Supplementary Material for RobusTor3D: Robust Multimodal 3D Object Detector for Autonomous Driving by Vision-Language Knowledge Blending

Anonymous submission

In this document, we provide additional content to supplement the main manuscript. Section A offers several examples of text description generation. Section B provides more implementation details. Section C presents additional qualitative and quantitative results.

A Dataset

Examples of the generated text descriptions

Based on the KITTI dataset(Geiger, Lenz, and Urtasun 2012) and the CADC dataset(Pitropov et al. 2021), we utilize GPT-2 to generate corresponding text descriptions from images. Figure 1 provides several examples.

KITTI dataset

In the KITTI dataset(Geiger, Lenz, and Urtasun 2012), there are three difficulty levels: Easy, Moderate (Mod.), and Hard, which are determined based on object size, visibility (occlusion), and truncation. The IoU threshold for the car class is set to 0.7, and 0.5 for the pedestrian and cyclist classes.

KITTI-C dataset

In the KITTI-C benchmark(Yu et al. 2023), all configurations are the same as the KITTI dataset.

CADC dataset

The CADC dataset(Pitropov et al. 2021) comprises 6901 LiDARs and images, including three classes(car, pedestrian, Truck). The dataset has been converted to KITTI format, 6315 for training and 586 for testing. We further divide the training set into train split set (referred to as the train set) comprising 5052 samples and validation split set (referred to as the val set) comprising 1263 samples with a 4:1 ratio. The IoU threshold for the car class is set to 0.7, and 0.5 for the pedestrian and truck classes.

B Implementation Details

On all datasets, we consider point clouds within the range of [-40m, 40m] × [0m, 70.4m] × [-3m, 1m] meters along the Y, X, and Z axes, respectively. For our RobusTor3D on the KITTI, KITTI-C, and CADC datasets, we consider point clouds within the range of [-40m, 40m] × [0m, 70.4m] × [-3m, 1m] meters along the Y, X and Z axis, respectively. For voxelization, the voxel size is set to (vH = 0.05m, vW =0.05m, vD = 0.1m) with anchor sizes of [3.9, 1.6, 1.56] for car, while [0.8, 0.6, 1.73] for pedestrian and [1.76, 0.6, 1.73] for cyclist/truck. Taking images with a resolution of 1280×384 as input.

C Experiments

Composite robustness experiments

All five challenges mentioned in the main manuscript exist, but only the experimental results under a single challenge are discussed. In the real world, these challenges may coexist, such as long tail + adverse weather, long tail + sensor spatial misalignment/motion blur, and modality loss + cross-domain shifts. Therefore, in this subsection, we discuss and analyze the robustness of our RobusTor3D and other methods under the above-mentioned composite challenges. The experimental results show that RobusTor3D has strong composite robustness.

Long tail + Adverse weather on KITTI-C. Table 1 shows the comparison results of 3D mAP across all classes with other methods on the KITTI-C val set under adverse weather conditions. We can see that our RobusTor3D has an absolute advantage in pedestrian and overall performance. In particular, the pedestrian category outperforms other methods by 7.59%-8.46%, 6.55%-12.06%, 5.33%-10.96%, and 3.11%-7.09% in rain, snow, fog, and strong sunlight, respectively. Furthermore, the performance in the car and cyclist categories is comparable. Although not optimal in all metrics, compared to other methods, RobusTor3D significantly alleviates the long-tail problem and improves overall performance, even in adverse weather conditions.

Long tail + Sensor spatial misalignment/Motion blur on KITTI-C. Table 2 presents a comparison of 3D mAP across all classes with other methods on the KITTI-C val set, under sensor spatial misalignment or motion blur. It can be observed that our RobusTor3D achieves the best performance on all metrics except for Cyclist, which is slightly lower than SSLFusion by 0.82% under motion blur. In particular, the pedestrian classification outperforms other methods by 0.33%-8.43% and 0.08%-8.40% under spatial alignment and motion blur, respectively. This demonstrates the superiority of RobusTor3D for long-tail categories.

