3DPPE: 3D Point Positional Encoding for Transformer-based Multi-Camera 3D Object Detection (Supplementary Materials)

Changyong Shu[1*] Jiajun Deng[2*] Fisher Yu[3] Yifan Liu[4†]
[1]Houmo AI, [2]University of Sydney, [3]ETH Zürich, [4]University of Adelaide,
changyong.shu89@gmail.com, jiajun.deng@sydney.edu.au
fisheryu@ethz.ch, yifan.liu04@adelaide.edu.au

1. Dataset and Metric

Dataset. we conduct experiments on nuScenes dataset, a comprehensive autonomous driving dataset that encompasses a variety of perception tasks, such as detection, tracking, and LiDAR segmentation. The nuScenes dataset comprises 1,000 distinct driving scenes, divided into three distinct subsets for training (700), validation (150), and testing (150) purposes, respectively. Each of these driving scenes includes 20 seconds of perceptual data that are annotated with a keyframe at a frequency of 2 Hz. The data collection vehicle employed in this study is equipped with one LiDAR, five radars, and six cameras that capture a surround view of the vehicle's environment.

Metrics. We follow the official protocol to report the nuScense Score (NDS), mean Average Precision (mAP), along with five true positive metrics including mean Average Translation Error (mATE), mean Average Scale Error (mASE), mean Average Orientation Error (mAOE), mean Average Velocity Error (mAVE) and mean Average Attribute Error (mAAE).

2. Experimental Details

For comprehensive comparison, we have conducted experiments with ResNet-50, ResNet-101 and VoVNet-99 as the backbone networks in our experiments. Following the setting of the PETR series, we use P4 feature by default. Specifically, P4 feature is obtained by upsampling the C5 feature (output of the 5th stage) and fused with the C4 feature (output of the 4th stage). The P4 feature with 1/16 input resolution or the C5 feature with 1/8 input resolution is used as the 2D feature. The monocular depth ranges from 0 to 61m. The region of 3D perception space is set to $[-61.2m, 61.2m]$ for X and Y dimension and $[-10m, 10m]$ for Z dimension. The 3D coordinates in point cloud are normalized to [0,1]. As for the hyper-parameters in each loss component, we set $\lambda_{sl1}/\lambda_{DFL}/\lambda_{cls}/\lambda_{reg}$ to be $0.25/0.25/2.0/1.0/$ respectively, and λ_{cls} and λ_{reg} is the loss weight for classification and regression follow PETR series. AdamW [2] optimizer with a weight decay of 0.01 is used for training model, and the learning rate is initialized as 2.0e-4 and decayed with cosine annealing scheme [1]. Unless otherwise stated, all experiments with a batch size of 8 are trained for 24 epochs on 4 Tesla V100 GPUs. Test augmentation methods are not used during the inference.

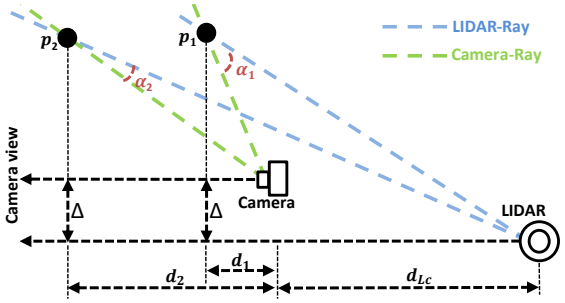

Figure 1. A mathematic model for the included angle α between camera-ray and LiDAR-ray in a surround-view system. d_{Lc} is the distance along the camera ray from LIDAR to the camera, Δ is the distance perpendicular to camera ray from LIDAR to the camera, Apparently, α_2 will be smaller than α_1 (approaching 0), and meanwhile d_2 becomes larger than d_1.

