2,107
		zurich_city_09_b	31	301
		zurich_city_09_c	155	1,505
		zurich_city_09_d	124	1,204
		zurich_city_09_e	93	903
		zurich_city_10_a	248	2,408
		zurich_city_10_b	217	2,107
	Night Total		1,147	11,137
	Train Total		4,340	42,140

Table D. The test sequence splits of DSEC-3DOD dataset.

Split	Time	Sequence	# Frames	# GT Scenes
Test	Day	zurich_city_00_a	62	602
		zurich_city_00_b	31	301
		zurich_city_01_e	31	301
		zurich_city_01_f	62	602
		zurich_city_02_b	31	301
		zurich_city_02_c	93	903
		zurich_city_02_d	62	602
		zurich_city_04_c	31	301
		zurich_city_04_d	31	301
		zurich_city_05_b	62	602
		zurich_city_06_a	31	301
		zurich_city_07_a	62	602
		zurich_city_08_a	31	301
		zurich_city_11_b	155	1,505
		zurich_city_11_c	93	903
	Day Total		868	8,428
	Night	zurich_city_03_a	31	301
		zurich_city_09_a	31	301
		zurich_city_09_c	31	301
		zurich_city_09_d	93	903
		zurich_city_10_a	31	301
		zurich_city_10_b	93	903
	Night Total		310	3,010
	Test Total		1178	11,438

Table E. The train/test sequence splits of Ev-Waymo dataset.

Dataset	Ev-Waymo		
Split	Sequence Name	No. Seq.	No. Labeled Scenes
Train	segment-207754730878135627_1140_000_1160_000, segment-13840133134545942567_1060_000_1080_000, segment-8327447186504415549_5200_000_5220_000, segment-10964956617027590844_1584_680_1604_680, segment-11918003324473417938_1400_000_1420_000, segment-15448466074775525292_2920_000_2940_000, segment-14830022845193837364_3488_060_3508_060, segment-11379226583756500423_6230_810_6250_810, segment-7861168750216313148_1305_290_1325_290, segment-13506499849906169066_120_000_140_000, segment-6229371035421550389_2220_000_2240_000, segment-15882343134097151256_4820_000_4840_000, segment-14098605172844003779_5084_630_5104_630, segment-8582923946352460474_2360_000_2380_000, segment-16485056021060230344_1576_741_1596_741, segment-915935412356143375_1740_030_1760_030, segment-3002379261592154728_2256_691_2276_691, segment-4348478035380346090_1000_000_1020_000, segment-2036908808378190283_4340_000_4360_000, segment-15844593126368860820_3260_000_3280_000, segment-5835049423600303130_180_000_200_000, segment-15696964848687303249_4615_200_4635_200, segment-7543690094688232666_4945_350_4965_350, segment-16372013171456210875_5631_040_5651_040, segment-14193044537086402364_534_000_554_000, segment-550171902340535682_2640_000_2660_000, segment-4641822195449131669_380_000_400_000, segment-7239123081683545077_4044_370_4064_370, segment-11928449532664718059_1200_000_1220_000, segment-5100136784230856773_2517_300_2537_300, segment-13182548552824592684_4160_250_4180_250, segment-14004546003548947884_2331_861_2351_861, segment-2570264768774616538_860_000_880_000, segment-7440437175443450101_94_000_114_000, segment-15717839202171538526_1124_920_1144_920, segment-8148053503558757176_4240_000_4260_000, segment-16977844994272847523_2140_000_2160_000, segment-5451442719480728410_5660_000_5680_000, segment-7290499689576448085_3960_000_3980_000, segment-16801666784196221098_2480_000_2500_000, segment-4916527289027259239_5180_000_5200_000, segment-16202688197024602345_3818_820_3838_820, segment-9758342966297863572_875_230_895_230, segment-12161824480686739258_1813_380_1833_380, segment-14369250836076988112_7249_040_7269_040, segment-2752216004511723012_260_000_280_000, segment-10444454289801298640_4360_000_4380_000, segment-17388121177218499911_2520_000_2540_000, segment-7885161619764516373_289_280_309_280, segment-16561295363965082313_3720_000_3740_000, segment-11199484219241918646_2810_030_2830_030, segment-4575961016807404107_880_000_900_000, segment-7566697458525030390_1440_000_1460_000, segment-10275144660749673822_5755_561_5775_561, segment-6193696614129429757_2420_000_2440_000, segment-12251442326766052580_1840_000_1860_000, segment-13271285919570645382_5320_000_5340_000, segment-9015546800913584551_4431_180_4451_180, segment-10596949720463106554_1933_530_1953_530, segment-15942468615931009553_1243_190_1263_190, segment-15125792363972595336_4960_000_4980_000, segment-1422926405879888210_51_310_71_310, segment-5576800480528461086_1000_000_1020_000, segment-1255991971750044803_1700_000_1720_000	64	126,330
Test	segment-18446264979321894359_3700_000_3720_000, segment-17152649515605309595_3440_000_3460_000, segment-16213317953898915772_1597_170_1617_170, segment-5183174891274719570_3464_030_3484_030, segment-3126522626440597519_806_440_826_440, segment-3077229433993844199_1080_000_1100_000, segment-10289507859301986274_4200_000_4220_000, segment-30779396576054160_1880_000_1900_000, segment-9243656068381062947_1297_428_1317_428, segment-2834723872140855871_1615_000_1635_000, segment-2736377008667623133_2676_410_2696_410, segment-15948509588157321530_7187_290_7207_290, segment-9231652062943496183_1740_000_1760_000, segment-4854173791890687260_2880_000_2900_000, segment-6324079979569135086_2372_300_2392_300, segment-6001094526418694294_4609_470_4629_470	16	31,550