Modality loss + Cross-domain shifts (KITTI→CADC). Table 3 shows comparison results of 3D AP across all diffi-

Method	Car(IoU=0.7)				Pedestrian (IoU=0.5)				Cyclist(IoU=0.5)				mAP_{All}			
	Rain	Snow	Fog	SSun	Rain	Snow	Fog	SSun	Rain	Snow	Fog	SSun	Rain	Snow	Fog	SSun
FocalsConv-F	86.01	85.85	84.47	85.84	57.54	58.69	61.89	60.09	74.13	**76.87**	75.10	79.06	72.56	73.81	73.82	75.00
Graph-VoI	86.60	**87.34**	**86.70**	86.73	57.27	57.33	56.26	56.83	74.11	74.12	71.65	71.80	72.66	72.93	71.54	71.79
SSLFusion	85.21	85.61	84.51	87.34	56.67	53.78	61.67	61.71	**76.61**	75.41	70.82	**81.01**	72.83	71.60	72.34	76.69
RobusTor3D (Ours)	**86.97**	86.96	86.53	**88.04**	**65.13**	**65.84**	**67.22**	**64.82**	75.00	75.25	**76.44**	78.17	**75.71**	**76.02**	**76.73**	**77.02**

Table 1: Comparison results of 3D mAP across all classes with other methods on KITTI-C val set under adverse weather conditions. SSun represents strong sunlight.

Method	Car(IoU=0.7)		Pedestrian (IoU=0.5)		Cyclist(IoU=0.5)		mAP_{All}	
	Noise	Motion blur	Noise	Motion blur	Noise	Motion blur	Noise	Motion blur
FocalsConv-F	65.81	86.37	48.54	60.16	54.22	74.09	56.18	73.55
Graph-VoI	54.91	86.61	40.44	56.92	46.30	73.74	47.21	72.43
SSLFusion	57.30	85.20	41.26	65.24	47.48	**75.60**	48.68	75.35
RobusTor3D (Ours)	**67.70**	**86.74**	**48.87**	**65.32**	**60.15**	74.78	**59.14**	**75.63**

Table 2: Comparison results of 3D mAP across all classes with other methods on KITTI-C val set under sensor spatial misalignment and motion blur. Noise represents sensor spatial alignment noise.

culty levels of the Car class with other methods under cross-domain(KITTI→CADC) and modality drop. It can be seen that when (KITTI→CADC), regardless of whether the image or point cloud (PC) is dropped, Graph-VoI and SSLFusion still completely fail. Compared with FocalsConv-F, our RobusTor3D achieves the best results in all indicators, with 3D mAP higher by 9.94% and 3.45% when the image or point cloud (PC) is dropped, respectively.

Comparison on KITTI val set

Table 4 compares RobusTor3D with several state-of-the-art multimodal 3D detection methods on the KITTI validation set. Our method achieves the best overall performance with a 3D mAP of 78.20%, surpassing the SOTA methods RAFDet(Zheng et al. 2025), PVAFN(Li et al. 2025), FullPointTrans(He et al. 2024), and SSLFusion(Ding et al. 2025) by 1.99%, 5.47%, 2.57%, and 1.9%. In addition, our method performs 1.53% lower than RAFDet in the cyclist category, but RobusTor3D outperforms RAFDet by 3.15% in the pedestrian category. RobusTor3D achieves the best performance in all three categories compared to other methods. These improvements especially reflect the effectiveness of our visual language-guided semantic enhancement and scene-level multimodal consistency learning, which helps to enhance the generalization ability of our method under different object types and scene conditions.

Ablation experiment

Ablation study of parameters in Offline Text Supervision Generation. In this section, we study the effect of different text generation methods in Offline Text Supervision Generation (OTSG) and parameter N_t in CLIP Text Encoder, as shown in Figure 2, where N_t represents the number of encoded words. Fig presents comparison results using BLIP, BLIP2, and GPT-2 as image-to-text generation methods and different N_t. In the original CLIP text encoder, the N_t=77, but we observe that there are invalid values 0 in the

Figure 1: Examples of generated text descriptions on KITTI and CADC datasets.

encoded text due to the short text description, which will decrease the performance extremely. Therefore, we reduce the value of N_t, and lots of experiments have shown that when using BLIP, BLIP2, and GPT-2 respectively, N_t=25, N_t=15, and N_t=15 achieve the best results.

