Table 8: Comparisons to FCOS3D without global NMS.

Method	NDS ↑	mAP ↑	mATE ↓	mASE ↓	mAOE ↓	mAVE ↓	mAAE ↓	global NMS	overlap region only
FCOS3D ‡	0.273	0.133	0.890	0.274	0.593	1.093	0.176	-	✓
FCOS3D	0.317	0.213	0.841	0.276	0.604	1.122	0.173	✓	✓
DETR3D (Ours)	0.356	0.231	0.825	0.280	0.400	0.863	0.223	-	✓
FCOS3D ‡	0.336	0.234	0.830	0.268	0.558	1.361	0.153	-	-
FCOS3D ‡	0.373	0.299	0.785	0.268	0.557	1.396	0.154	✓	-
DETR3D (Ours)	0.374	0.303	0.860	0.278	0.437	0.967	0.235	-	-

Table 9: Time complexity.

Models	FCOS3D	FCOS3D (w/o global NMS)	FCOS3D (w/o global NMS, w/o per-image NMS)	MVOD (ours)
FPS	1.1	1.2	3.3	3.1

Supplementary Material

Comparison to FCOS3D without global NMS.

We also compare our MVOD method to FCOS3D without global NMS. We test in both settings: benchmarking using all data and using ground-truth bounding boxes only in the overlap regions. As shown in Table 8, our model outperforms FCOS3D without global NMS. This suggests that FCOS3D is very reliant on global NMS, with a significant drop in performance when it is disabled, while our method completely eliminates the need for it. In addition, we also remove the per-image NMS in FCOS3D and the model fails. So we ignore the results in this table.

Time complexity.

We compare the time complexity of FCOS3D with and without global NMS to that of our proposed MVOD model. Table 9 shows that MVOD is more efficient at inference time thanks to its NMS-free design. We also provide the performance when the NMS is not used in the FCOS3D. The per-image NMS contributes to a large portion of the overhead, which is not required by our model. We also note that further improvements in efficiency can be achieved by implementing more GPU-friendly feature sampling in our model. These frame rates were measured on a single Nvidia RTX 3090.

More visualization.

We also provide qualitative results in Figure 3 to facilitate an intuitive understanding of model performance. We project the predicted bounding boxes into 6 cameras as well as a BEV perspective. In general, our method generates reasonable results and even detects relatively small objects. However, our method still exhibits substantial translation error (in line with results in Table 4.2): Although our model avoids explicit depth prediction, depth estimation is still a core challenging in this problem.

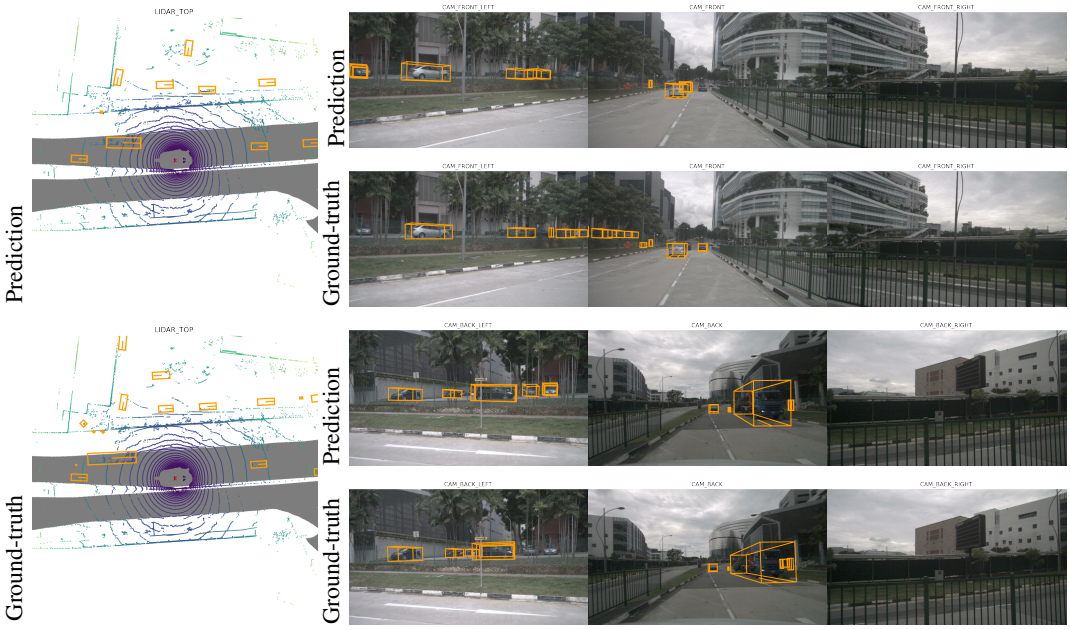

Figure 3: We visualize DETR3D predictions in both BEV and image views. Our model is capable of detecting rather small objects and even objects that were not annotated as ground-truth (cars in `CAM_BACK_LEFT`). Some failure cases include the far ahead car in `CAM_FRONT`, that was not detected.