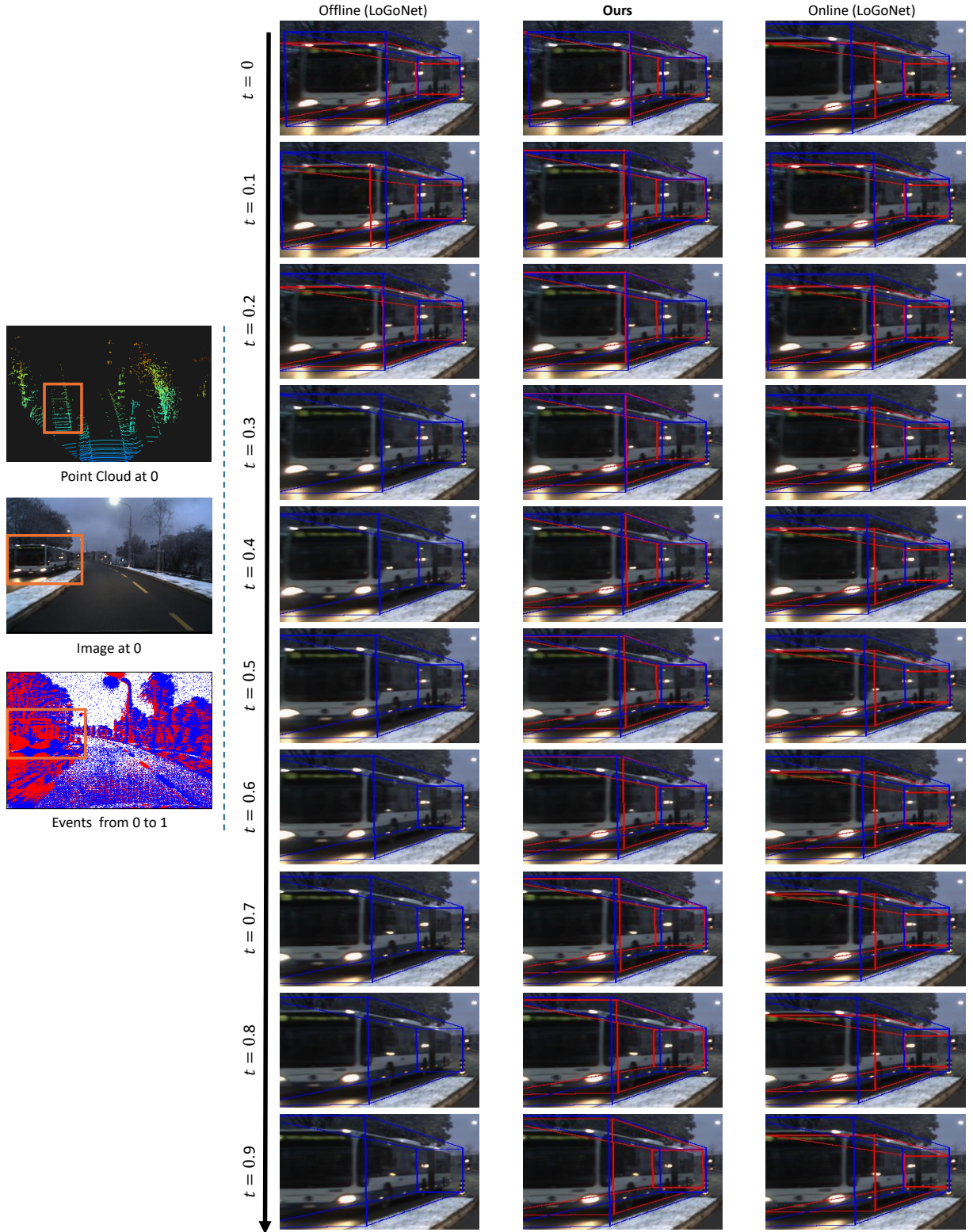


Figure E. Qualitative comparisons with other offline and online methods on the DSEC-3DOD dataset.  $t = 0$  represents the active time, while  $t = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$  denote the blind times. The **blue** box indicates ground truth, and the **red** box shows predictions. For better understanding, we overlaid the results onto the images generated by the interpolation method [7].

