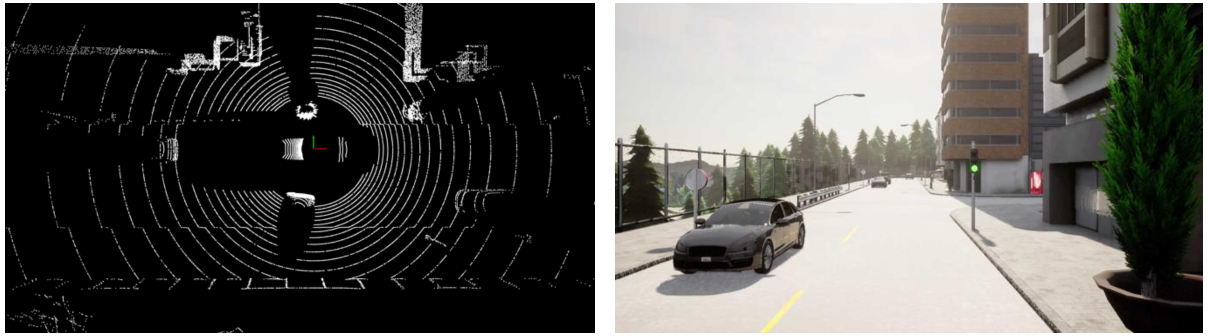
JiSAM: Alleviate Labeling Burden and Corner Case Problems in Autonomous Driving via Minimal Real-World Data

Runjian Chen, Wenqi Shao*, Bo Zhang, Shaoshuai Shi, Li Jiang, Ping Luo*

Motivation

- LiDAR perception heavily relies on labeled real-world data, which is notoriously time-and-energy-consuming [1,2] to annotate and lacks corner cases like rare traffic participants. On the contrary, generating labeled LiDAR point clouds with corner cases is a piece of cake in simulators. Thus, we explore how to use synthetic point clouds to benefit real world 3D perception with minimal real data (eg. 2.5% of all LiDAR frames in NuScenes).



Simulation Data

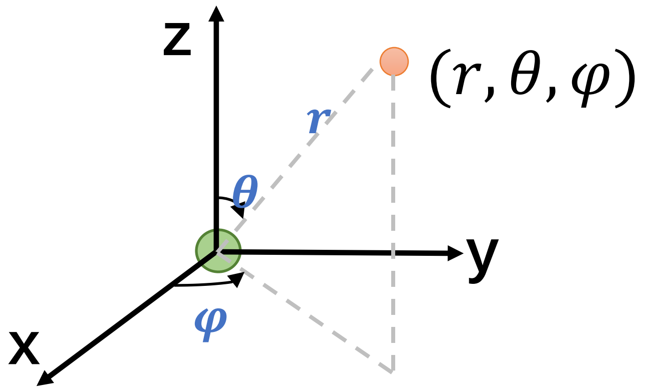
Real Data

Previous Works and Challenges

- It is non-trivial to incorporate simulation point clouds for real-world 3D perception. [3,4,5] directly incorporate synthetic point cloud to train models but still lags far behind models trained on real data.
- Two main challegenes:
- Sample efficiency of synthetic data. Although the amount of synthetic data can be unlimited, training time and storage are restricted.
- Simulation-to-real gap including unrealistic intensity and object shape difference. Intensity in CARLA is computed via a linear function of xyz location, making it meaningless. Object shapes in real world and CARLA varies, leading to various local point distributions.

Method

➢ **Jittering Augmentation.** Set noise level to zero when collecting simulation point clouds. During training, we convert points into Spherical Coordinate and apply random jittering to augment data (then back to Cartesian), improving distribution diversity in local areas.