Ablation study of how to integrate text features. Unlike the projected relationships between point clouds and images, there is no direct correspondence between point clouds and texts. Therefore, we need to consider how to integrate text features into the detection model. We consider three methods: (a1) directly convert the dimension of the text feature to the same dimension as the point cloud through linear transformation, and then use the projection matrix to project the point cloud to the image plane to obtain voxel-level image features, and then fuse the three modalities; (a2) the method employed in the paper, first convert the text feature to the image feature space and perform fusion, and then obtain

Method	Image drop + Cross-domain (KITTI→CADC)				PC dorp+ Cross-domain (KITTI→CADC)			
	Easy	Mod.	Hard	mAP	Easy	Mod.	Hard	mAP
Graph-VoI	0	0	0	0	0	0	0	0
SSLFusion	0	0	0	0	0	0	0	0
FocalsConv-F	42.40	45.30	48.34	45.35	0.21	0.17	0.21	0.20
RobusTor3D (Ours)	**53.42**	**54.44**	**58.02**	**55.29**	**3.86**	**3.33**	**3.77**	**3.65**

Table 3: Comparison results of 3D AP cross all difficulty levels of the Car class with other methods under cross-domain(KITTI→CADC) and modality drop. PC represents point cloud.

Methods	Car	Ped.	Cyclist	mAP_{All}
MVXNet*	80.04	54.25	59.91	64.73
AutoAlign	80.35	63.43	74.25	72.67
FocalsConv-F*	86.32	63.16	73.62	74.36
Graph-VoI*	88.13	57.78	72.51	72.80
RAFDet	83.93	63.76	**80.93**	76.21
PVAFN	86.22	54.98	76.99	72.73
FullPointTrans	86.50	66.22	78.16	75.63
SSLFusion	88.40	60.87	79.63	76.30
SSLFusion*	87.14	61.74	79.43	76.10
RobusTor3D	**88.28**	**66.91**	79.40	**78.20**

Table 4: Comparison results with other methods on KITTI val set.

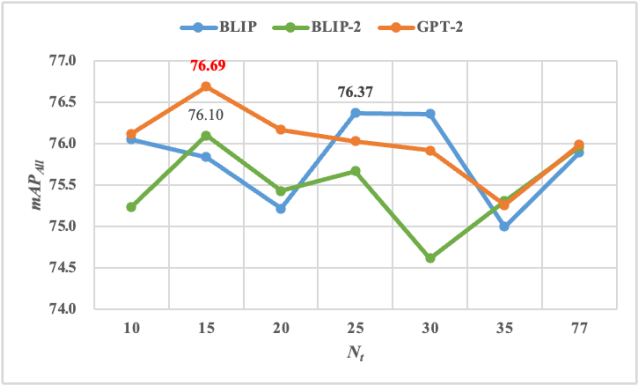

Figure 2: Ablation study of parameters in Offline Text Generation. N_t represents the number of encoded words.

voxel-level features and voxel features based on the projection transformation; (a3) first convert the image feature to the text feature space and perform fusion, and then convert the dimension of the fused feature to the same dimension as the point cloud through linear transformation before fusing. Table 5 shows the experimental comparison of the three methods. Although (a2) is not optimal in the cyclist category, it achieves the best performance in other categories and overall. In particular, the performance of the pedestrian category is 2.17% and 1.71% higher than (a1) and (a3), respectively. Therefore, (a2) is the most effective.

Ablation study of fusion methods. Our network architecture involves two feature fusions: (1) image features $\mathcal{F}_I$ and text features $\mathcal{F}_T$; (2) projected features $\mathcal{F}_{sv\_clip}$ and point cloud features $\mathcal{F}_{SV}$. Therefore, we explored different fusion methods, as shown in Table 6, where Concat, SA, Multiply, and Gate denote concatenation, self-attention, Hadamard product, and adaptive gated fusion, respectively. Although the add+Gate fusion method achieves optimal performance in the car, cyclist, and overall mAP_{All}, it performs poorly in the pedestrian category. Therefore, considering the long-tail problem, we chose a simple and direct addition that is more balanced between each category and overall performance, without introducing additional computation and parameters. This is also because we designed a scene-level consistency comparison learning strategy to ensure that different modalities can ensure global semantic consistency.

Methods	Car	Ped.	Cyclist	mAP_{All}
(a1)	87.71	63.57	**80.87**	77.38
(a2)	**87.76**	**65.74**	79.41	**77.64**
(a3)	86.50	64.03	79.74	76.76

Table 5: Ablation study of how to integrate text features.