3. Analysis of 3D Positional Encoding

In this section, we first decoupled the positional encoding of PETR into three factors, *i.e.*, depth values number N_{D}, discretization method M_{D} and depth range R_{D}. Then, the influence of each factor is explored through ablation studies in Sec. 3.1. According to the experimental results, we summarize a feasible physical model to explain the meaning of the positional encoding in PETR. In Sec. 3.2, a new assumption of using LiDAR-ray as positional encod-

*These authors contributed equally to this work.
†Corresponding authors.

Table 1. Quantitative comparison of different depth values number $N_{\mathtt{D}}$, discretization methods $M_{\mathtt{D}}$ and depth range $R_{\mathtt{D}}$. SID and UD denote spacing-increasing discretization and uniform discretization respectively. The invariable performances of top 2-4 rows with diverse $M_{\mathtt{D}}$ indicate that $M_{\mathtt{D}}$ is irrelevant. Thus we fix $M_{\mathtt{D}}$ as simplest UD but change $R_{\mathtt{D}}$ as in row 5 and 6, consistent performances demonstrate that $R_{\mathtt{D}}$ is also incoherence. Finally, we fix $M_{\mathtt{D}}$ and $R_{\mathtt{D}}$ but reduce the $N_{\mathtt{D}}$ to 32 and 2 respectively in last 2 rows, the immune performances declare that $N_{\mathtt{D}}$ also does not largely affect the results.

$N^{\mathtt{D}}$	$M_{\mathtt{D}}$	$R_{\mathtt{D}}$	NDS↑	mAP↑	mATE↓
64	LID	[1,61]	0.338	0.275	0.853
64	SID	[1,61]	0.343	0.275	0.856
64	UD	[1,61]	0.337	0.273	0.847
64	UD	[1,31]	0.340	0.274	0.855
64	UD	[31,61]	0.336	0.272	0.849
32	UD	[1,61]	0.342	0.274	0.857
2	-	[1,61]	0.345	0.276	0.845

ing is proposed. Extensive experiments provide evidence for the new assumption. All experiments in this section are performed without CBGS and the backbone is set to ResNet-50, C5 feature is selected as 2D image feature, the train image size is set to 704×256 defaultly.

3.1. 3D Camera-Ray PE

PETR series divide the depth range $R_{\mathtt{D}}$ [1m, 61m] into $N^{\mathtt{D}} = 64$ depth bins following linear-increasing discretization (LID). Therefore, one pixel corresponds to 64 separated 3d points lying on the corresponding camera ray. The 3D coordinates of these points are fed together into a 3D positional encoding encoder to generate the PE. In order to clarify what location information is encoded in 3D PE, we further explore the effectiveness of different depth bin numbers $N_{\mathtt{D}}$, discretization methods $M_{\mathtt{D}}$ and depth range $R_{\mathtt{D}}$ as shown in Table 1. Surprisingly, the results turn out to be almost invariable under different settings, where the fluctuations of NDS, mAP and $mATE$ are smaller than 0.8%, 0.4% and 1.0% respectively. It gives a intuitive hint that the performance remains virtually unchanged through the separated 3D points sliding on the camera-ray. If the sampled points can represent the direction of the camera-ray, it already provides equivalent information to the PETR's 3D encoding. Thus, we propose a 3D camera-ray assumption that we could encode the 2D feature by two points on the camera-ray penetrating this pixel.

3.2. LiDAR-Ray PE Assumption

In this section, we further reduce $N_{\mathtt{D}}$ to 1 with fixed depth d such as 0.2m, 1m, 15m, 30m and 60m respectively. As listed in Table 2, smaller d leads to inferior performance (row 1 and 2 in Table 2). It indicates that the scheme of current PE is no longer camera-ray, while on-par result (compared to results listed in Table 1) is achieved when d is larger than 15m. The phenomenon above enlightens us that the PE in PETR with $N^{\mathtt{D}} = 1$ represents a LiDAR-ray. Determination of a ray direction requires two points. As the fixed LiDAR's location provides the start point of ray, thus we can determine LIDAR-ray direction with one point.

