SemiDet3D: A Baseline for Semi-supervised 3D Object Detection

Junbo Yin[1,2]*, Jin Fang[2]*, Dingfu Zhou[2], Shaoqing Xu[2,3],
Jianbing Shen[1], Liangjun Zhang[2] and Wenguan Wang[4]

[1] Beijing Institute of Technology [2] Robotics and Autonomous Driving Laboratory, Baidu Research

[3]Beihang University [4]ETH Zürich

{yinjunbocn,wenguanwang.ai}@gmail.com fangjin@baidu.com

Abstract

In this technical report, we address the semi-supervised point cloud-based 3D object detection in autonomous driving scenarios. In particular, a simple yet effective semi-supervised framework "SemiDet3D" is proposed to address this challenge. SemiDet3D consists of two crucial components, which are self-supervised pre-training and semi-supervised fine-tuning. During the pre-training, we construct a new pretext task that focuses on point-wise contrastive learning with the unlabeled point cloud samples. This provides a useful representation for the subsequent detection tasks. In the semi-supervised fine-tuning, we aim to utilize both the labeled and unlabeled point cloud following the idea of noisy student model. Elaborate pseudo labels produced by the teacher model are used to iteratively train the student model. Further equipped with technique improvements like Weighted Box Fusion, Test Time Augmentation and Model Ensemble, our final model achieves 3rd on the test set of ONCE benchmark.

1. Introduction

With the rapid development of the LiDAR sensors and their wide application in the field of robotic and autonomous driving, point cloud-based scene understanding such as 3D object detection [6, 15, 27, 28, 31, 32] has received great attention in recent years. With the aid of deep neural networks, impressive performance has been achieved on different public datasets [1, 7, 10, 20]. Some representative 3D object detectors are VoxelNet [33], PointRCNN [17], PointPillars [11], PV-RCNN [15] and CenterPoint [28] etc. However, most existing 3D object detectors typically rely on a large amount of manually labeled data, but collecting labels in point clouds is expensive and time-consuming [5]. By contrast, it is easy for self-driving vehicles to collect large-scale unlabeled data. How to effectively leverage these unlabeled data remains an open problem in the community.

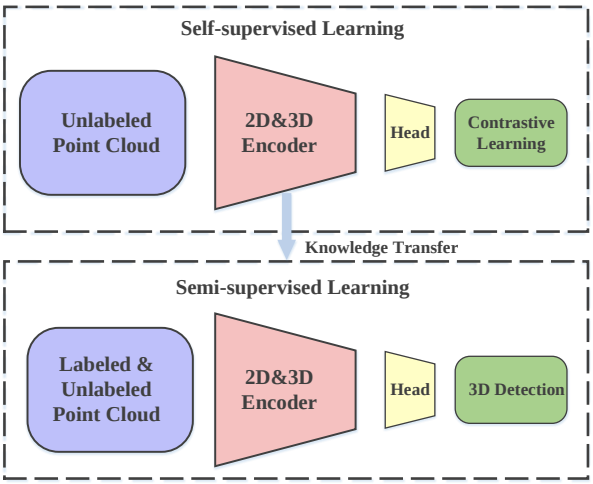

Figure 1: Overview of our framework. We adopt a typical two-stage framework to address the semi-supervised point cloud-based 3D object detection.

Self-supervised and semi-supervised learning are two typical ways of utilizing the unlabeled samples for training models. Conceptually, self-supervised learning is one form of unsupervised learning, since no manual labels are required. While semi-supervised learning approaches need a small amount of labeled data together with a large amount of unlabeled data during training. In general, semi-supervised learning falls between unsupervised learning and supervised learning belonging to a special instance of weak supervision. Recently, these techniques [2, 8, 9, 21] are being widely studied in 2D domain and have achieved impressive success for image classification task. For the self-training in 2D task, it first defines a *pretext task*, and then gets the labels from the unlabeled data itself for free to train the pretext task. After the convergence of pretext task training, a pre-trained model is obtained that could be used in downstream tasks by supervised fine-tuning.

*Equal contribution.

Recently, the unsupervised, semi-supervised and weakly-supervised techniques have been applied on point cloud for 3D tasks such as classification, semantic segmentation [25] and object detection [26]. PointContrast [25] as one pioneer unsupervised learning framework pre-trains the backbone architecture (e.g., a sparse residual U-Net) on the indoor ScanNet [3] for learning the point cloud representation with the contrastive losses. Then the pre-trained backbone has been finetuned on various datasets for different tasks including both semantic segmentation and object detection. Besides unsupervised learning, semi-supervised learning is another way of utilizing the unlabeled data. SESS [30] and 3DIoUMatch [23] are two pioneers of this domain, which reveal the ability to integrate teacher and student models in 3D object detection. Meng *et al*. [13] further propose a weakly supervised method that requires only a few weakly annotated scenes and precisely labeled object instances, which greatly saves annotation efforts.