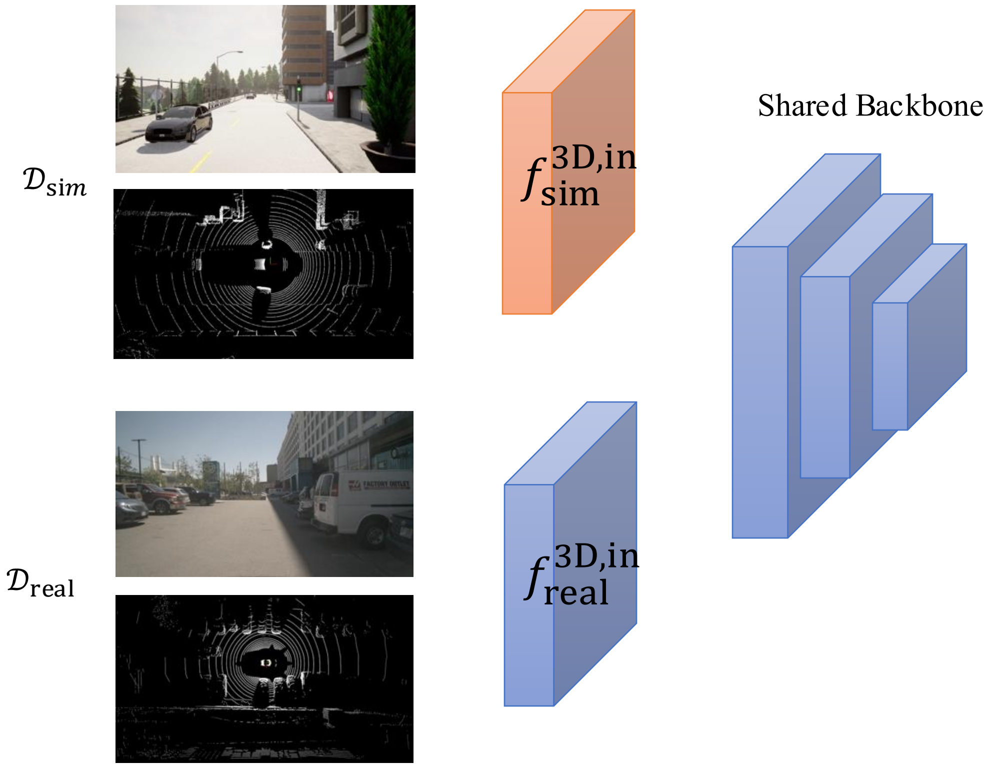


$\dot{r} = r + \delta_r$ $\dot{\theta} = \theta + \delta_\theta$ $\dot{\varphi} = \varphi + \delta_\varphi$

$\delta_r \sim N(0,\Delta_r)$ $\delta_\theta \sim N(0,\Delta_\theta)$ $\delta_\varphi \sim N(0,\Delta_\varphi)$

