SegSy3D: Segmentation-Guided Self-Training and Model Synergy for Cross-Platform 3D Detection

Yongchun Lin
Guangdong University of Technology
linyongchun@mails.gdut.edu.cn

Huitong Yang
The University of Queensland
huitongy0126@gmail.com

Liang Lei
Guangdong University of Technology
leiliang@gdut.edu.cn

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

Xinliang Zhang
SenseTime
zhangxinliang@sensetime.com

Zhiyong Wang
SenseTime
wangzhiyong@sensetime.com

Xiaofeng Wang
SenseTime
wangxiaofeng@senseauto.com

Abstract

SegSy3D *(Segmentation-Guided Self-Training and Model Synergy) is a cross-platform 3D object detection framework that addresses large domain gaps via domain adaptation from vehicle-mounted sensors to drones and quadruped robots. We employ segmentation-guided pseudo-label refinement together with a joint self-training objective on preserved high-confidence pseudo-labels, yielding consistent gains when transferring from labeled vehicle data to unlabeled robotic platforms with heterogeneous sensors. At test time, we adopt a Model Synergy strategy that dynamically selects historical checkpoints with diverse knowledge and assembles them into a synergistic ensemble to enhance cross-platform robustness. Evaluated on RoboSense Challenge 2025 Track 5, SegSy3D achieved 56.14 $AP^{40}_{0.5}$ for Car, 55.07 $AP^{40}_{0.5}$ for Pedestrian, and an overall mAP of 55.61, ranking third in Phase 2. These results demonstrate its effectiveness in bridging domain gaps, enhancing multi-platform 3D perception, and providing a practical solution for real-world robotic deployments.*

1. Introduction

Accurate 3D object detection is essential for autonomous robotic systems, enabling precise localization, volumetric estimation, and motion prediction in applications from autonomous driving to aerial surveillance and ground robotics [1–14]. As robotics expands beyond automotive domains to heterogeneous platforms like drones and quadruped robots, there is a growing demand for 3D detection systems that can operate under diverse perception constraints and environmental conditions [15–25]. These capabilities are critical for safe navigation, cooperative decision-making, and long-term autonomy in unstructured environments [23, 26–32].

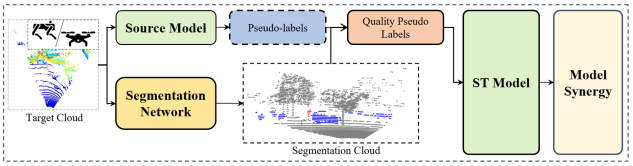

Figure 1. **SegSy3D.** A source detector generates pseudo-labels on target point clouds; a segmentation network refines them into quality pseudo-labels; the refined labels drive self-training to adapt the detector; at inference, Model Synergy assembles prior checkpoints for robust cross-platform detection.

Recent advances in LiDAR-based 3D object detection have achieved remarkable progress for automotive applications. Foundational works like PointNet [33] and PointNet++ [34] established effective point cloud feature learning paradigms, while methods including PointRCNN [35], PointPillars [36], PV-RCNN [37], SECOND [38], and CenterPoint [39] achieved state-of-the-art performance. Large-scale benchmarks like KITTI [4], nuScenes [2], and Waymo [3] have further accelerated progress by enabling systematic evaluation and comparison.

Despite these advances, models designed for road-driving scenarios often degrade significantly when applied to UAVs or legged robots [40, 41]. The challenges arise from (i) geometric domain shifts, *e.g.*, top-down viewpoints of drones versus near-ground views of quadrupeds; (ii) distributional discrepancies in point density, occlusion patterns, and scene

layout; and (iii) annotation scarcity, as 3D labeling for non-vehicle platforms is labor-intensive and costly. These factors hinder direct transferability and highlight the necessity of domain-adaptive frameworks [42–47].

To address these issues, unsupervised domain adaptation (UDA) and domain generalization (DG) methods have been explored [48–51]. Early approaches focused on pseudo-labeling to transfer supervision from source to target domains [40, 52, 53], generating target-domain labels using source-trained models. Building upon this idea, ST3D [41] introduced a self-training framework that iteratively refines pseudo-labels on the target domain, improving detection accuracy and robustness. Subsequent works further enhanced this framework by incorporating geometry-aware prototype alignment (GPA-3D) [54], semi-supervised domain adaptation (SSDA3D) [55], and adversarial adaptation techniques (UADA3D) [56, 57], enabling more effective cross-domain feature alignment and mitigating the noise in pseudo-labels. Complementary strategies include adversarial learning and domain-invariant representation extraction [49, 50, 58]. However, most prior studies remain constrained to vehicle-to-vehicle settings (*e.g.*, KITTI →nuScenes), leaving cross-platform adaptation relatively underexplored.

