RobuAlign: Robust Alignment in Multi-Modal 3D Object Detection

Dongyi Fu
Harbin Institute of Technology
3129506344@qq.com

Yongchun Lin
Harbin Institute of Technology
emmlyc24@gmail.com

Huitong Yang
Guangdong University of Technology
huitongy0126@gmail.com

Haoang Li
HKUST (GZ)
haoangli@hkust-gz.edu.cn

Yadan Luo
The University of Queensland
y.luo@uq.edu.au

Xianjing Cheng
Harbin Institute of Technology
chengxianjing@hit.edu.cn

Yong Xu
Harbin Institute of Technology
laterfall2@yahoo.com.cn

Abstract

The failure of vehicle-mounted cameras and LiDAR sensors is inevitable in practice, resulting in incomplete data. To enhance the ability to robustly handle incomplete data, we propose a robust alignment method (RobuAlign) in Multi-Modal 3D object detection. We design a novel corruption simulation method. During the preprocessing process, we use a variety of sampling methods on input images and point clouds to simulate different degrees of sensor failures. With the image encoder and point cloud encoder, we obtain image Bird's Eye View (BEV) features and LiDAR BEV features. To improve image BEV features, we apply depth prior generated by point cloud projection. At last, we combine the enhanced image BEV features with LiDAR BEV features. Our method achieves significant improvements (i.e., 32.69% mAP and 46.56% NDS).

1. Introduction

Multi-modal fusion promotes autonomous driving systems [1–3]. Different sensors can enhance each other. For example, camera data are superior in providing semantic and textural information in a perspective view, when estimating depth information from images, the accuracy of recovering 3D structures may be compromised due to the error of camera-LiDAR calibration. By contrast, point clouds provide much spatial topological structure and geometry information [4–7]. Integrating data from various sensors, such as 2D images, and LiDAR, and harmonizing their characteristics in a unified manner becomes pivotal.

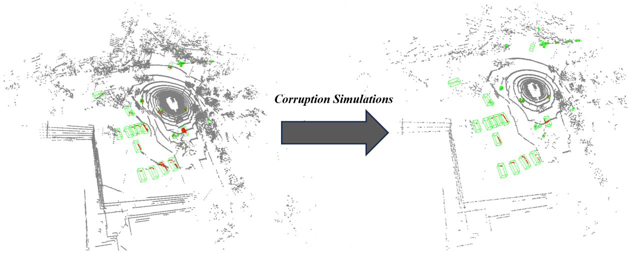


Figure 1. Through the proposed Corruption Simulation strategy, our model provides strong robustness even in the absence of LiDAR beams(*e.g.*, common corruptions and sensor failures).

Due to villainous weather, complex lighting conditions, and ill-posed scenes, unexpected situations such as hardware failures or sensor malfunctions are often encountered [8]. Existing LiDAR-camera fusion methods lack sufficient robustness in these ill-posed scenarios. Current multi-modal fusion methods may be influenced by missing information from images or inaccurate 3D depth signals from LiDAR sensors, leading to a degradation in performance under such extreme conditions [9, 10].

We analyzed the fundamental reasons for the lack of robustness in existing methods. Currently, multi-modal

Technical Report of the 2024 RoboDrive Challenge.
Track 5: Robust Multi-Modal BEV Detection.