As shown in Figure 1 (b), we calculate the discrepancy (`Dis`) between camera-ray and LiDAR-ray with the cosine of their included angle:

$$\begin{aligned} \mathtt{Dis} &= 1 - \cos(\alpha) \\ &= 1 - \cos(\alpha_c - \arctan(\frac{\tan\alpha_c + \frac{\Delta}{d}}{1 + \frac{d_{L_c}}{d}})) \\ &\approx 0.0 \quad \mathtt{when} \;\; d \gg d_{L_c} \mathtt{and} \; d \gg \Delta, \end{aligned} \tag{1}$$

where α_c is the azimuth angle of camera-ray, α is the included angle between camera-ray and LiDAR-ray, d_{L_c} is the distance between camera and LiDAR along camera view (which ranges from 0.5m to 1.2m in universal sensor configuration), Δ is the distance between camera and LiDAR vertical to camera view (which ranges from 0.0m to 1.0m in universal sensor configuration). When $d \gg d_{L_c}$ and $d \gg \Delta$, the discrepancy between camera-ray and LiDAR-ray is almost eliminated. This further demonstrates that when d is larger, the LiDAR-ray will have a similar direction as the camera-ray, thus, the experiment results will be on par with the original PETR positional encoding even with one point.

Table 2. Quantitative comparison of different fixed depth d when depth point number $N_{\mathtt{D}}$ is 1.

$N^{\mathtt{D}}$	$M_{\mathtt{D}}$	d	NDS↑	mAP↑	mATE↓
1	-	0.2	0.304	0.229	0.948
1	-	1	0.323	0.251	0.907
1	-	15	0.333	0.275	0.842
1	-	30	0.340	0.271	0.844
1	-	60	0.338	0.275	0.849

4. More Ablation Study

Experiment settings in this section are following setting in the above analysis of 3D positional encoding.

4.1. More Studies on PE Similarity

To demonstrate the 3D point PE is capable of more precise locating capability, we randomly select position on background and object (appeared on the cross-view) respectively from the front view, the similarity between the position and all pixels of the surround views is computed. Figure 2 is the complete version of the Figure 6 in main paper.

In terem of point selected on cross-view object, as illustrated in Figure 3, the 3D point PE can find the related object from other view without redundant focus on the background. In term of the PE selected at the road, the 3D point PE tend to focus on the closer region round the selected position compared to the 3D camera-ray PE, as vividly shown in Figure 4.

4.2. Main improvement comes from reasonable 3D point PE

We conduct well designed experiments, as list in Table 3, to demonstrate the main contribution comes from 3D point PE but not supervision of sparse depth maps. We would like to clarify it through two comparisons: (a) Without depth supervision, replacing camera-ray PE with 3D point PE achieves 0.6% NDS improvement (34.3% in 6-th row V.S. 33.7% in second row). (b) We show that depth supervision can also be applied to camera-ray PE. Under depth supervision, Our model with 3D point PE (last row) consistently outperforms that with camera-ray PE (5-th row) by 0.9% NDS and 0.8% mAP. These results together demonstrate the main technical improvement is the more reasonable 3D point PE.

Table 3. Comparison of different 3D position-aware feature w/o sparse depth maps. Camera-ray and feature-guided (extended version of camera-ray) are proposed in PETR and PETRv2 respectively. 3D point is our proposed method.

3D Position-aware		NDS↑	mAP↑	mATE↓
Camera-Ray	w/o	0.337	0.274	0.852
	w	-	-	-
Feature-guided	w/o	0.352	0.283	0.843
	w	0.359	0.291	0.826
3D-Point	w/o	0.343	0.266	0.832
	w	0.368	0.299	0.807

4.3. Accuracy of depth approximation

We follow the common practice to evaluate the depth accuracy in terms of SILog, AbsRel, SqRel, and RMSE. As shown in Table 4, our hybrid depth net effectively reduces the depth error, thus improving 3D detection performance. This table will be merged to Table 4 of the manuscript in the revision.

Table 4. Evaluation of depth prediction on the nuScenes val set.