In this work, we propose **SegSy3D** (Segmentation-Guided Self-Training and Model Synergy), a cross-platform adaptation framework built upon Voxel R-CNN with Gaussian Blobs (GBlobs) [59] that transfers detectors trained on vehicle-mounted sensors to drone and quadruped robot platforms. An overview is shown in Fig. 1. SegSy3D integrates: **(1)** Progressive self-training for transferring knowledge from labeled sources to unlabeled targets; **(2)** A segmentation-guided refinement model to upgrade pseudo-label quality; and **(3)** Model Synergy, which adaptively selects and assembles the most suitable prior checkpoints into a unified meta-model, leveraging long-term information to guide the current test batch and enabling multi-temporal test-time adaptation and model fusion [60]. Overall, SegSy3D advances robust cross-platform 3D perception by effectively bridging large domain gaps and leveraging complementary strengths of multiple adapted detectors.

2. Methodology

2.1. Overview

SegSy3D is a self-training framework for cross-platform 3D object detection without target-domain annotations (Fig. 2). Starting from a detector pre-trained on labeled source data, we (i) generate pseudo-labels on unlabeled target point clouds; (ii) refine them with a source-trained segmentation model [61] and confidence calibration; (iii) adapt the detector via progressive self-training [41]; and (iv) at inference, apply *Model Synergy* [60] to select prior checkpoints, weight them, and assemble a unified predictor for robust cross-platform fusion.

We follow Voxel R-CNN [62] and employ GBlob-sVFE [59] to encode per-voxel covariance from point deviations and concatenate it with voxel features, capturing fine-grained geometry to mitigate cross-platform shifts. The rest is standard: sparse 3D convolutions, height compression to bird's-eye view, 2D aggregation; a *CenterHead* proposes candidates, and the Voxel R-CNN head refines them via RoI-grid pooling and MLPs for box regression and classification.

2.2. Segmentation-Guided Pseudo-Label Generation and Refinement

The quality of pseudo-labels is crucial for effective self-training. Conventional approaches typically rely only on detection confidence, which may discard true positives with low scores or retain noisy false positives under domain shift. To overcome this limitation, we propose a cross-domain guided pseudo-label generation strategy that combines semantic segmentation cues with detector outputs.

Specifically, we first train a segmentation network (CD-SegNet) [61] on the labeled source domain. This network is then applied to target domain point clouds to produce class-specific point-level predictions, such as car point clouds P_i^{car} and pedestrian point clouds $P_i^{\text{pedestrian}}$. These semantic predictions highlight regions likely to contain objects of interest. By intersecting these regions with detection outputs $\tilde{L}_i^{\text{t}}$, we obtain segmentation-guided labels $\tilde{L}_i^{\text{seg}}$ that help recover potential true positives.

Meanwhile, the detection outputs are filtered using confidence thresholds: high-confidence predictions are retained as positive labels $\tilde{L}_i^{\text{P}}$, while low-confidence ones are used as negatives $\tilde{L}_i^{\text{N}}$. To construct the final pseudo-label set for training, we combine both sources:

$$\tilde{L}_i^{\text{all\_P}} = \{\tilde{L}_i^{\text{P}}, \tilde{L}_i^{\text{seg}}\}, \tag{1}$$

where $\tilde{L}_i^{\text{all\_P}}$ are our self-training positive labels. This integration offers two key benefits: (1) segmentation cues provide complementary semantic information that compensates for the weaknesses of detection confidence under domain shifts, and (2) confidence filtering ensures reliability by suppressing noisy detections. As a result, the generated pseudo-labels are both more accurate and more diverse, significantly enhancing self-training and improving cross-domain 3D object detection performance.

2.3. Self-Training

Our procedure follows the general ST3D pipeline [41] with extensions for cross-domain adaptation. We first train a detector on the source domain and generate pseudo-labels for the target domain using the method in Sec. 2.2. The refined pseudo-labels, obtained by incorporating semantic cues from a source-trained segmentation model together with