ONCE Dataset [12] is the officially provided dataset for this challenge which includes 1 million LiDAR samples and 7 million scene images under different weather conditions. However, only about 15k frames have provided the 3D object detection annotation where the training, validation, and testing are around 5k, 3k, and 8k frames respectively. To fully utilize both the annotations and a large number of unlabeled samples, we first employ the self-learning technique to learn a pre-trained model and then use the limited labeled data to generate pseudo-label for unlabeled samples iteratively in a semi-supervised manner, as shown in Fig. 1.

2. Proposed Approach

In this section, we first give a brief introduction to the baseline 3D object detectors in §2.1. Then, in §2.2, we describe our self-supervised strategy for pre-training a point cloud encoder with the large-scale unlabeled data. Next, we introduce a semi-supervised learning method to further improve the detectors with both labeled and unlabeled data in §2.3. Finally, some post-processing techniques are presented in §2.4.

2.1. Baseline 3D Object Detector

We revisit the architectural details of the 3D object detectors that we used in this challenge. To be specific, we adopt an anchor-based model PV-RCNN [15] and an anchor-free model CenterPoint [28] to handle different difficulties.

PV-RCNN. PV-RCNN is a two-stage detector that integrates the advantages of both point-wise and voxel-wise feature representation. In the first stage, it aims to generate dense voxel-wise proposals with a 3D Sparse Convolution Network. Meanwhile, a Voxel Set Abstraction layer is also leveraged to aggregate the point-wise representation and improve the fine-grained localization ability. Specifically, a fixed number of keypoints, *e.g*., 4096 points in ONCE dataset, are first sampled from the input point cloud with Furthest-Point-Sampling (FPS). Then each keypoint abstracts its surrounding voxel features in different feature levels. The 3D Sparse Convolution Network is optimized by a region proposal loss, while the keypoints are optimized by point-wise segmentation loss. In the second stage, the 3D proposals produced by the first stage aggregate the features of keypoints for accurate and robust proposal refinement via a keypoints set abstraction layer. In particular, each grid in a proposal (denoted by $6 \times 6 \times 6$ grids) abstracts the features of neighboring keypoints. Then, each proposal is vectorized by a 256 dimensions feature for iou-aware confidence estimation and localization residual regression. PV-RCNN requires predefined 3D anchors for generating dense proposals, thus it has great capabilities in detecting large objects such as car, bus and truck. However, it is incapable of capturing accurate direction information for small objects like pedestrian and cyclist due to the limitation of anchors. Therefore, we resort to an anchor-free model CenterPoint to further address these classes.

CenterPoint. CenterPoint [28] proposes to represent and detect objects with rotationally invariant points, which is more flexible to cover small objects in the bird-eye view. It encodes the point cloud with a 3D Sparse Convolution Network, as well as devises an elaborate anchor-free head for 3D object detection. Specifically, it involves a heatmap head and a regression head. The former head is used to predict the centers of objects with the help of rendered Gaussian kernels, while the latter head aims to estimate other properties of objects, including sub-voxel offset, height, sizes and rotation. In our implementation, multi-head strategy is used to handle different classes, where each heatmap head only needs to focus on one class. This simplifies the learning process and improves performance on classes with fewer labels, *e.g*., pedestrian in ONCE dataset.

2.2. Self-supervised Pre-training

Self-supervised learning has shown promising performance in 2D images recognition domains. It makes better use of the large-scale unlabeled data to provide a pre-trained model that is useful in certain downstream tasks. To achieve this, it first defines a pretext task and then generates the supervisory signals directly from the input data for free. In this challenge, ONCE dataset provides 1 million unlabeled point clouds with high diversity, which makes it possible to learn a 'universal' representation for the point cloud encoders of common 3D object detectors.

To effectively leverage the large-scale unlabeled point cloud, we propose to perform point-wise contrastive learning for self-supervised pre-training, which is inspired from PointContrast [25]. Specifically, given an input point cloud, we first apply two different augmentations, which include rotation, translation, scaling and random sampling. This

results in two different views of the input point cloud. Then we sample keypoints from the original point cloud with FPS and find correspondence in the two augmented views. Compared with PointContrast that computes correspondence with nearest neighbor search, our correspondence is obtained directly from the recorded point index, which is more accurate and efficient. After that, we acquire feature representation for each keypoint by bilinear interpolation on the bird-eye view feature map. The point-wise features are then embedded into latent space and info InfoNCE [14] loss is applied on each positive and negative point pair to optimize the network. After the training process, the resultant encoder can be applied to the downstream detector for semi-supervised fine-tuning.