$L_{\texttt{smooth-L1}}$	$L_{\texttt{dfl}}$	SILog↓	AbsRel↓	SqRel↓	RMSE↓
		63.71	2.54	58.81	20.09
✓		18.78	0.09	0.73	5.35
✓	✓	17.65	0.08	0.67	4.92

References

[1] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 1

[2] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1

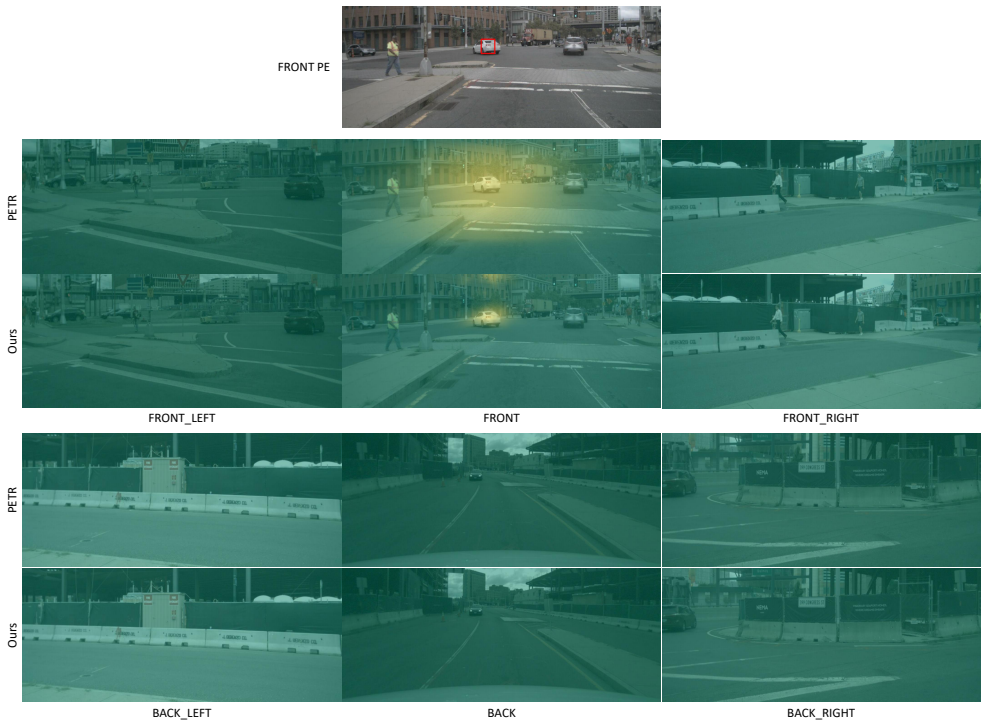

Figure 2. Representative similarity comparison of 3D camera-ray PE in PETR and ours 3D point PE, best viewed in color.