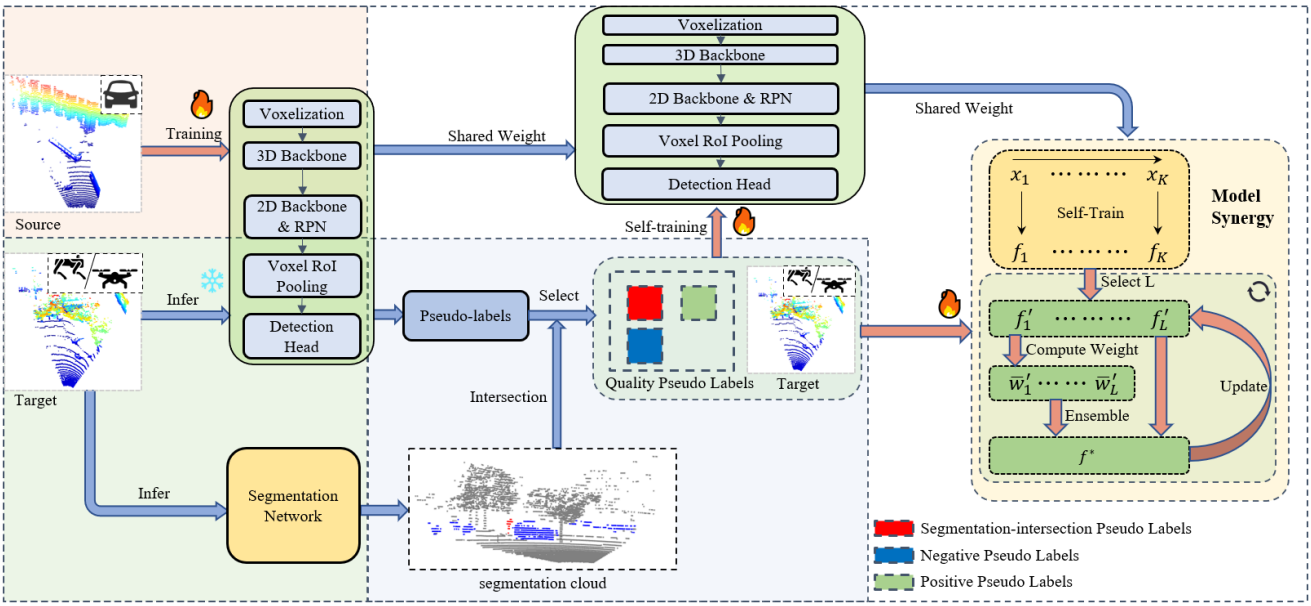

Figure 2. Architecture of SegSy3D. A detector pre-trained on the labeled source domain first produces pseudo-labels on unlabeled target scans. A source-trained segmentation network provides geometry-consistent masks to refine these labels; the retained high-quality labels supervise progressive self-training to adapt the detector to the target domain. At test time, Model Synergy (MOS) selects suitable prior checkpoints, estimates ensemble weights, and assembles a unified model for multi-temporal adaptation and robust fusion. Overall, SegSy3D couples refined pseudo-label self-training with MOS to strengthen cross-domain robustness.

confidence filtering, are then used to retrain the detector on the target domain, yielding the adapted model f_{ST}

To further enhance adaptation, we apply Model Synergy for test time adaptation (MOS) [60] starting from f_{ST} as initialization. MOS operates over a bank of historical checkpoints collected during self-training and, guided by the target data, adaptively selects and assembles them to yield a second model f_{MOS}. This MOS refinement continues test-time adaptation without labels, injecting complementary long-term knowledge and closing remaining domain gaps.

During inference, we adopt a class-specific fusion rule: pedestrian predictions are taken from f_{ST} and car predictions from f_{MOS}. This multi-temporal fusion exploits the complementary advantages of the two models, where f_{ST} benefits from self-training with segmentation guidance and f_{MOS} leverages MOS adaptation to better capture target-specific variations. Formally, the final prediction B_{result} set is

$$B_{\text{result}} = B^{\text{pedestrian}}_{f_{\text{ST}}} \cup B^{\text{car}}_{f_{\text{MOS}}}. \tag{2}$$

This strategy provides a principled mechanism for integrating complementary adaptation processes into a single predictor, delivering robust improvements across object categories and platforms.

2.4. Training Objectives

We use the CornerNet modified focal loss [63] to emphasize peak responses and down-weight non-peaks, providing effective foreground–background rebalancing with limited memory overhead. Center and box parameters are trained with masked ℓ_1 regression in the CenterPoint style [39] (valid-target masking, code-wise aggregation, localization scaling). For region-wise predictions, we adopt BCE with logits for classification and Smooth-ℓ_1 (Huber) for box refinement [64], with all terms normalized by the count of valid samples.

Let $\mathcal{L}^{(h)}_{\text{focal}}$ and $\mathcal{L}^{(h)}_{\ell_1}$ denote the heatmap and masked-ℓ_1 losses at feature level h, and let $\mathcal{L}^{\text{roi}}_{\text{BCE}}$ and $\mathcal{L}^{\text{roi}}_{\text{Smooth-}\ell_1}$ be the ROI classification and regression losses. The overall objective is

$$\begin{aligned}\mathcal{L}_{\text{total}} = \sum_h \Big(\lambda_{\text{hm}}\, \mathcal{L}^{(h)}_{\text{focal}} + \lambda_{\text{loc}}\, \mathcal{L}^{(h)}_{\ell_1}\Big) \\ + \ \lambda_{\text{roi-cls}}\, \mathcal{L}^{\text{roi}}_{\text{BCE}} + \ \lambda_{\text{roi-reg}}\, \mathcal{L}^{\text{roi}}_{\text{Smooth-}\ell_1}.\end{aligned} \tag{3}$$

Unless stated otherwise, we use unit code weights and set λ_{hm}=1.0, λ_{loc}=2.0, $\lambda_{\text{roi-cls}}$=1.0, and $\lambda_{\text{roi-reg}}$=1.0.