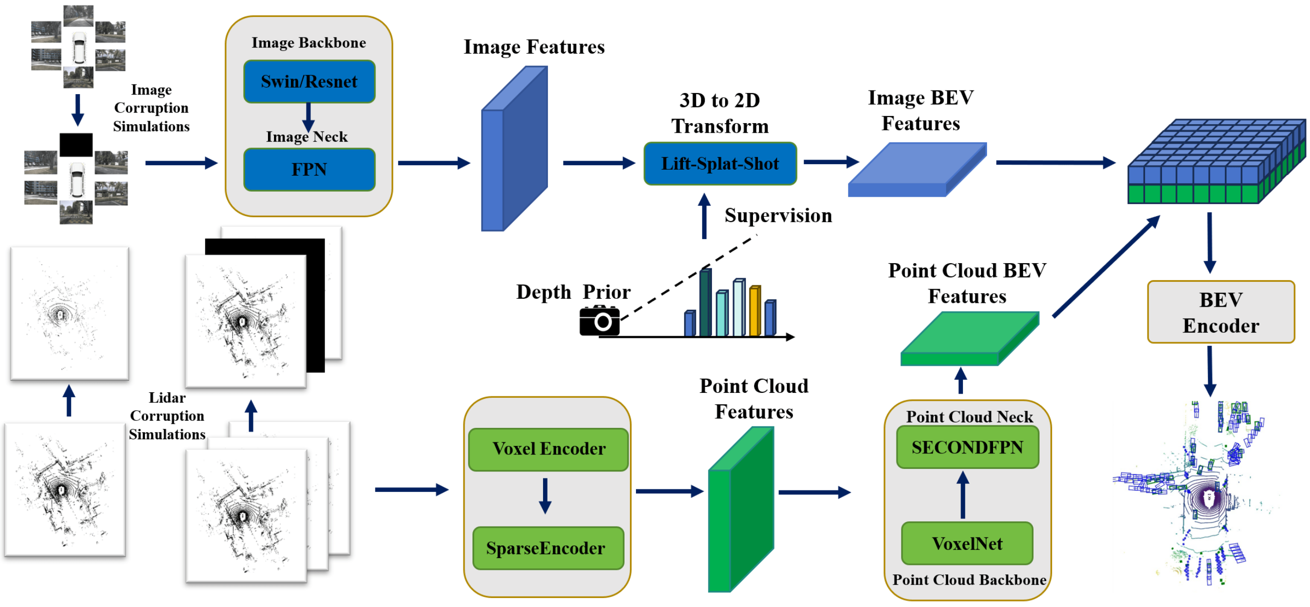


Figure 2. **The architecture of RobuAlign network.** RobuAlign consists of four parts: image feature extractor, point cloud feature encoder, multi-modal fusion module, and BEV Encoder.

3D object detection methods are mainly divided into hard-correlation [4, 11, 12] and soft-correlation methods [13, 14]: 1) Hard correlation methods are efficient, but it has inherent drawback lies in the offset between LiDAR and image calibration coefficients, as well as the projection errors in transforming images from the PV space to the Bird's Eye View (BEV) space. When data loss occurs due to LiDAR or camera failures, it exacerbates the inherent errors of hard correlation. 2) Soft-correlation methods mainly rely on the powerful information interaction capabilities of transform and cross-attention to associate LiDAR and image features. If point clouds are too sparse or the images drop frames, queries in the Transformer will be unable to retrieve corresponding relevant features, resulting in misalignment.

To address the aforementioned issues and enhance the robustness of the network to extreme driving scenarios, we propose a corruption simulation strategy. For simulating camera failures, we randomly mask out one perspective of the surround-view images. To simulate distortions in LiDAR data, we uniformly downsample the global point cloud data by 1/5 and 1/10 to represent different levels of LiDAR sensor distortions, as shown in Fig. 1. These two modalities of data augmentation strategies are performed simultaneously. Through continuous optimization of training strategies and experimental validation, adopting a 1/10 uniform downsampling strategy for LiDAR point clouds achieves optimal results in the twenty-fourth iteration, surpassing the baseline network NDS by 5.3% and mAP by 10.46%. With our corruption simulation 3D detection network can adapt to sudden or continuous sensor failures.

2. Related Work

In this section, we review various methods for object detection, including camera-based methods, LiDAR-based methods, and multi-sensor fusion methods.

2.1. Camera-Based Methods

Among various camera-based methods [15–20], those based on Bird's Eye View (BEV) seem to exhibit more prominent performance. BEVDet [15] is a camera-based method that performs 3D object detection in BEV. In camera-based 3D object detection, achieves significant improvement. BEVDet4D [16] is an enhanced version of BEVDet [15], which adds temporal cues. Specifically, after the View Transformer stage, it aligns and fuses the current frame's BEV features with the previous frame's BEV features, doubling the BEV features and allowing camera-based methods to achieve performance close to that of LiDAR-based methods.