2.3. Semi-supervised Self-training

To further take advantage of the large-scale unlabeled data, we adopt a semi-supervised strategy modified from the noisy student model [24]. Noisy student is a simple self-training method that achieves promising performance on 2D image classification. We adapt noisy student to 3D object detection task with several non-trivial modifications.

In particular, we first train a teacher detection model with the labeled data and the self-supervised pre-trained encoder. Then, we use the teacher model to produce pseudo labels on the unlabeled data which serve as the targets of the student model. Here, test time augmentation and epoch-wise model ensemble strategies (See §2.4) are used to generate pseudo labels with high quality. Besides, we also carefully filter the pseudo labels by confidence score after Non-maximum Suppression (NMS). Formally, given a batch of labeled and unlabeled data, the teacher model only infers pseudo label on the unlabeled data, while the student model is supervised by both the pseudo labels from unlabeled data and real labels from the labeled data. After several epochs, the parameters of the teacher model are updated by the student model, and the self-training process is iteratively performed until convergence. Thanks to the self-supervised pre-training and semi-supervised fine-tuning, the detectors could learn knowledge from the large-scale unlabeled data, and thus improve the detection performance. We named our method SemiDet3D, which provides a simple yet effective baseline for semi-supervised 3D object detection in an autonomous driving setting.

2.4. Further Improvements

In this section, we elaborate several post-processing techniques that help to further improve the detection performance, which includes Weighted Box Fusion (WBF), Test Time Augmentation (TTA) and Model Ensemble.

Weighted Box Fusion. Weighted Box Fusion [18] is originally proposed for 2D object detection, which is an effective way to aggregate bounding boxes from different sources. Concretely, given boxes from multiple models, WBF first performs clustering on these boxes according to the intersection-over-union (IoU). Then, it recalculates the box coordinates and confidence score by using all the boxes in a cluster and produces an updated box. The centers and sizes of the updated box are obtained by weighted sums of each box in the cluster, where the weights are the confidence score of the corresponding box. As for the rotation of the updated box, we use the one with the highest confidence score. The final confidence score of the updated box is the average confidence of all the boxes in the cluster.

Test Time Augmentation. During the inference process, we perform data augmentation on each input point cloud which is similar to the training process. This enforces the network to give more robust predictions. Specifically, double flip (*e.g.*, along x, y or xy axes) and multiple rotations (*e.g.*, $\pm 22.5^\circ$, 157.5° and 180°) are applied on the input point cloud, and the output boxes and transformed reversely and aggregated by WBF. We find this technique effectively improves the orientation estimation, which is of crucial importance under the ONCE metrics.

Model Ensemble. We further investigate model ensemble to achieve better results, which consists of epoch-level ensemble and detector-level ensemble. For the former one, we perform inference with the models from the last 10 epochs and aggregate the 10 groups of detections with WBF. For the latter one, we just combine the detection results from PV-RCNN and CenterPoint models following WBF and produce the final detection results.

3. Experimental Results

In this section, we first introduce the dataset and metrics we used, and then we evaluate the effectiveness of our proposed method.

3.1. ONCE Dataset

Dataset. ONCE Dataset[1] [12] is a large scale autonomous driving dataset with 1 million LiDAR samples and 7 million scene images under different weather conditions, while only 15,000 frames are with annotation, in which 5k frames are used for training, 3k frames for validation and 8k frames for testing. All the scenes are provided in 2 FPS and annotated in 1 FPS. The objects are categorized into 5 classes, i.e. "Car", "Bus", "Truck", "Pedestrian" and "Cyclist".