3. Experiments

3.1. Dataset

We use the official data provided by the *RoboSense Challenge 2025* [70] held at IROS 2025. This competition builds upon the legacy of the *RoboDepth Challenge 2023* [71, 72] at ICRA 2023 and the *RoboDrive Challenge 2024* [73, 74]

Method	Car	Pedestrian	mAP
Source-only			
PV-RCNN [37]	22.24	37.54	29.89
LION [65] w/ Mamba [66]	39.98	38.85	39.42
LION w/ Retnet [67]	33.87	44.41	39.14
LION w/ Rwkv [68]	31.43	35.46	33.44
Part-A^2 [69] w/ GBlobs [59]	45.96	39.80	42.88
PointPil. [36] w/ GBlobs	37.13	36.44	36.79
Centerpoint [39] w/ GBlobs	45.45	42.75	44.10
Voxel R-CNN [62] w/ GBlobs	51.11	46.04	48.57
Self-training			
st3d	28.97	43.51	36.24
st3d++	28.53	41.49	35.01
Ours	**56.14**	**55.07**	**55.61**

Table 1. Performance of different models on cross-platform 3D object detection. The upper block reports results of source-only baselines trained on the source domain and directly evaluated on the Phase 2 Quadruped target-domain dataset, while the lower block presents results of self-training methods. All results are reported using AP_{3D} at IoU threshold 0.5 with 40 recall positions, denoted as $AP^{40}_{0.5}$.

at ICRA 2024, continuing the collective effort to advance robust and scalable robot perception. Each track in this competition is grounded on an established benchmark designed for evaluating real-world robustness and generalization [6, 30, 75–77]. Specifically, this task is built upon the **Pi3DET** benchmark [6] in **Track 5**, which studies cross-platform LiDAR-based 3D object detection across vehicle, drone, and quadruped platforms through viewpoint normalization and unified pre-training [78].

The source domain contains 10,031 labeled LiDAR scenes with 3D bounding-box annotations from the Vehicle platform, while the target domains are 5,885 unlabeled scenes from the Drone platform (Phase 1) and 8,025 unlabeled scenes from the Quadruped platform (Phase 2). The task focuses on cross-platform adaptation, where significant distribution gaps and the absence of labels in the target domains pose a challenging problem of unsupervised domain adaptation across heterogeneous robotic platforms, reflecting real-world deployment scenarios in autonomous driving, aerial perception, and quadruped navigation.

Evaluation Metric. We adopt the KITTI evaluation metric [4] for evaluating our methods. Specifically, we report the 3D Average Precision (AP) over 40 recall positions with an IoU threshold of 0.5. In Phase 2, we evaluate on the Car and Pedestrian categories, denoted as Car 3D $AP^{40}_{0.5}$ and Pedestrian 3D $AP^{40}_{0.5}$, and take their mean as the final mAP score.

3.2. Implementation Details

We implement our framework in PyTorch [79] based on the OpenPCDet codebase [80]. For source-domain training, we adopt the Adam optimizer [81] with a OneCycle learning rate schedule [82] for 30 epochs, using a batch size of 26 per GPU and an initial learning rate of 0.01, following common practice in 3D object detection [59]. The input point clouds are cropped to the range [0, -75.2, -2, 75.2, 75.2, 4] m, and the voxel size is set to (0.1, 0.1, 0.15) m. During self-training, we train for 16 epochs with a reduced initial learning rate of 0.005.

3.3. Comparative Study

As shown in the upper block of Table 1, all models are trained on the source domain and directly evaluated on the Phase 2 Quadruped target set. The official PV-RCNN baseline yields the lowest performance, with 22.24 $AP^{40}_{0.5}$ for Car, 37.54 for Pedestrian, and an overall mAP of 29.89, highlighting the large cross-platform gap. Other models, including Mamba Lion, RetNet Lion, and CenterPoint GBlobs, achieve intermediate results, showing the benefit of GBlobs-based architectures. Voxel R-CNN GBlobs achieves the best performance, with 51.11 for Car, 46.04 for Pedestrian, and an overall mAP of 48.57, demonstrating strong source-only generalization.

As shown in the lower block of Table 1, we compare self-training methods on the Phase 2 Quadruped target domain. The PV-RCNN-based ST3D and ST3D++ baselines achieve 36.24 and 35.01 mAP, respectively, showing moderate improvements over source-only training. In contrast, **SegSy3D**—built on Voxel R-CNN with GBlobs, initialized on the labeled source domain, optimized using the pseudo-label set $\tilde{L}_i^{\text{all\_P}}$ defined in Eq. (1), and further refined via segmentation-guided pseudo-label refinement, MOS-based test-time assembly, and model fusion—achieves the best overall performance. SegSy3D reaches 56.14 $AP^{40}_{0.5}$ for Car, 55.07 $AP^{40}_{0.5}$ for Pedestrian, and an overall mAP of 55.61, clearly validating its effectiveness for cross-platform adaptation. The significant improvement over baseline models demonstrates the advantage of combining high-quality pseudo-label selection with temporal consistency and model synergy. Moreover, SegSy3D consistently improves detection across both large and small objects, indicating robust feature adaptation. These results highlight that segmentation-guided refinement effectively filters noisy pseudo-labels, while MOS-based fusion leverages complementary detector strengths to achieve superior multi-platform 3D perception.