2.2. LiDAR-Based Methods

LiDAR-based methods [5, 21, 22] use point clouds as input, which provide rich spatial information. With point clouds, accurate depth values can be obtained, which is an advantage that camera-based methods do not have [21, 23–28]. On the nuScenes [1] validation and test sets, LiDAR-based methods generally outperform camera-based methods. **Multi-Sensor Fusion Methods.** Current multi-sensor fusion works can be summarized as follows: some works [13, 14] use extracted features from images and point clouds as tokens, and then use Transformer [29] and its attention mechanism to predict 3D bounding boxes. Some works [4, 12] fuse image BEV features and point cloud features into a joint BEV, which is then processed through a BEV Encoder. Some other works [30] focus on preserving modality-specific useful information, instead of directly adopting multi-modal fusion.

3. Method

3.1. Overview

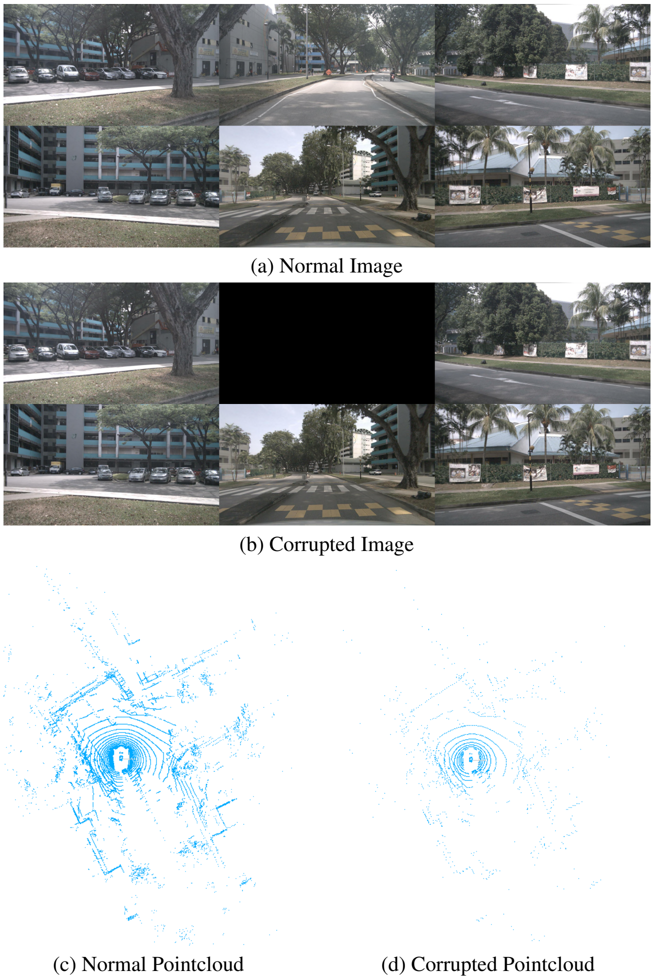
To address the challenges faced by existing LiDAR-camera fusion methods [4, 12–14, 30] in dealing with common corruption and sensor failures, we propose the RobuAlign method. Our overall architecture is shown in Fig. 2. At the pre-processing stage, we first apply a corruption simulation strategy by randomly masking one image and using uniform downsampling on the global point cloud to simulate multi-sensor failures.

For the image branch, F_{in} is first processed through an image encoder composed of ResNet [31], Swin Transformer [32], and FPN [33], yielding image features F_{out}. Subsequently, the Lift-Splat-Shoot [34] method is taken, in conjunction with the depth map F_{depth} generated from the point cloud, to convert image features to image BEV features, resulting in F_{bev}. For the point cloud branch, P_{in} is first preprocessed by a voxel encoder and an intermediate encoder, then point cloud features P_{out} are extracted through a point cloud encoder consisting of VoxelNet [35] and SECOND [36]. Finally, F_{out} and P_{out} are simply concatenated to obtain multi-modal BEV features of $MM_{\text{bev}} = Concat(F_{\text{bev}}, P_{\text{out}})$.

3.2. Simulation Strategy

In conventional training for 3D object detection, the possibility of sensor damage is typically not taken into account. Consequently, models trained in this manner may experience a decrease in accuracy in areas where sensors encounter issues. To enhance the overall robustness of the model in partial abnormal situations, we simulate corruption and sensor failures similar to those depicted in Fig. 3. Our corruption simulation involves two scenarios: damage to one perspective of the camera results in the failure of obtaining images, and malfunction of LiDAR sensor leads to sparse point cloud scans. To mimic the above scenarios, we propose a corruption simulation strategy. For one thing, we simulate the malfunction of a certain camera of camera clusters by discarding one image. For another, down-sampling is applied to the corresponding point cloud to mimic two levels of data missing.