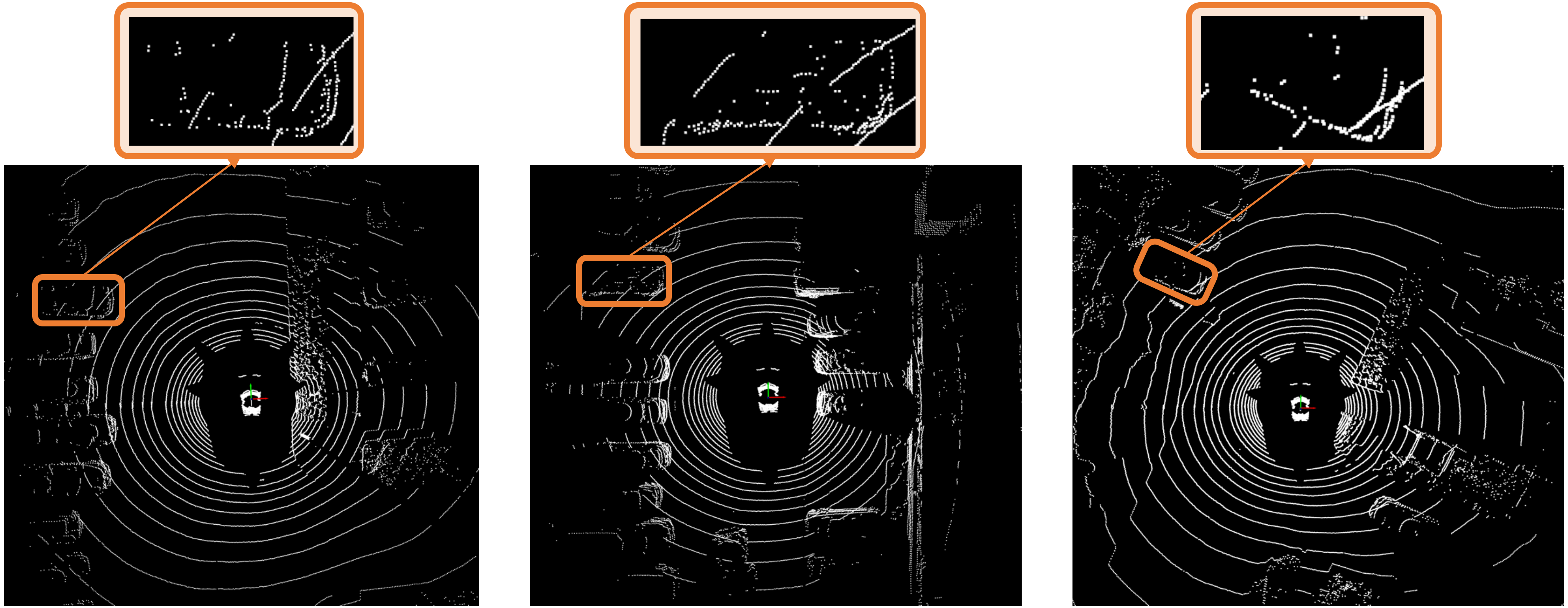
Method

➢ **Domain-aware Backbones.** Point clouds in different domains have different numbers of features. Thus we apply separate input blocks for different domains, which only brings less than 0.025% overhead during training and make full use of all informative channels.



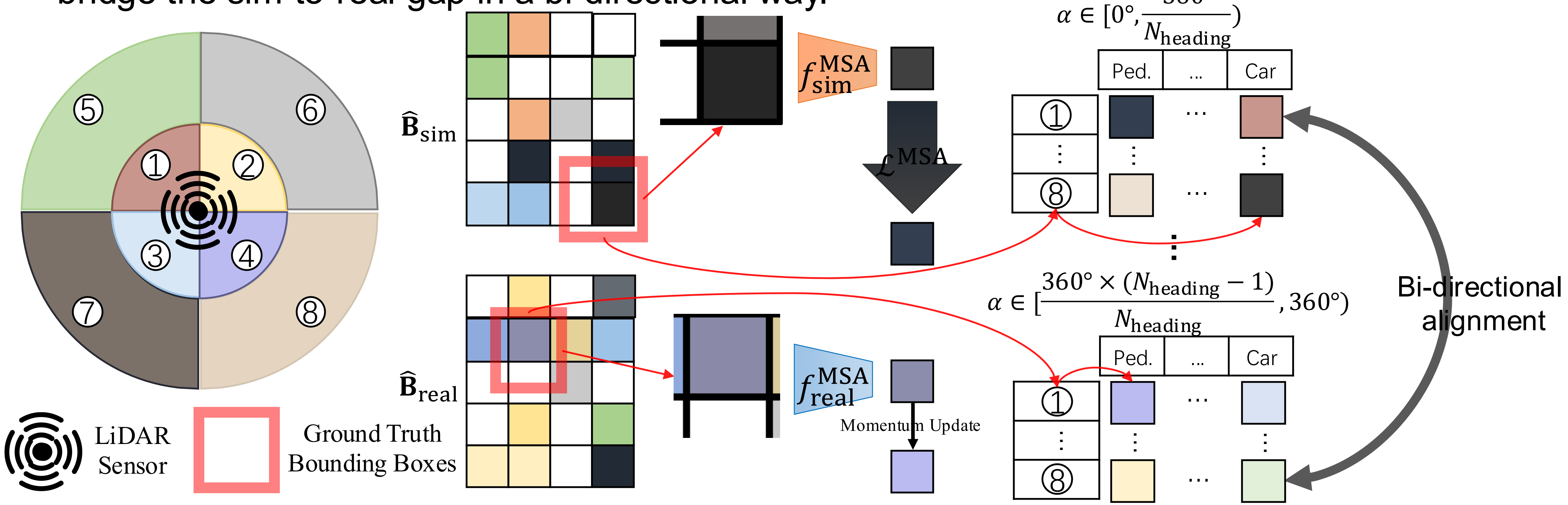
Method

➢ **Memory-based Sectorized Alignment.** We observe that objects in the same class with similar headings in the same partition of the surrounding environment, have similar point distributions, which motivates us to propose Memory-based Sectorized Alignment Loss.