3.4. Ablation Study

Table 2 reports the results of our ablation study on Phase 2, evaluated by 3D $AP^{R40}_{0.5}$ for Car, Pedestrian, and their mean (mAP). The Default setting corresponds to the baseline model trained with ST3D. Incorporating segmentation-

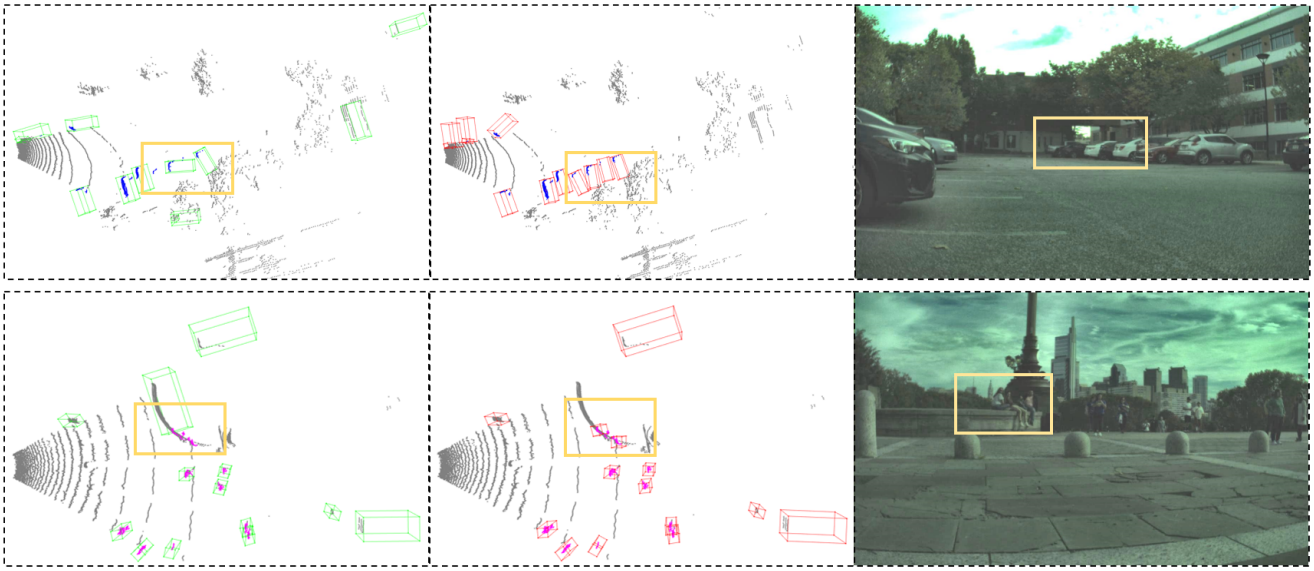

Figure 3. Qualitative comparison: left is the official baseline, middle is **SegSy3D**, and right is the camera view. Yellow boxes highlight the regions of interest; green/red boxes denote predicted bounding boxes; blue points are segmented cars, and magenta points indicate segmented pedestrians. All point clouds are from the Phase 2 Quadruped target domain.

guided pseudo-label refinement (L_i^{seg}, Sec. 2.2) clearly improves performance, demonstrating the benefit of filtering noisy labels. Further introducing Model Synergy (MOS) for multi-temporal optimization leads to additional gains, highlighting the effectiveness of temporal consistency in self-training. Finally, the full **SegSy3D** framework, which fuses complementary adapted models, achieves the best overall performance, significantly surpassing the baseline and validating the advantage of our complete approach.

Unlike filtering based solely on detector confidence, segmentation-guided refinement exploits semantic cues from the source domain to (i) recover true positives with low confidence that are geometrically consistent and (ii) suppress false positives with high confidence that are geometrically inconsistent. As illustrated in Fig. 3, these cues enable recovery of geometry-consistent low-score objects. This yields pseudo-labels with higher precision and recall, providing a more reliable self-training signal. Together with MOS, which dynamically assembles complementary checkpoints at test time, the adapted detector achieves stronger generalization across platforms and object scales.

4. Conclusion

We presented **SegSy3D**, a cross-platform 3D object detection framework that transfers knowledge from labeled vehicle data to unlabeled robotic platforms. It integrates segmentation-guided self-training on refined pseudo-labels with a test-time Model Synergy that selects and assembles prior checkpoints into a unified predictor for robust inference. The framework effectively reduces domain gaps across heterogeneous sensors and platforms. Limitations include additional compute and memory for synergy; future work will explore parameter-efficient synergy and budgeted checkpoint selection, distillation into compact students, temporal test-time adaptation, and uncertainty-aware pseudo-labeling for scalable deployment.

Method	Car	Pedestrian	mAP
Our w/o ($\tilde{L}_i^{seg}$ + MOS)	53.81	53.72	53.77
Our w/o MOS	56.01	**55.07**	55.54
Ours	**56.14**	**55.07**	**55.61**

Table 2. Component ablation studies. Models are first trained on the source domain and then adapted via self-training on the Phase 2 Quadruped target-domain data.

References

[1] Eduardo Arnold, Omar Y. Al-Jarrah, Mehrdad Dianati, Saber Fallah, David Oxtoby, and Alex Mouzakitis. A survey on 3d object detection methods for autonomous driving applications. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3782–3795, 2019. 1

[2] 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. 1

[3] 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. 1

[4] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. 1, 4

[5] Rong Li, Yuhao Dong, Tianshuai Hu, Ao Liang, et al. 3EED: Ground everything everywhere in 3D. *arXiv preprint arXiv:2511.01755*, 2025.