$\mathcal{F}_I$ and $\mathcal{F}_T$	$\mathcal{F}_{sv\_clip}$ and $\mathcal{F}_{SV}$	Car	Ped.	Cyc.	mAP_{All}
Concat	Add	86.28	64.16	79.40	76.95
SA	Add	87.59	65.55	78.24	77.13
Multiply	Add	87.65	65.68	79.31	77.55
Add	Multiply	87.75	63.81	77.96	76.50
Add	Gate	**88.54**	64.88	**79.71**	**77.71**
Multiply	Multiply	86.67	61.63	78.68	75.66
Add	Add	87.76	**65.74**	79.41	77.64

Table 6: Ablation study of fusion methods. Concat, SA, Multiply, and Gate denote concatenation, self-attention, Hadamard product, and adaptive gated fusion, respectively.

Ablation study of alignment method of $\mathcal{F}_{sv\_clip}$ and $\mathcal{F}_{SV}$. To make the prior knowledge F_{sv_clip} extracted by Vision-Language Knowledge Extraction (V-LKE) better enhance the point cloud features F_{SV}, we explored different alignment methods from different levels to make the multimodal features consistent, which is helpful for subsequent fusion. Table 7 compares the alignment methods at three levels: (1) Semantic level, Kullback-Leibler Divergence (KL) loss is used to make the distribution probability of heterogeneous modalities consistent. The smaller the KL loss, the closer their distribution is. (2) Instance-level, according to the set anchor box, the predicted 3D instance is obtained through the point cloud branch, and the 3D image-level instance is obtained according to the projection matrix, and the contrast loss of the 3D-2D instance is calculated. (3)

Scene level, we explored different loss calculation methods to limit scene-level alignment, including L1 loss, L2 loss, SmoothL1 Loss, Relation Loss (Park et al. 2019), and contrastive loss. Among them, the contrast loss conducted three experiments, namely: calculating the similarity between $\mathcal{F}_{sv\_clip}$ and $\mathcal{F}_{SV}$, the similarity between $\mathcal{F}_{SV}$ and $\mathcal{F}_{sv\_clip}$, and the similarity between the both.

From Table 7, it can be observed that the semantic-level and scene-level alignment methods are better than the instance-level ones. The reason we analyze is that the instance-level method obtains 2D instances by projecting 3D instances onto the 2D image feature plane, rather than by predicting instances through a 2D detector. Therefore, instance-level alignment relies more on instances predicted by the point cloud branch. Compared to images, point clouds are sparse for distant targets and it is difficult to predict smaller targets, especially pedestrians and cyclists. Therefore, the performance on these two categories is extremely low. Moreover, although the semantic-level method achieves the best performance in both overall and scene-level comparative learning methods, with cyclists performing slightly better, we still take the long-tail problem into consideration, especially the pedestrian category, which is smaller in size. Therefore, we chose the scene-level comparative learning method with the highest performance for the pedestrian category to achieve consistent alignment.

Ablation study of parameters in loss function. The total loss function includes five types of losses,

$$\mathcal{L} = \alpha\mathcal{L}_{cls} + \lambda\mathcal{L}_{reg} + \gamma\mathcal{L}_{dir} + \beta\mathcal{L}_{CL\_clip} + \delta\mathcal{L}_{CL\_tip}. \quad (1)$$

The SigmoidFocalCrossEntropy loss, Weighted SmoothL1 loss, and Weighted CrossEntropy are adopted for classification loss $\mathcal{L}_{cls}$, regression loss $\mathcal{L}_{reg}$, and orientation loss $\mathcal{L}_{dir}$ (Chen et al. 2022), respectively. And $\alpha = \beta = \delta = 1.0, \lambda = 2.0, \gamma = 0.2$, where β, δ are set as the learnable parameters. Table 8 shows the ablation experiment of β, δ. It can be observed that when both β, δ=0.1, the performance in the car and cyclist categories is the best, but the performance in the pedestrian category is extremely low. In contrast, when the two parameters are set to learnable, they are not affected by the mutual influence of the two contrastive losses. The model automatically learns the optimal parameter configuration, which greatly improves the pedestrian and overall performance, alleviating the long tail problem.

Visualization

In this section, we provide visual comparisons of our RobusTor3D and other methods on the KITTI, KITTI-C, and CADC benchmarks, demonstrating that RobusTor3D achieves better detection results for real snow scenes, sensor spatial misalignment, long-tail categories (pedestrians and cyclists), severe occlusions, and distant objects.