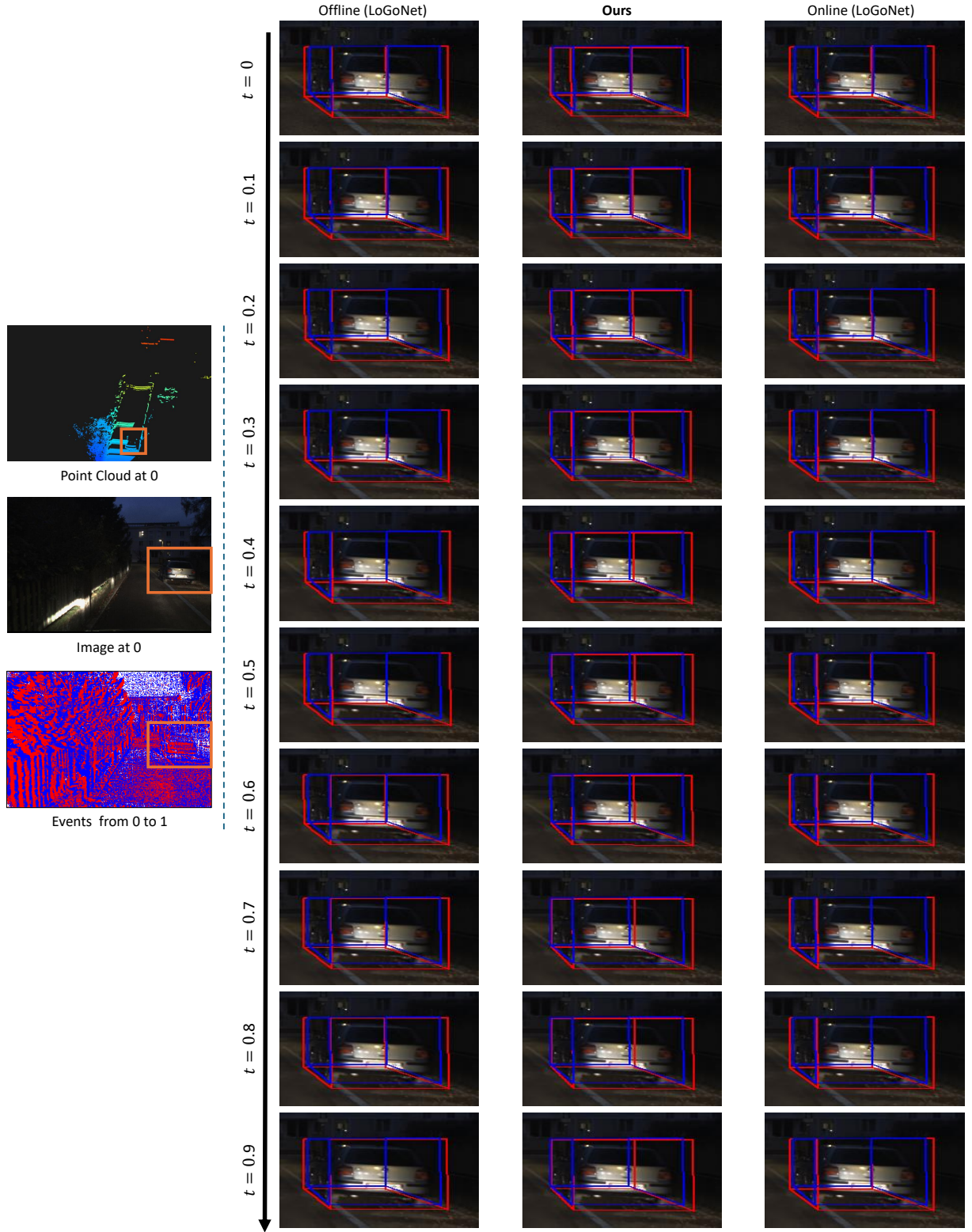


Figure F. Qualitative comparisons with other offline and online methods on the DSEC-3DOD dataset.  $t = 0$  represents the active time, while  $t = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$  denote the blind times. The blue box indicates ground truth, and the red box shows predictions. For better understanding, we overlaid the results onto the images generated by the interpolation method [7].