[6] Ao Liang et al. Perspective-invariant 3D object detection. In *IEEE/CVF International Conference on Computer Vision*, pages 27725–27738, 2025. 4

[7] Xiang Xu et al. Beyond one shot, beyond one perspective: Cross-view and long-horizon distillation for better LiDAR representations. In *IEEE/CVF International Conference on Computer Vision*, pages 25506–25518, 2025.

[8] Xiang Xu, Lingdong Kong, Hui Shuai, Liang Pan, Ziwei Liu, and Qingshan Liu. LiMoE: Mixture of LiDAR representation learners from automotive scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27368–27379, 2025.

[9] Xuzhi Wang, Xinran Wu, Song Wang, et al. Monocular semantic scene completion via masked recurrent networks. In *IEEE/CVF International Conference on Computer Vision*, pages 24811–24822, 2025.

[10] Hengwei Bian et al. DynamicCity: Large-scale 4D occupancy generation from dynamic scenes. In *International Conference on Learning Representations*, 2025.

[11] Runnan Chen et al. CLIP2Scene: Towards label-efficient 3D scene understanding by CLIP. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7030, 2023.

[12] Youquan Liu et al. La La LiDAR: Large-scale layout generation from LiDAR data. *arXiv preprint arXiv:2508.03691*, 2025.

[13] Jiahao Sun, Chunmei Qing, Xiang Xu, et al. An empirical study of training state-of-the-art LiDAR segmentation models. *arXiv preprint arXiv:2405.14870*, 2024.

[14] Ao Liang, Youquan Liu, Yu Yang, et al. LiDARCrafter: Dynamic 4D world modeling from LiDAR sequences. *arXiv preprint arXiv:2508.03692*, 2025. 1

[15] Hui Ye, Rajshekhar Sunderraman, and Shihao Ji. Uav3d: A large-scale 3d perception benchmark for unmanned aerial vehicles, 2024. 1

[16] Marco Hutter, Christian Gehring, Dominic Jud, Andreas Lauber, C. Dario Bellicoso, Vassilios Tsounis, Jemin Hwangbo, Karen Bodie, Peter Fankhauser, Michael Bloesch, Remo Diethelm, Samuel Bachmann, Amir Melzer, and Mark Hoepflinger. Anymal - a highly mobile and dynamic quadrupedal robot. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 38–44, 2016.

[17] Kenneth Chaney, Fernando Cladera, Ziyun Wang, Anthony Bisulco, M Ani Hsieh, Christopher Korpela, Vijay Kumar, Camillo J Taylor, and Kostas Daniilidis. M3ED: Multi-robot, multi-sensor, multi-environment event dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4016–4023, 2023.

[18] Lingdong Kong, Dongyue Lu, Xiang Xu, Lai Xing Ng, Wei Tsang Ooi, and Benoit R. Cottereau. EventFly: Event camera perception from ground to the sky. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1472–1484, 2025.

[19] Rong Li et al. SeeGround: See and ground for zero-shot open-vocabulary 3D visual grounding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3707–3717, 2025.

[20] Runnan Chen et al. Towards label-free scene understanding by vision foundation models. In *Advances in Neural Information Processing Systems*, volume 36, pages 75896–75910, 2023.

[21] Xiaoshuai Hao, Guanqun Liu, Yuting Zhao, et al. MSC-Bench: Benchmarking and analyzing multi-sensor corruption for driving perception. *arXiv preprint arXiv:2501.01037*, 2025.

[22] 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.

[23] Youquan Liu et al. Segment any point cloud sequences by distilling vision foundation models. In *Advances in Neural Information Processing Systems*, volume 36, pages 37193–37229, 2023. 1

[24] Dekai Zhu, Yixuan Hu, Youquan Liu, et al. Spiral: Semantic-aware progressive LiDAR scene generation and understanding. *arXiv preprint arXiv:2505.22643*, 2025.

[25] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. RangeNet++: Fast and accurate LiDAR semantic segmentation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4213–4220, 2019. 1

[26] Xiaoshuai Hao et al. SafeMap: Robust HD map construction from incomplete observations. In *International Conference on Machine Learning*, pages 22091–22102. PMLR, 2025. 1

[27] Xiaoshuai Hao, Mengchuan Wei, Yifan Yang, et al. Is your HD map constructor reliable under sensor corruptions? In *Advances in Neural Information Processing Systems*, volume 37, pages 22441–22482, 2024.

[28] 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.

[29] 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.

[30] Ye Li, Lingdong Kong, Hanjiang Hu, Xiaohao Xu, and Xiaonan Huang. Is your LiDAR placement optimized for 3D scene understanding? In *Advances in Neural Information Processing Systems*, volume 37, pages 34980–35017, 2024. 4

[31] Xiang Xu et al. FRNet: Frustum-range networks for scalable LiDAR segmentation. *IEEE Transactions on Image Processing*, 34:2173–2186, 2025.