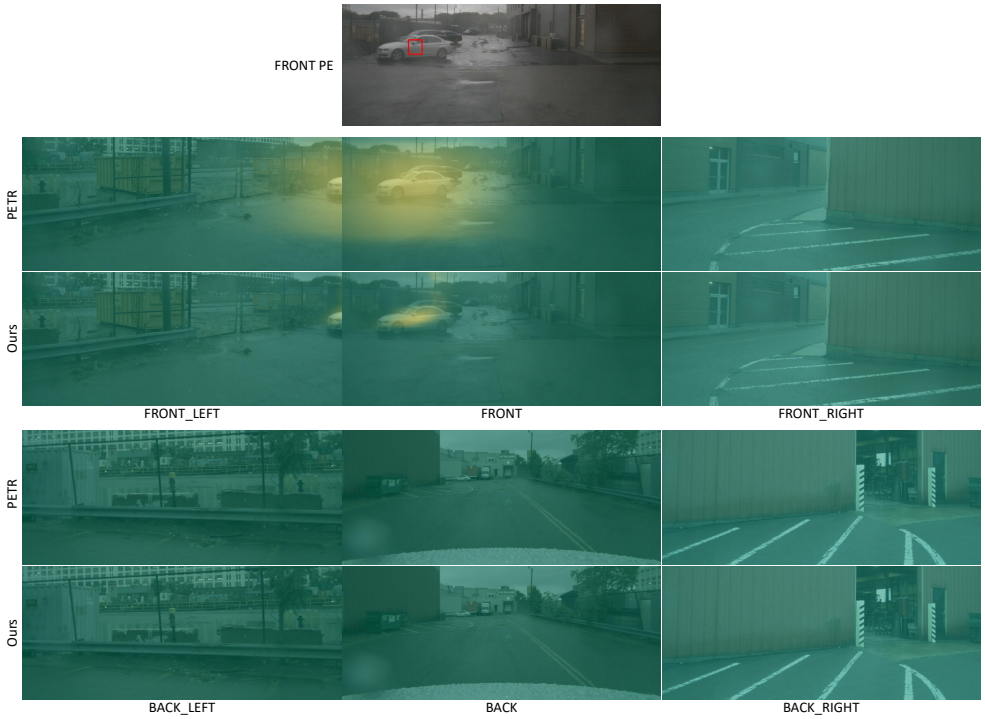

Figure 3. Representative similarity comparison of 3D camera-ray PE in PETR and ours 3D point PE, best viewed in color. The position is selected on the car object appeared in cross-view.

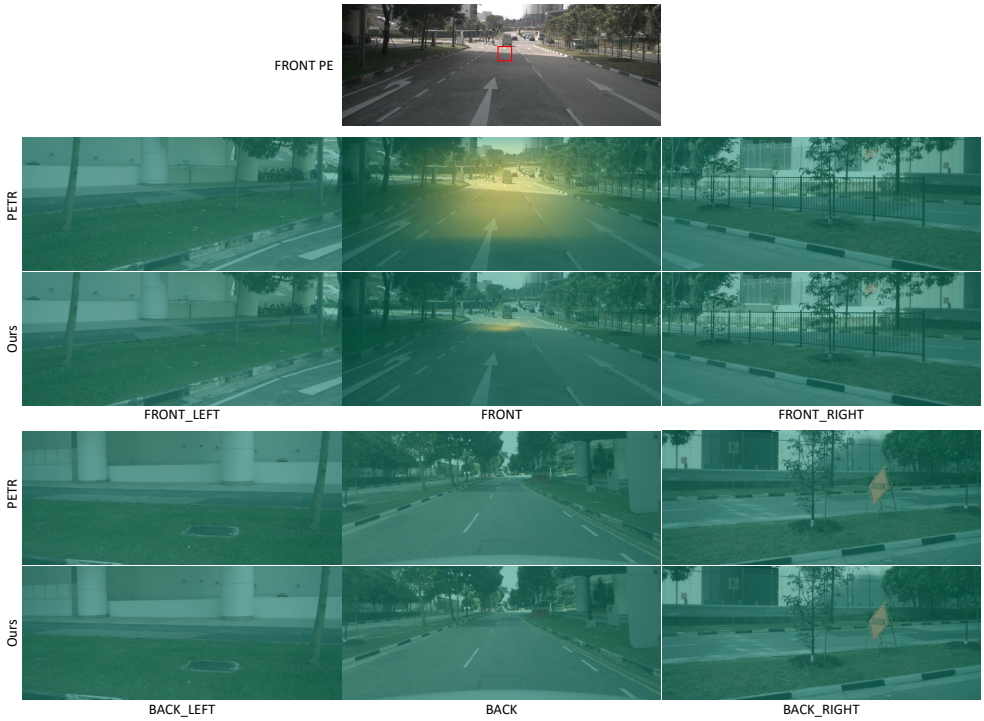

Figure 4. Representative similarity comparison of 3D camera-ray PE in PETR and ours 3D point PE, best viewed in color. The position is selected on the background.