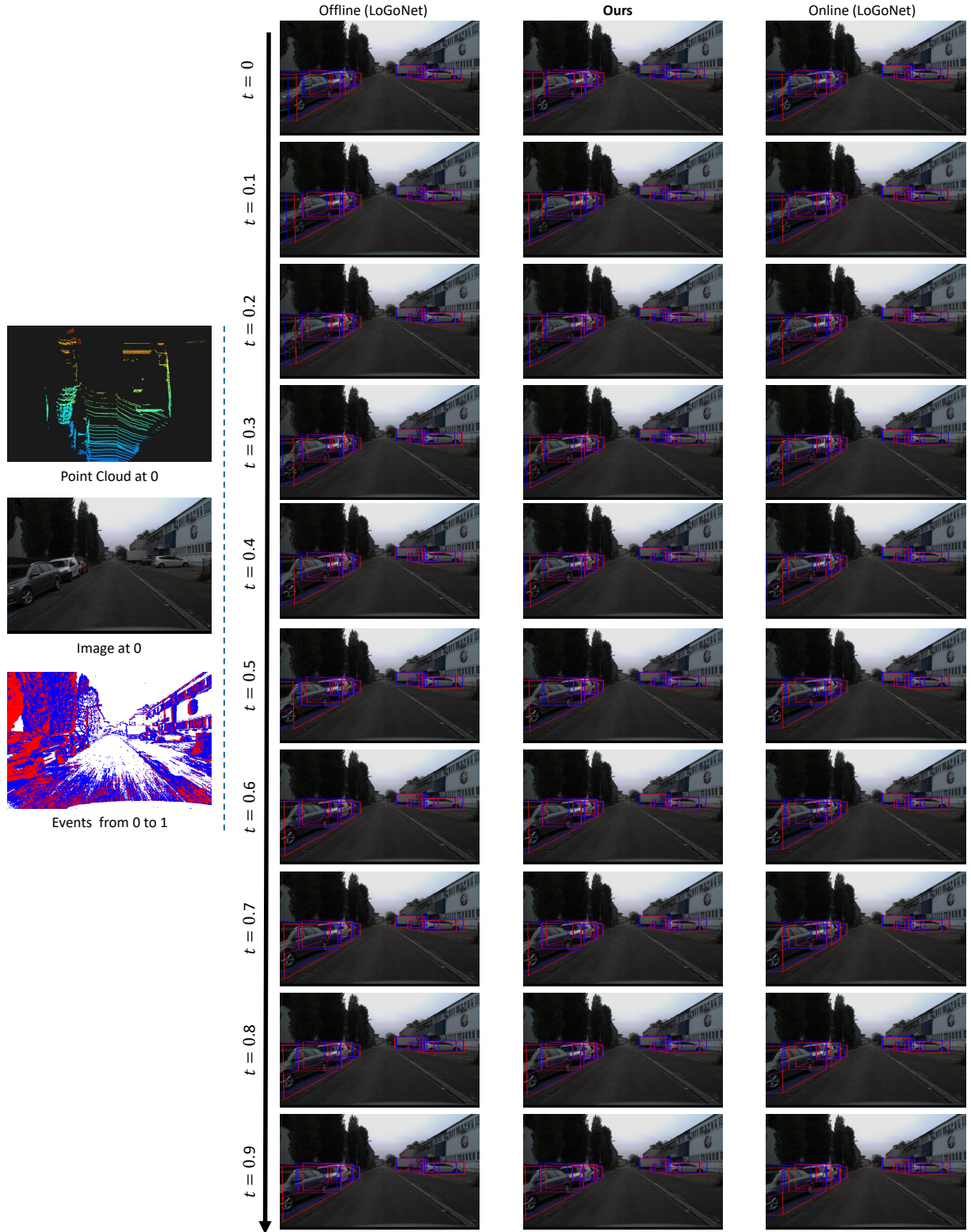


Figure G. Qualitative comparisons with other offline and online methods on the DSEC-3DOD dataset.  $t = 0$  represents the active time, while  $t = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$  denote the blind times. The blue box indicates ground truth, and the red box shows predictions. For better understanding, we overlaid the results onto the images generated by the interpolation method [7].