Track 5 primarily explored the model's ability to handle partial missing sensor data. Normally, each sample includes a 360-degree view composed of six camera images (See Fig. 3(a)), a point clouds from a rooftop LiDAR (See Fig. 3(c)), and data from five millimeter-wave radars (radar was not considered in this competition). We noticed that in the test set, camera images might appear masked (See Fig. 3(b)), and point clouds may become sparser (See Fig. 3(d)). Therefore, we simulate these forms of corruption by randomly masking camera images and sub-sampling point clouds at ratios of 1/10 and 1/5 respectively.



(a) Normal Image

(b) Corrupted Image

(c) Normal Pointcloud

(d) Corrupted Pointcloud

Figure 3. Comparison of normal and corrupted data

3.3. Network

The input image tensor is $F_i \in \mathbb{R}^{BN \times 3 \times H \times W}$, where B is the batch size, N is the number of views, $H \times W$ is the size of images. We utilize Swin (including Swin-Tiny, Swin-Base, Swin-Large, etc.) [32] or ResNet [31] as the backbone to obtain image features at three scales, $F_1 \in \mathbb{R}^{BN \times 512 \times \frac{H}{8} \times \frac{W}{8}}$, $F_2 \in \mathbb{R}^{BN \times 1024 \times \frac{H}{16} \times \frac{W}{16}}$, and $F_3 \in \mathbb{R}^{BN \times 2048 \times \frac{H}{32} \times \frac{W}{32}}$ respectively. And FPN [33] as the neck of the image branch reduces the dimensions of the feature maps $F_{\text{out}} \in \mathbb{R}^{B \times N \times 128 \times \frac{H}{8} \times \frac{W}{8}}$.