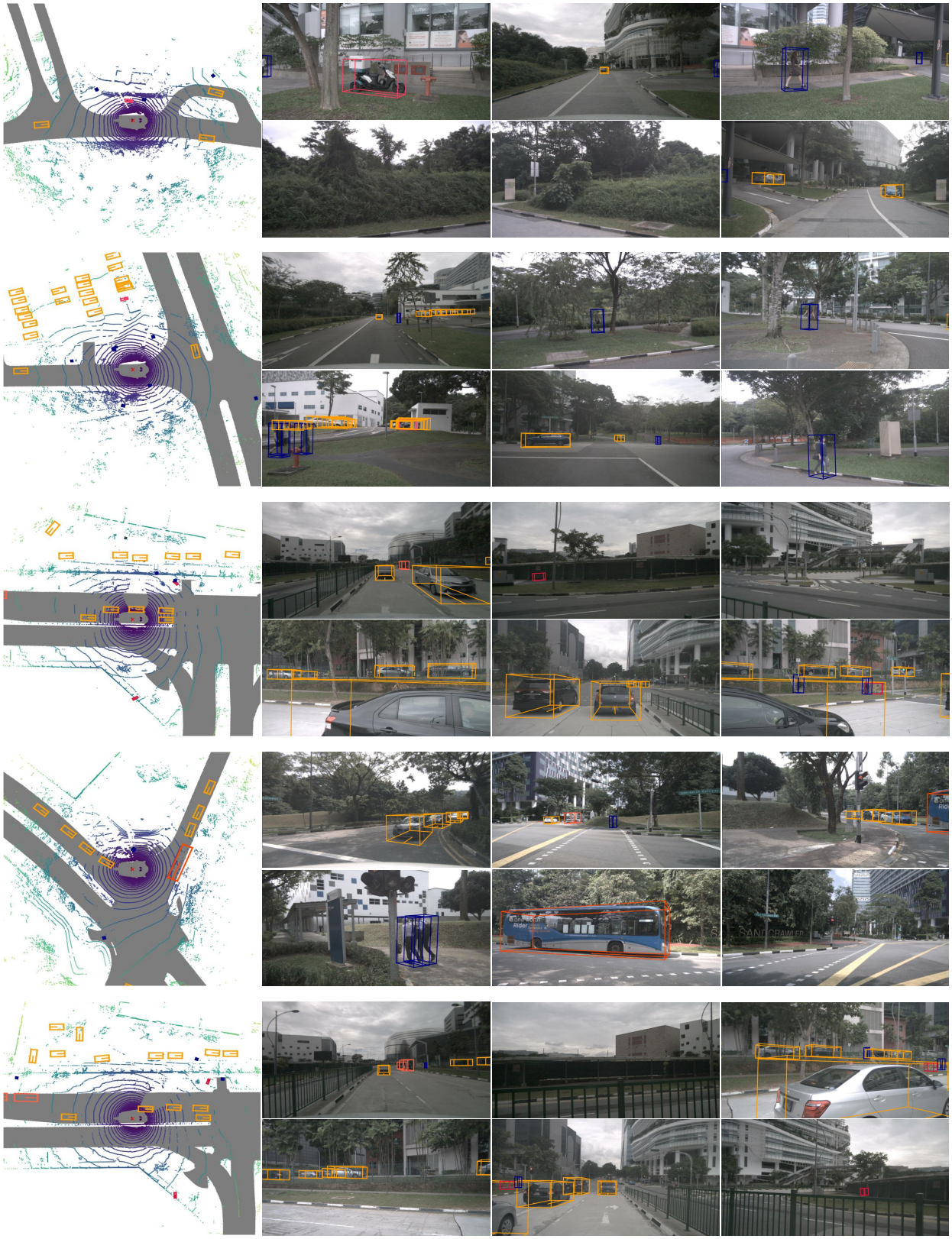

Figure 5. Qualitative results for our method, best viewed in zoom and color.