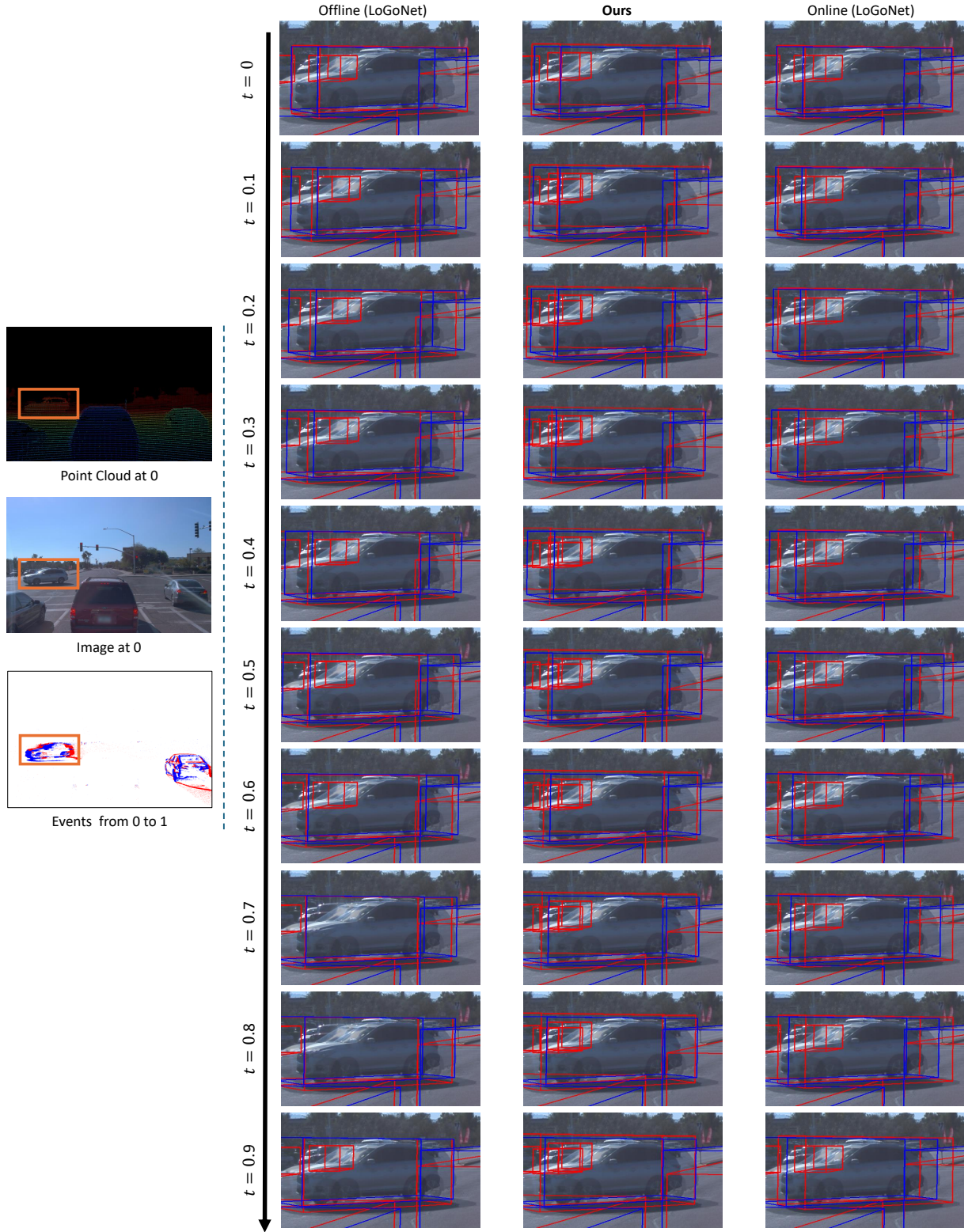


Figure H. Qualitative comparisons of our method with other offline and online evaluations on the Ev-Waymo dataset.  $t = 0$  represents the active time, while  $t = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$  denote the blind times. The blue box represents the ground truth, while the red box shows the prediction results of each method. For easier understanding, images at active timestamps 0 and 1 are overlaid.