We also need to transform the image features to image

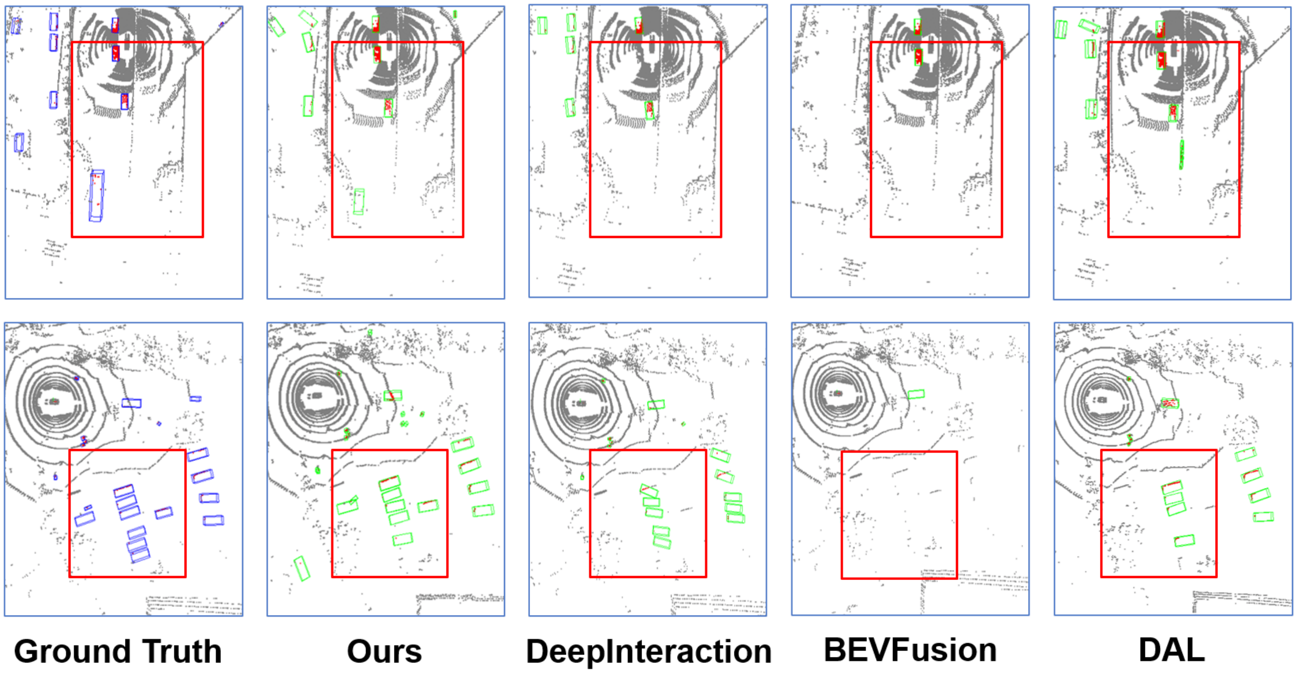


Figure 4. Qualitative comparison between RobuAlign and other SOTA methods on the nuScenes dataset. Purple boxes and green boxes are the predictions and ground truth, respectively. Best viewed with color and zoom-in.

BEV features, which requires depth estimation of image features. Benefiting from the Lift-Splat-Shot [34], we can obtain depth distribution of image features by using depth ground truth $F_{\text{depth}} \in \mathbb{R}^{BN\times 3\times H\times W}$,which is derived from the projection of LiDAR, to supervise the transformation process with the same loss function as BEVDepth [37]. Efficient Voxel Pooling module [9] is used to aggregate BEV features $F_{\text{bev}} \in \mathbb{R}^{N\times 32\times 256\times 256}$. For the point cloud branch, we first voxelize the point cloud with [35, 36, 38–40] to get the 3D voxel features $P_{\text{in}} \in \mathbb{R}^{N\times 5}$, where N indicates the number of voxels, 5 describes the number of features obtained per voxel. VoxelNet [35] is employed as the 3D Feature Extractor, we further use SECONDFPN [36] as our 3D Neck network to obtain LiDAR features $P_{\text{out}} \in \mathbb{R}^{B\times 384\times 256\times 256}$. Considering the efficiency of our network, we simply concatenate the LiDAR BEV features and image BEV features: $MM_{bev} \in R^{B\times 160\times 256\times 256} = Concat(F_{bev}, P_{out})$, and adopt the same effective post-processing process as DAL [12].

4. Experiments

4.1. Datasets

This work follows the protocol in the 2024 RoboDrive Challenge [41] when preparing the training and test data. Specifically, the model is trained on the official *train* split of the nuScenes dataset [1] (which contains 700 scenes) and tested on the held-out competition evaluation sets (which contains 150 scenes). The evaluation data was created following RoboDepth [8, 42, 43], RoboBEV [10, 44, 45], and Robo3D [46, 47]. The corruption types are mainly from three sources, namely camera corruptions, camera failures, and LiDAR failures. For more details, please refer to the corresponding GitHub repositories.

4.2. Evaluation Metrics

For 3D object detection, official pre-defined metrics from nuScenes [1] are as follows: mean Average Precision (mAP), Average Translation Error (ATE), Average Scale Error (ASE), Average Orientation Error (AOE), Average Velocity Error (AVE), Average Attribute Error (AAE), and nuScenes Detection Score (NDS).

4.3. Training Schedules

The framework is implemented using the PyTorch framework [48] and is based on the MMDetection3D codebase [49]. Our models are trained with a batch size of 24 on 3090 GPUs. We load the pre-trained ResNet101 and train our model for 24 epochs with CBGS using cycle learning rate policy with an initial value of 2.0×10^{-4}. Especially, we adjust the learning rate to 1/2 in the 20th iteration, and do the same in the 23rd iteration, affected by common corruption and sensor failure, the long-tail distribution phenomenon of data is more serious, so we use the same loss function as

Table 1. Ablation experiments on Robodrive-sensor-p2.