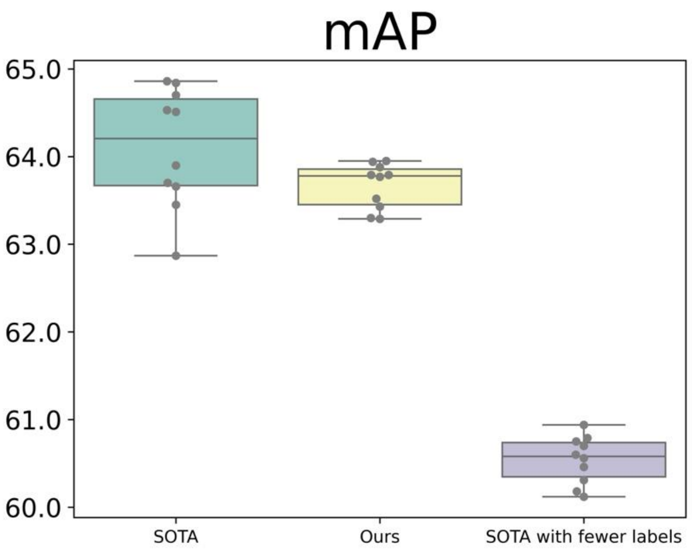
Method

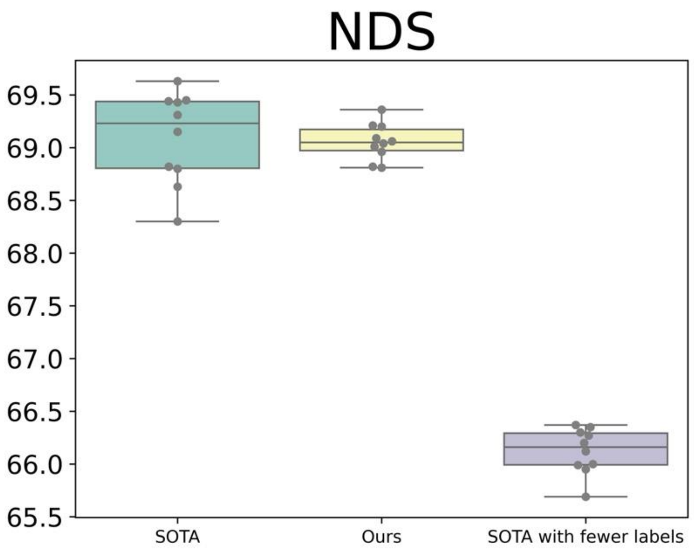
➢ **Memory-based Sectorized Alignment.** Divide surrounding scene of ego-vehicle using shape context. Build memory bank for each class in each partition within each discretized heading angles. Keep update memory bank during training and conduct alignment loss to bridge the sim-to-real gap in a bi-directional way.