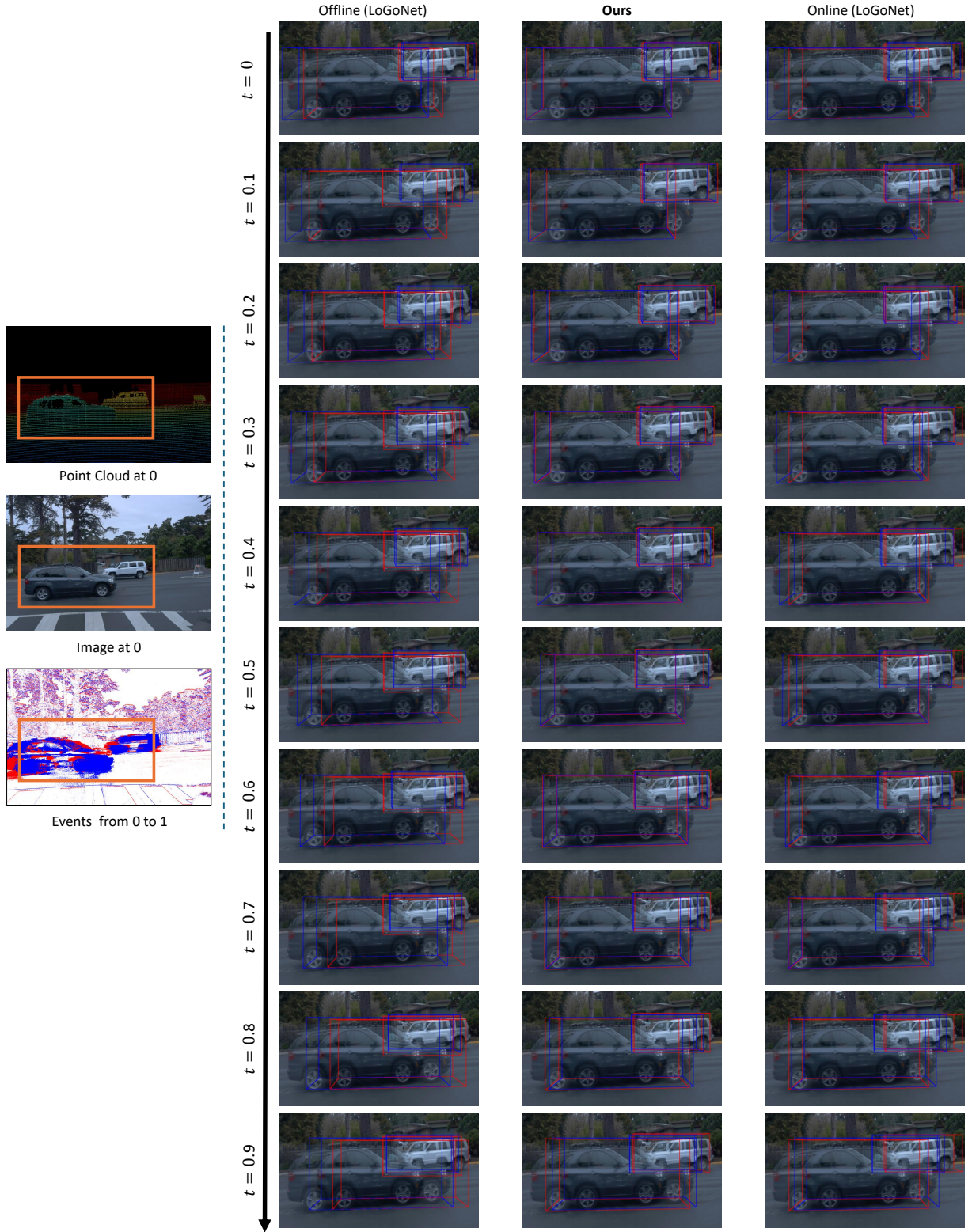


Figure I. Qualitative comparisons of our method with other offline and online evaluations on the Ev-Waymo dataset.  $t = 0$  represents the active time, while  $t = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$  denote the blind times. The **blue** box represents the ground truth, while the **red** box shows the prediction results of each method. For easier understanding, images at active timestamps 0 and 1 are overlaid.