Visualization comparison on the KITTI dataset. Figure 3 shows the visual comparisons of our RobusTor3D and other methods on the KITTI dataset, where green, light blue, and yellow 3D boxes represent Cars, Pedestrians, and Cyclists, respectively. The missed, false, and correct detections are labeled with red, purple, and orange dashed ellipses, respectively. It can be observed that FocalsConv-F* and Graph-VoI* FocalsConv-F* and Graph-VoI* fail to detect severely occluded and distant pedestrians and cyclists. SSLFusion* has some missed detections of pedestrians (tail class) and false detections of cars. Our RobusTor3D can accurately detect pedestrians and cyclists that are far away and severely obscured, further demonstrating the robustness of RobusTor3D.

Visualization comparison on the KITTI-C test set. Figure 4 shows the visual results comparisons of our RobusTor3D and other methods on the KITTI-C test set with sensor spatial alignment noise. The green, light blue, and yellow 3D boxes represent Cars, Pedestrians, and Cyclists. Red, purple, orange, and cyan-blue dashed ellipses denote missed, false, correct detections, and wrong orientation, respectively. Due to the presence of alignment noise, it can be observed that FocalsConv-F* and SSLFusion* have some wrong detections of pedestrians (tail class). FocalsConv-F* and Graph-VoI*, FocalsConv-F* and SSLFusion* fail to detect distant pedestrians and cars. SSLFusion* also has the wrong orientation when detecting car category. Our RobusTor3D can accurately detect pedestrians and cars that are far away and severely obscured, further demonstrating the alignment noise robustness of RobusTor3D.

Visualization comparison on the CADC test set. Figure 5 shows the visual results comparisons of our RobusTor3D and other methods on the CADC test set. The green, light blue, and yellow 3D boxes represent Cars, Pedestrians, and Trucks. Red, purple, and orange dashed ellipses denote missed, false, and correct detections, respectively. Due to the influence of snow, target detection is limited. We can see that FocalsConv-F* and Graph-VoI* often fail completely (fail to detect vehicles and pedestrians) or have difficulty detecting distant or small targets. SSLFusion* has some missed detections (pedestrians) and false detections (misdetecting cars as trucks). In contrast, our RobusTor3D can successfully detect all targets, but there are also some false detections (car in the bottom row). These results prove that RobusTor3D is also robust to real snow scenes, but there is still room for improvement to further improve accurate detection.

References

Chen, Y.; Li, Y.; Zhang, X.; Sun, J.; and Jia, J. 2022. Focal sparse convolutional networks for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5428–5437.

Ding, B.; Xie, J.; Nie, J.; and Cao, J. 2025. SSLFusion: Scale and Space Aligned Latent Fusion Model for Multimodal 3D Object Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2735–2743.

Geiger, A.; Lenz, P.; and Urtasun, R. 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, 3354–3361. IEEE.

He, Y.; Yu, H.; Yang, Z.; Liu, X.; Sun, W.; and Mian, A. 2024. Full point encoding for local feature aggregation in 3-d point clouds. *IEEE transactions on neural networks and learning systems*, 36(5): 8867–8881.

Alignment level	Alignment method	Car	Ped.	Cyclist	mAP_{All}
Semantic level	KL Loss	87.75	66.30	**80.35**	**78.14**
Instance level	Contrastive Loss (CL)	86.73	57.60	77.09	73.81
Scene level	L1 loss	88.56	66.34	79.37	78.09
	L2 loss	87.91	62.77	76.83	75.84
	SmoothL1 Loss	87.76	64.83	80.14	77.58
	Relation Loss	87.55	63.89	79.19	76.88
	CL (F_{sv_clip}, F_{SV})	87.79	64.94	79.39	77.37
	CL ($\mathcal{F}_{SV}$, $\mathcal{F}_{sv\_clip}$)+ CL($\mathcal{F}_{sv\_clip}$, F_{SV})	87.70	64.49	77.14	76.44
	CL ($\mathcal{F}_{SV}$, $\mathcal{F}_{sv\_clip}$)	**87.81**	**66.48**	79.61	**78.14**

Table 7: Ablation study of alignment methods between $\mathcal{F}_{sv\_clip}$ and $\mathcal{F}_{SV}$.