[32] 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. 1

[33] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation, 2017. 1

[34] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space, 2017. 1

[35] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud, 2019. 1

[36] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds, 2019. 1, 4

[37] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection, 2021. 1, 4

[38] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10), 2018. 1

[39] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking, 2021. 1, 3, 4

[40] Yan Wang, Xiangyu Chen, Yurong You, Li Erran, Bharath Hariharan, Mark Campbell, Kilian Q. Weinberger, and Wei-Lun Chao. Train in germany, test in the usa: Making 3d object detectors generalize, 2020. 1, 2

[41] Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. St3d: Self-training for unsupervised domain adaptation on 3d object detection, 2021. 1, 2

[42] Maximilian Jaritz, Tuan-Hung Vu, Raoul de Charette, Emilie Wirbel, and Patrick Pérez. xMUDA: Cross-modal unsupervised domain adaptation for 3D semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12605–12614, 2020. 2

[43] 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.

[44] Xidong Peng, Runnan Chen, Feng Qiao, et al. Learning to adapt SAM for segmenting cross-domain point clouds. In *European Conference on Computer Vision*, pages 54–71. Springer, 2024.

[45] Jingyi Xu, Weidong Yang, Lingdong Kong, Youquan Liu, Qingyuan Zhou, Rui Zhang, Zhijun Li, Wen-Ming Chen, and Ben Fei. Visual foundation models boost cross-modal unsupervised domain adaptation for 3d semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 26(11):20287–20301, 2025.

[46] Xuzhi Wang et al. NUC-Net: Non-uniform cylindrical partition network for efficient LiDAR semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(9):9090–9104, 2025.

[47] Youquan Liu et al. Multi-space alignments towards universal LiDAR segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14648–14661, 2024. 2

[48] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation, 2018. 2

[49] Qiangeng Xu, Yin Zhou, Weiyue Wang, Charles R. Qi, and Dragomir Anguelov. Spg: Unsupervised domain adaptation for 3d object detection via semantic point generation, 2021. 2

[50] 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, 2023. 2

[51] 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. 2

[52] Zhuoxiao Chen, Yadan Luo, Zheng Wang, Mahsa Baktashmotlagh, and Zi Huang. Revisiting domain-adaptive 3d object detection by reliable, diverse and class-balanced pseudo-labeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3714–3726, October 2023. 2

[53] Lingdong Kong, Xiang Xu, Youquan Liu, Jun Cen, Runnan Chen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. LargeAD: Large-scale cross-sensor data pretraining for autonomous driving. *arXiv preprint arXiv:2501.04005*, 2025. 2

[54] Zheng Li, Hao Chen, and Rui Zhao. Gpa-3d: Geometry-aware prototype alignment for unsupervised domain adaptive 3d object detection from point clouds. *arXiv preprint arXiv:2308.08140*, 2023. 2

[55] Yifan Wang, Zhiqiang Xu, and Bo Li. Ssda3d: Semi-supervised domain adaptation for 3d object detection from point clouds. *arXiv preprint arXiv:2212.02845*, 2022. 2

[56] Michael Wozniak, Anna Smith, and John Doe. Uada3d: Unsupervised adversarial domain adaptation for 3d object detection with sparse lidar and large domain gaps. *arXiv preprint arXiv:2403.17633*, 2024. 2

[57] S. S. Kotha, L. Zhang, and H. Wang. Cl3d: Unsupervised domain adaptation for cross-lidar 3d detection. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2023. 2

[58] Xiang Xu et al. 4D contrastive superflows are dense 3D representation learners. In *European Conference on Computer Vision*, pages 58–80, 2024. 2

[59] Dušan Malić, Christian Fruhwirth-Reisinger, Samuel Schulter, and Horst Possegger. Gblobs: Explicit local structure via gaussian blobs for improved cross-domain lidar-based 3d object detection, 2025. 2, 4

[60] Zhuoxiao Chen, Junjie Meng, Mahsa Baktashmotlagh, Yonggang Zhang, Zi Huang, and Yadan Luo. Mos: Model synergy for test-time adaptation on lidar-based 3d object detection, 2025. 2, 3

[61] Wentao Qu, Jing Wang, YongShun Gong, Xiaoshui Huang, and Liang Xiao. An end-to-end robust point cloud semantic

segmentation network with single-step conditional diffusion models, 2025. 2

[62] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection, 2021. 2, 4

[63] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 734–750. Springer, 2018. 3

[64] Ross Girshick. Fast r-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448. IEEE, 2015. 3

[65] Zhe Liu, Jinghua Hou, Xingyu Wang, Xiaoqing Ye, Jingdong Wang, Hengshuang Zhao, and Xiang Bai. LION: Linear group RNN for 3D object detection in point clouds. *Advances in Neural Information Processing Systems*, 2024. 4

[66] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 4

[67] Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models, 2023. 4

[68] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranthi Kiran GV, Xuzheng He, Haowen Hou, Jiaju Lin, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartlomiej Koptyra, Hayden Lau, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Bolun Wang, Johan S. Wind, Stanislaw Wozniak, Ruichong Zhang, Zhenyuan Zhang, Qihang Zhao, Peng Zhou, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. Rwkv: Reinventing rnns for the transformer era, 2023. 4