Experiments

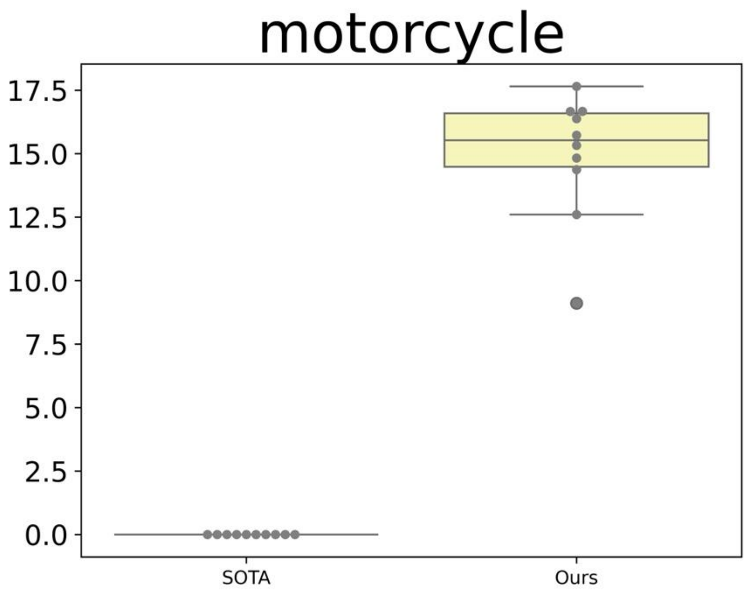
- **Reduce Labeling Burden.** We label 2.5% of all LiDAR frames in NuScenes [1] and generate 80,000 labeled point clouds in CARLA to train 3D object detector Transfusion [6]. Meanwhile, we also train Transfusion with all available data in NuScenes. JiSAM achieves comparable performance with a small amount of real labels.





Experiments

➢ **Corner Case Problem.** We manually eliminate motorcycle labels in real dataset. JiSAM achieves ~16 mAP for missing class.



Experiments

➢ **Ablation Study** shows the effectiveness of each component

Syn.	D.S.B.	S.A.	J.A.	mAP	NDS
✗	✗	✗	✗	60.18	65.69
✔	✗	✗	✗	58.29 (-1.89)	64.71 (-0.98)
✓	✓	✗	✗	62.30 (+2.12)	68.07 (+2.38)
✓	✓	✓	✗	63.71 (+3.53)	69.09 (+3.40)
✓	✓	✗	✓	63.28 (+3.10)	68.89 (+3.20)
✓ (50%)	✓	✓	✓	63.90 (+3.72)	68.97 (+3.28)
✓	✓	✓	✓	63.95 (+3.77)	69.36 (+3.67)

Experiments

➢ **Comparison to previous detectors.**

Model	Real	Sim	mAP	NDS
Tranfusion [1]	✓	✗	64.51	69.31
Tranfusion with JiSAM	✗	✓	63.95	69.36
VoxelNext [6]	✓	✗	60.53	66.65
CenterPoint [39]	✓	✗	59.22	66.48
Second [35]	✓	✗	50.59	62.29
PointPillar [18]	✓	✗	44.63	58.23

Table 1. Comparison to different 3D object detectors. "Real" means whether the model is trained on all real labels in NuScenes. "Sim" indicates whether the model is trained with simulation data.

Discussion

- **Seamless Integration with Other 3D Detectors.** JiSAM is a plug-and-play module. It can be seamlessly integrated with other 3D detectors with minimal adjustment.
- **Orthogonal to State-of-The-Art LiDAR Generative Model.** Performance might be further improved when we use LiDAR point clouds generated by SOTA generative models.

Paper **Code** **Personal Website**

Citations

[1] Caesar H, Bankiti V, Lang A H, et al. nuscenes: A multimodal dataset for autonomous driving[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 11621-11631.

[2] Wang T, He C, Wang Z, et al. Flava: Find, localize, adjust and verify to annotate lidar-based point clouds[C]//Adjunct Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology. 2020: 31-33.

[3] DeBortoli R, Fuxin L, Kapoor A, et al. Adversarial training on point clouds for sim-to-real 3d object detection[J]. IEEE Robotics and Automation Letters, 2021, 6(4): 6662-6669.

[4] Zhang B, Cai X, Yuan J, et al. Resimad: Zero-shot 3d domain transfer for autonomous driving with source reconstruction and target simulation[J]. arXiv preprint arXiv:2309.05527, 2023.

[5] Wu C, Bi X, Pfrommer J, et al. Sim2real transfer learning for point cloud segmentation: An industrial application case on autonomous disassembly[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2023: 4531-4540.

[6] Bai X, Hu Z, Zhu X, et al. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 1090-1099.