β	δ	Car	Ped.	Cyclist	mAP_{All}
1	1	87.72	63.40	78.95	76.69
0.5	0.5	87.44	65.32	79.21	77.32
0.2	0.2	87.75	66.64	77.99	77.46
0.1	0.1	**88.50**	64.41	**79.74**	77.55
Learnable		88.28	**66.91**	79.40	**78.20**

Table 8: Ablation study of parameters in loss function.

Li, Y.; Wen, J.; Gong, R.; Ren, B.; Li, W.; Cheng, C.; Liu, H.; and Sebe, N. 2025. Pvafn: Point-voxel attention fusion network with multi-pooling enhancing for 3d object detection. *Expert Systems with Applications*, 281: 127608.

Park, W.; Kim, D.; Lu, Y.; and Cho, M. 2019. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3967–3976.

Pitropov, M.; Garcia, D. E.; Rebello, J.; Smart, M.; Wang, C.; Czarnecki, K.; and Waslander, S. 2021. Canadian adverse driving conditions dataset. *The International Journal of Robotics Research*, 40(4-5): 681–690.

Yu, K.; Tao, T.; Xie, H.; Lin, Z.; Liang, T.; Wang, B.; Chen, P.; Hao, D.; Wang, Y.; and Liang, X. 2023. Benchmarking the robustness of lidar-camera fusion for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3188–3198.

Zheng, Z.; Huang, Z.; Zhao, J.; Lin, K.; Hu, H.; and Chen, D. 2025. RAFDet: Range view augmented fusion network for point-based 3D object detection. *IEEE Transactions on Multimedia*.

Figure 3: Comparison of visualization results on KITTI test set. The green, light blue, and yellow 3D boxes represent Cars, Pedestrians, and Cyclists. Red, purple, and orange dashed ellipses denote missed, false, and correct detections.

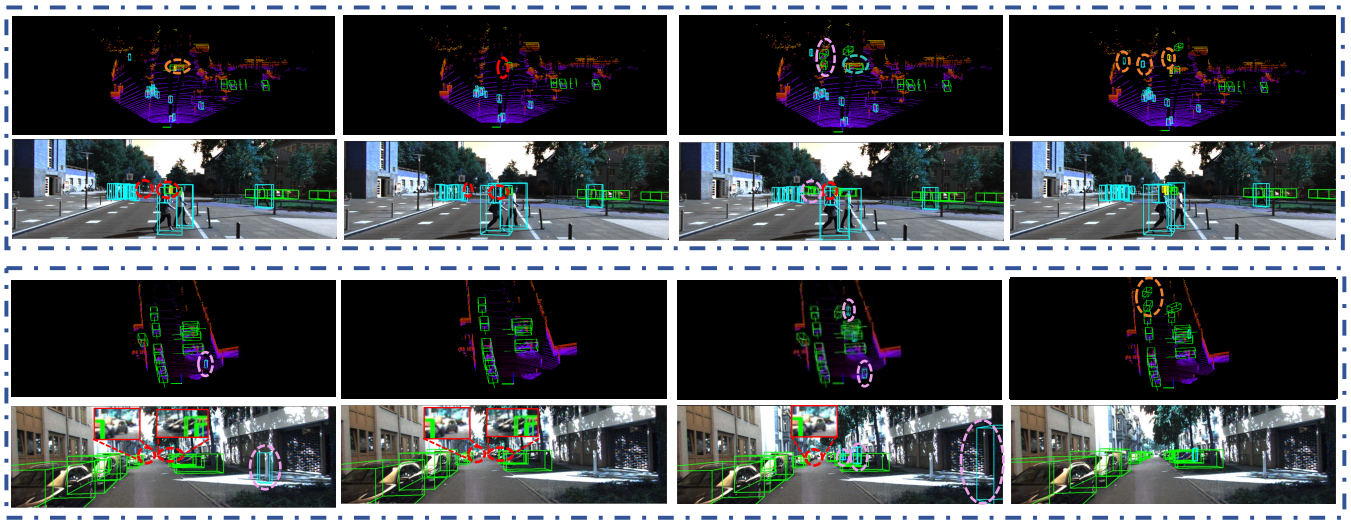

Figure 4: Comparison of visualization results on KITTI-C test set with sensor spatial alignment noise. The green, light blue, and yellow 3D boxes represent Cars, Pedestrians, and Cyclists. Red, purple, orange, and cyan-blue dashed ellipses denote missed, false, correct detections, and wrong orientation.