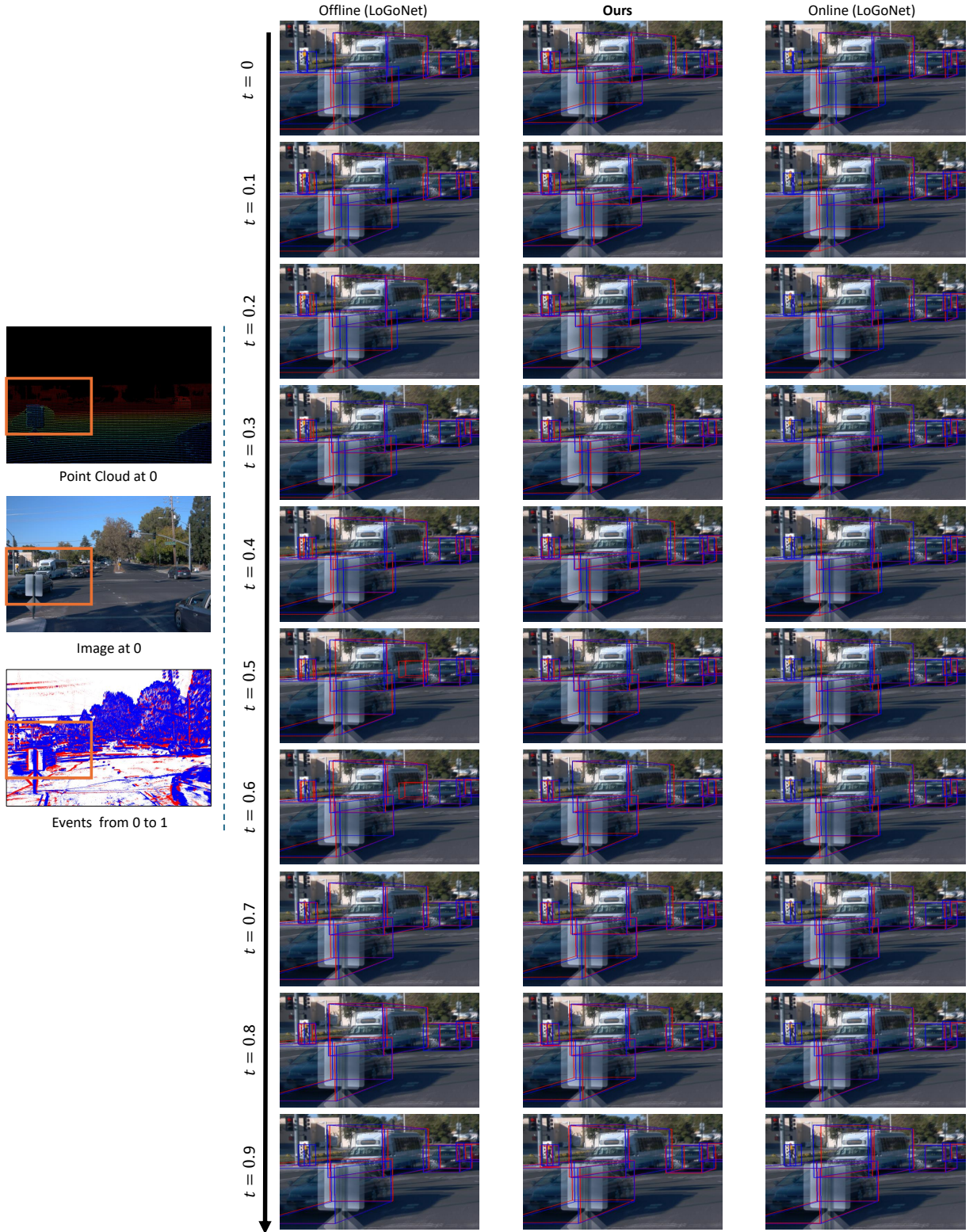


Figure J. Qualitative comparisons of our method with other offline and online evaluations on the Ev-Waymo dataset.  $t = 0$  represents the active time, while  $t = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$  denote the blind times. The blue box represents the ground truth, while the red box shows the prediction results of each method. For easier understanding, images at active timestamps 0 and 1 are overlaid.