Versions	NDS	mAP	mATE	mASE	mAOE	mAVE	mAAE
DAL-Large [12]	41.26	22.20	40.97	28.89	60.42	45.26	22.86
version1	44.02	26.18	40.89	28.20	57.96	42.64	21.01
version2	45.07	28.61	39.34	27.89	57.46	45.32	22.28
version3	45.17	28.74	39.23	27.76	57.36	45.25	22.39
version4	46.20	32.43	41.64	28.12	61.35	46.53	22.45
version5	46.56	32.69	41.38	28.10	60.00	45.79	22.61

Table 2. Comparison against LiDAR-Camera Fusion Methods

Modal	NDS	mAP	mATE	mASE	mAOE	mAVE	mAAE
DeepInteraction-Base [30]	35.69	18.03	41.30	38.82	59.33	65.93	27.86
BEVFusion [4]	39.13	21.59	44.02	29.96	52.97	64.62	25.06
DAL-Large [12]	41.26	22.20	40.97	28.89	60.42	45.26	22.86
Ours	46.56	32.69	41.38	28.10	60.00	45.79	22.61

DAL [12] to enhance static data.

4.4. Ablation Study

We conducted ablation studies regarding: 1) image-processing backbone, 2) corruption simulations, and 3) training schedules.

We design 5 versions. Let us first introduce the common modules they use. For all versions, we randomly mask one camera image from surrounding views and use FPN [33] as the neck of the image branch, Lift-Splat-Shot [34] as the 2D to 3D Transformer, and VoxelNet [35] and SECOND [36] as the backbone and neck of the point cloud branch.

In the following, we introduce the differences between five versions:

- Version 1: We use ResNet50 [31] as the image-processing backbone, uniformly sub-sample the point cloud by 1/5 and apply image masking **with approximately 40%** of the overall dataset, training for 20 epochs.
- Version 2: We use ResNet50 [31] as the image-processing backbone, uniformly sub-sample the point cloud by 1/5 and apply image masking to the overall dataset, training for **24** epochs.
- Version 3: We use ResNet50 [31] as the image-processing backbone, uniformly sub-sample the point cloud by 1/5 and apply image masking to the overall dataset, training for 20 epochs.
- Version 4: We use ResNet50 [31] as the image-processing backbone, uniformly sub-sample the point cloud by **1/10** and apply image masking to the overall dataset, training for 24 epochs.
- Version 5: We use **ResNet101** [31] as the image-processing backbone, uniformly sub-sample the point cloud by **1/10** and apply image masking to the overall dataset, training for **30** epochs.

For the best-performing version 5, we switch to a larger backbone ResNet101 [31], use the most effective corruption simulation strategies from the previous versions, and appropriately extend the training schedule for 30 epochs. This ensures that the model can adequately fit the corruption data without overfitting, achieving the highest NDS among all the versions. As shown in Table. 1, Our network surpasses the baseline by 5.3% NDS, 10.49% mAP. The corruption simulation strategy has greatly improved the robust performance of the network, completing 3D detection tasks under extremely harsh conditions.

4.5. Comparison Against LiDAR-Camera Fusion Methods

As shown in Table. 2, our method, RobuAlign, outperforms the current predominant LiDAR-Camera Fusion methods [4, 12, 13, 30] on the robodrive leaderboard, significantly improves mAP by 14.66%, and NDS by 10.87%. The superior performance of our method is owed to our corruption simulation strategy and robustness-aware multi-modal fusion module. Our corruption simulation strategy provides our network training with various challenging data, improving the robustness, and alleviating the problem of sensor failures. On account of the property of the outdoor point clouds, namely sparsity and non-uniform density, it is prone to result in the lose of the 3D topology and geometric relations. Our robust multi-modal correlation structure and pre-training method alleviates this problem greatly. As shown in the red box region in the Fig. 4, we can accurately detect the object, regardless of whether it is close or far. It is obvious that the laser scan is missing to a certain extent, which has significant affects on the accurate detection of the objects by other SOTA methods.

5. Conclusion

In the 2024 RoboDrive Challenge's Track 5: Robust Multi-Modal BEV Detection, we propose a novel robust multi-modal alignment module with brilliant corruption simulations on the training set to enhance the model's robustness to partial missing sensor data and adopt different training schedules to ensure the model and data were well-fitted. Ultimately, these improvements led us to achieve third place.

References

[1] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11621–11631, 2020.

[2] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020.

[3] Lingdong Kong, Xiang Xu, Jiawei Ren, Wenwei Zhang, Liang Pan, Kai Chen, Wei Tsang Ooi, and Ziwei Liu. Multi-modal data-efficient 3d scene understanding for autonomous driving. *arXiv preprint arXiv:2405.05258*, 2024.