[69] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network, 2020. 4

[70] Lingdong Kong, Shaoyuan Xie, Zeying Gong, Ye Li, Meng Chu, Ao Liang, Yuhao Dong, Tianshuai Hu, Ronghe Qiu, Rong Li, Hanjiang Hu, Dongyue Lu, Wei Yin, Wenhao Ding, Linfeng Li, Hang Song, Wenwei Zhang, Yuexin Ma, Junwei Liang, Zhedong Zheng, Lai Xing Ng, Benoit R. Cottereau, Wei Tsang Ooi, Ziwei Liu, Zhanpeng Zhang, Weichao Qiu, Wei Zhang, Ji Ao, Jiangpeng Zheng, Siyu Wang, Guang Yang, Zihao Zhang, Yu Zhong, Enzhu Gao, Xinhan Zheng, Xueting Wang, Shouming Li, Yunkai Gao, Siming Lan, Mingfei Han, Xing Hu, Dusan Malic, Christian Fruhwirth-Reisinger, Alexander Prutsch, Wei Lin, Samuel Schulter, Horst Possegger, Linfeng Li, Jian Zhao, Zepeng Yang, Yuhang Song, Bojun Lin, Tianle Zhang, Yuchen Yuan, Chi Zhang, Xuelong Li, Youngseok Kim, Sihwan Hwang, Hyeonjun Jeong, Aodi Wu, Xubo Luo, Erjia Xiao, Lingfeng Zhang, Yingbo Tang, Hao Cheng, Renjing Xu, Wenbo Ding, Lei Zhou, Long Chen, Hangjun Ye, Xiaoshuai Hao, Shuangzhi Li, Junlong Shen, Xingyu Li, Hao Ruan, Jinliang Lin, Zhiming Luo, Yu Zang, Cheng Wang, Hanshi Wang, Xijie Gong, Yixiang Yang, Qianli Ma, Zhipeng Zhang, Wenxiang Shi, Jingmeng Zhou, Weijun Zeng, Kexin Xu, Yuchen Zhang, Haoxiang Fu, Ruibin Hu, Yanbiao Ma, Xiyan Feng, Wenbo Zhang, Lu Zhang, Yunzhi Zhuge, Huchuan Lu, You He, Seungjun Yu, Junsung Park, Youngsun Lim, Hyunjung Shim, Faduo Liang, Zihang Wang, Yiming Peng, Guanyu Zong, Xu Li, Binghao Wang, Hao Wei, Yongxin Ma, Yunke Shi, Shuaipeng Liu, Dong Kong, Yongchun Lin, Huitong Yang, Liang Lei, Haoang Li, Xinliang Zhang, Zhiyong Wang, Xiaofeng Wang, Yuxia Fu, Yadan Luo, Djamahl Etchegaray, Yang Li, Congfei Li, Yuxiang Sun, Wenkai Zhu, Wang Xu, Linru Li, Longjie Liao, Jun Yan, Benwu Wang, Xueliang Ren, Xiaoyu Yue, Jixian Zheng, Jinfeng Wu, Shurui Qin, Wei Cong, and Yao He. The RoboSense challenge: Sense anything, navigate anywhere, adapt across platforms. https://robosense2025.github.io, 2025. 3

[71] Lingdong Kong, Yaru Niu, Shaoyuan Xie, Hanjiang Hu, Lai Xing Ng, Benoit Cottereau, 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. 3

[72] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Lai Xing Ng, Benoit R. Cottereau, and Wei Tsang Ooi. RoboDepth: Robust out-of-distribution depth estimation under corruptions. In *Advances in Neural Information Processing Systems*, volume 36, pages 21298–21342, 2023. 3

[73] 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. 3

[74] 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. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3878–3894, 2025. 3

[75] Shaoyuan Xie, Lingdong Kong, Yuhao Dong, Chonghao

Sima, Wenwei Zhang, Qi Alfred Chen, Ziwei Liu, and Liang Pan. Are VLMs ready for autonomous driving? an empirical study from the reliability, data, and metric perspectives. In *IEEE/CVF International Conference on Computer Vision*, pages 6585–6597, 2025. 4

[76] Zeying Gong, Tianshuai Hu, Ronghe Qiu, and Junwei Liang. From cognition to precognition: A future-aware framework for social navigation. In *IEEE International Conference on Robotics and Automation*, pages 9122–9129, 2025.

[77] Meng Chu, Zhedong Zheng, Wei Ji, Tingyu Wang, and Tat-Seng Chua. Towards natural language-guided drones: GeoText-1652 benchmark with spatial relation matching. In *European Conference on Computer Vision*, pages 213–231, 2024. 4

[78] RoboSense Challenge 2025 Organizers. Robosense challenge 2025: Track 5 - cross-platform 3d object detection. https://robosense2025.github.io/track5, 2025. 4

[79] 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 (NeurIPS)*, volume 32, 2019. 4

[80] OpenPCDet Development Team. Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet, 2020. 4

[81] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR)*, 2015. 4

[82] Leslie N. Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, 11006:369–386, 2019. 4