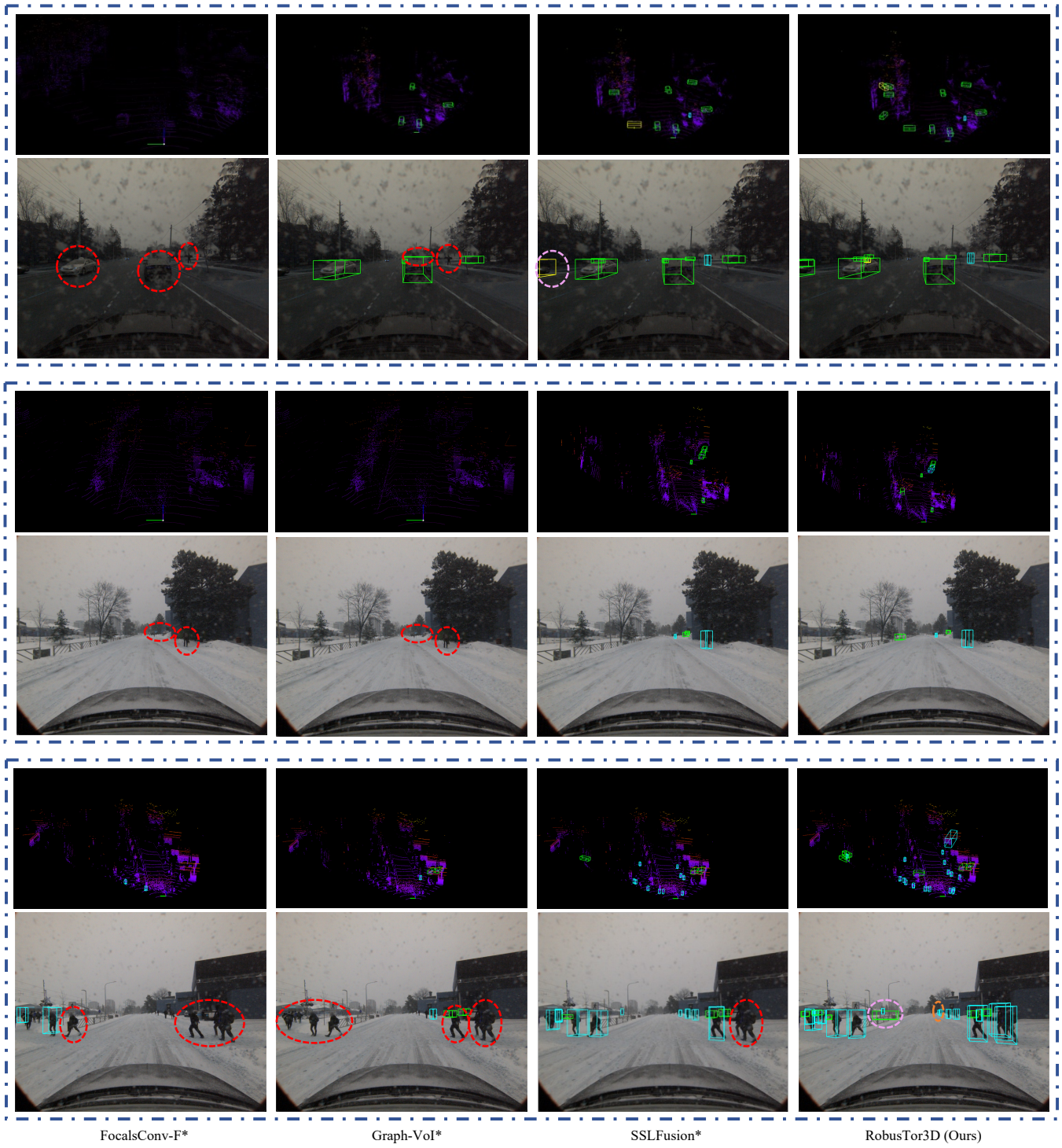

Figure 5: Comparison of visualization results on CADC test set. The green, light blue, and yellow 3D boxes represent Cars, Pedestrians, and Trucks. Red, purple, and orange dashed ellipses denote missed, false, and correct detections, respectively.