[4] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *IEEE International Conference on Robotics and Automation*, pages 2774–2781, 2023.

[5] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019.

[6] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by clip. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7030, 2023.

[7] Lingdong Kong, Youquan Liu, Runnan Chen, Yuexin Ma, Xinge Zhu, Yikang Li, Yuenan Hou, Yu Qiao, and Ziwei Liu. Rethinking range view representation for lidar segmentation. In *IEEE/CVF International Conference on Computer Vision*, pages 228–240, 2023.

[8] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Lai Xing Ng, Benoit Cottereau, and Wei Tsang Ooi. Robodepth: Robust out-of-distribution depth estimation under corruptions. *Advances in Neural Information Processing Systems*, 36, 2024.

[9] Junjie Huang and Guan Huang. Bevpoolv2: A cutting-edge implementation of bevdet toward deployment. *arXiv preprint arXiv:2211.17111*, 2022.

[10] Shaoyuan Xie, Lingdong Kong, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Benchmarking and improving bird's eye view perception robustness in autonomous driving. *arXiv preprint arXiv:2405.17426*, 2024.

[11] Charles R. Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas. Frustum pointnets for 3d object detection from RGB-D data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 918–927. Computer Vision Foundation / IEEE Computer Society, 2018.

[12] Junjie Huang, Yun Ye, Zhujin Liang, Yi Shan, and Dalong Du. Detecting as labeling: Rethinking lidar-camera fusion in 3d object detection. *arXiv preprint arXiv:2311.07152*, 2023.

[13] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1090–1099, 2022.

[14] Junjie Yan, Yingfei Liu, Jianjian Sun, Fan Jia, Shuailin Li, Tiancai Wang, and Xiangyu Zhang. Cross modal transformer via coordinates encoding for 3d object dectection. *arXiv preprint arXiv:2301.01283*, 2(3):4, 2023.

[15] Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021.

[16] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022.

[17] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.

[18] Tai Wang, ZHU Xinge, Jiangmiao Pang, and Dahua Lin. Probabilistic and geometric depth: Detecting objects in perspective. In *Conference on Robot Learning*, pages 1475–1485. PMLR, 2022.

[19] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *ICCV*, pages 1991–1999, 2019.

[20] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *ICCV*, pages 913–922, 2021.

[21] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11784–11793, 2021.

[22] Benjin Zhu, Zhengkai Jiang, Xiangxin Zhou, Zeming Li, and Gang Yu. Class-balanced grouping and sampling for point cloud 3d object detection. *arXiv preprint arXiv:1908.09492*, 2019.

[23] Youquan Liu, Lingdong Kong, Jun Cen, Runnan Chen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. Segment any point cloud sequences by distilling vision foundation models. In *Advances in Neural Information Processing Systems*, volume 36, 2024.

[24] Lingdong Kong, Jiawei Ren, Liang Pan, and Ziwei Liu. Lasermix for semi-supervised lidar semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21705–21715, 2023.

[25] Fangzhou Hong, Lingdong Kong, Hui Zhou, Xinge Zhu, Hongsheng Li, and Ziwei Liu. Unified 3d and 4d panoptic segmentation via dynamic shifting networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3480–3495, 2024.

[26] Shuo Wang, Xinhai Zhao, Hai-Ming Xu, Zehui Chen, Dameng Yu, Jiahao Chang, Zhen Yang, and Feng Zhao. Towards domain generalization for multi-view 3d object detection in bird-eye-view. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13333–13342, 2023.

[27] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer, 2022.

[28] Lingdong Kong, Niamul Quader, and Venice Erin Liong. Conda: Unsupervised domain adaptation for lidar segmentation via regularized domain concatenation. In *IEEE International Conference on Robotics and Automation*, pages 9338–9345, 2023.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017.

[30] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. 35:1992–2005, 2022.

[31] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[32] 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. In *ICCV*, pages 10012–10022, 2021.

[33] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017.

[34] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, pages 194–210. Springer, 2020.

[35] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018.

[36] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

[37] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1477–1485, 2023.

[38] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.

[39] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.

[40] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7652–7660, 2018.

[41] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Yaru Niu, Wei Tsang Ooi, Benoit R. Cottereau, Lai Xing Ng, Yuexin Ma, Wenwei Zhang, Liang Pan, Kai Chen, Ziwei Liu, Weichao Qiu, Wei Zhang, Xu Cao, Hao Lu, Ying-Cong Chen, Caixin Kang, Xinning Zhou, Chengyang Ying, Wentao Shang, Xingxing Wei, Yinpeng Dong, Bo Yang, Shengyin Jiang, Zeliang Ma, Dengyi Ji, Haiwen Li, Xingliang Huang, Yu Tian, Genghua Kou, Fan Jia, Yingfei Liu, Tiancai Wang, Ying Li, Xiaoshuai Hao, Yifan Yang, Hui Zhang, Mengchuan Wei, Yi Zhou, Haimei Zhao, Jing Zhang, Jinke Li, Xiao He, Xiaoqiang Cheng, Bingyang Zhang, Lirong Zhao, Dianlei Ding, Fangsheng Liu, Yixiang Yan, Hongming Wang, Nanfei Ye, Lun Luo, Yubo Tian, Yiwei Zuo, Zhe Cao, Yi Ren, Yunfan Li, Wenjie Liu, Xun Wu, Yifan Mao, Ming Li, Jian Liu, Jiayang Liu, Zihan Qin, Cunxi Chu, Jialei Xu, Wenbo Zhao, Junjun Jiang, Xianming Liu, Ziyan Wang, Chiwei Li, Shilong Li, Chendong Yuan, Songyue Yang, Wentao Liu, Peng Chen, Bin Zhou, Yubo Wang, Chi Zhang, Jianhang Sun, Hai Chen, Xiao Yang, Lizhong Wang, Dongyi Fu, Yongchun Lin, Huitong Yang, Haoang Li, Yadan Luo, Xianjing Cheng, and Yong Xu. The robodrive challenge: Drive anytime anywhere in any condition. *arXiv preprint arXiv:2405.08816*, 2024.

[42] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Benoit Cottereau, Lai Xing Ng, and Wei Tsang Ooi. The robodepth benchmark for robust out-of-distribution depth estimation under corruptions. https://github.com/ldkong1205/RoboDepth, 2023.

[43] Lingdong Kong, Yaru Niu, Shaoyuan Xie, Hanjiang Hu, Lai Xing Ng, Benoit Cottereau, Ding Zhao, Liangjun Zhang, Hesheng Wang, Wei Tsang Ooi, Ruijie Zhu, Ziyang Song, Li Liu, Tianzhu Zhang, Jun Yu, Mohan Jing, Pengwei Li, Xiaohua Qi, Cheng Jin, Yingfeng Chen, Jie Hou, Jie Zhang, Zhen Kan, Qiang Lin, Liang Peng, Minglei Li, Di Xu, Changpeng Yang, Yuanqi Yao, Gang Wu, Jian Kuai, Xianming Liu, Junjun Jiang, Jiamian Huang, Baojun Li, Jiale Chen, Shuang

Zhang, Sun Ao, Zhenyu Li, Runze Chen, Haiyong Luo, Fang Zhao, and Jingze Yu. The robodepth challenge: Methods and advancements towards robust depth estimation. *arXiv preprint arXiv:2307.15061*, 2023.

[44] Shaoyuan Xie, Lingdong Kong, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Robobev: Towards robust bird's eye view perception under corruptions. *arXiv preprint arXiv:2304.06719*, 2023.

[45] Shaoyuan Xie, Lingdong Kong, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. The robobev benchmark for robust bird's eye view detection under common corruption and domain shift. https://github.com/Daniel-xsy/RoboBEV, 2023.

[46] Lingdong Kong, Youquan Liu, Xin Li, Runnan Chen, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Robo3d: Towards robust and reliable 3d perception against corruptions. In *IEEE/CVF International Conference on Computer Vision*, pages 19994–20006, 2023.

[47] Lingdong Kong, Youquan Liu, Xin Li, Runnan Chen, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. The robo3d benchmark for robust and reliable 3d perception. https://github.com/ldkong1205/Robo3D, 2023.

[48] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, 2019.

[49] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. https://github.com/open-mmlab/mmdetection3d, 2020.