Metric. The evaluation metric for ONCE dataset benchmark is mean AP (average precision) [4] over the three classes (the "Car", "Bus" and "Truck" classes are merged into "Vehicle" class during evaluation). For each class, ONCE benchmark follows the 3D IoU-based AP used in KITTI [7], but furthermore, the orientation of the object

[1] https://once-for-auto-driving.github.io

Method	Vehicle				Pedestrian				Cyclist				mAP
	overall	0-30m	30-50m	50m-inf	overall	0-30m	30-50m	50m-inf	overall	0-30m	30-50m	50m-inf	
Baselines													
PV-RCNN [16]	77.77	89.39	72.55	58.64	23.50	25.61	22.84	17.27	59.37	71.66	52.58	36.17	53.55
CenterPoint [29]	66.79	80.10	59.55	43.39	49.90	56.24	42.61	26.27	63.45	74.28	57.94	41.48	60.05
Improved Baselines													
PV-RCNN [16]	78.03	89.57	72.55	60.03	29.08	32.45	27.50	18.16	62.96	74.64	56.94	39.94	56.69
CenterPoint [29]	77.00	87.77	69.56	56.88	46.16	56.25	38.68	21.43	64.19	76.27	58.29	39.28	62.45
Improved Baselines + Painting													
CenterPoint [29]	73.02	85.15	67.05	52.55	50.71	58.31	43.76	24.12	64.11	75.58	59.07	40.74	62.61
Improved Baselines + SemiDet3D													
PV-RCNN [16]	81.09	89.35	77.52	62.73	41.55	50.75	35.77	21.01	67.57	79.28	62.72	43.40	63.40
CenterPoint [29]	76.93	85.35	71.89	59.18	57.78	65.97	49.21	28.75	68.70	80.26	64.18	43.70	67.80
Improved Baselines + SemiDet3D + EpochEnsemble													
PV-RCNN [16]	81.04	89.32	77.48	63.88	42.37	51.62	36.83	22.87	68.47	79.36	63.51	43.86	63.96
CenterPoint [29]	78.46	86.87	73.78	62.22	62.69	71.66	53.60	33.96	70.77	81.25	64.94	51.00	70.64
Improved Baselines + SemiDet3D + EpochEnsemble + TTA													
PV-RCNN [16]	83.02	91.10	79.94	67.35	55.70	64.52	50.40	32.54	73.35	83.35	68.78	53.39	70.69
CenterPoint [29]	80.98	87.29	77.69	67.41	73.87	80.70	67.10	49.87	77.77	85.90	73.14	60.48	77.54
Improved Baselines + SemiDet3D + EpochEnsemble + TTA + WBF													
Ensemble Model	83.02	91.10	79.94	67.35	73.87	80.70	67.10	49.87	77.77	85.90	73.14	60.48	78.22

Table 1: Evaluation results of ablation studies on validation split.

Classes	**AP@50** (%)			
	overall	0-30m	30-50m	50m-inf
Vehicle	83.02	91.10	79.94	67.35
Pedestrian	73.87	80.70	67.10	49.87
Cyclist	77.77	85.90	73.14	60.48
mAP	78.22	85.90	73.39	59.23

Table 2: The detailed evaluation results with proposed *SemiDet3D* framework on validation split.

Usernames	**mAP** (%)	**AP** (%)		
		Vehicle	Pedestrian	Cyclist
basedet	85.12	88.38	84.45	82.52
zzhxyw	82.89	85.45	83.54	79.69
FangJin	79.55	83.96	78.22	76.48
sslad-hiker	79.09	82.99	76.08	78.20
hailanyi	78.65	83.02	74.81	78.11

Table 3: The top-5 evaluation results on the official testing server, in which our result is highlighted in blue.

which cannot fall into $-90°$ to $90°$ of the ground truth orientation will be considered as false positive.

3.2. Implementation Details

All the supervised models are trained with Adam optimizer for 80 epochs, while the learning rate is 0.003, weight decay is 0.01 and momentum is 0.9. We follow [12] to employ the data augmentation strategies.

3.3. Ablation Study

Due to the limitation of the submission number on the official testing server, we evaluate the effectiveness of each module of *SemiDet3D* on the validation split. The detailed performance can be found in Tab. 1. We choose PV-RCNN and CenterPoint as our baseline detectors, due to their outstanding performance in detecting the vehicle and pedestrian respectively. We make some modifications for the two baselines, which bring 3.14% and 2.40% improvement, respectively. Refer to [12], with PointPainting [22], the performance drops about 1.46% in testing split, we improve the PointPainting framework and achieve better result with CenterPoint in validation split, but we finally remove the painting module due to the heavy time consuming but limited improvement. Our proposed *SemiDet3D* framework brings about 6.71% and 5.35% improvement for the two detectors. With the Epoch Ensemble, Test Time Augmentation (TTA) and Weighted Box Fusion (WBF) [19] strategies, the final model could achieve 78.22% mAP.

3.4. Evaluation Results on Validation Split

The evaluation result with proposed *SemiDet3D* framework on validation split is shown in Tab. 2, where we can

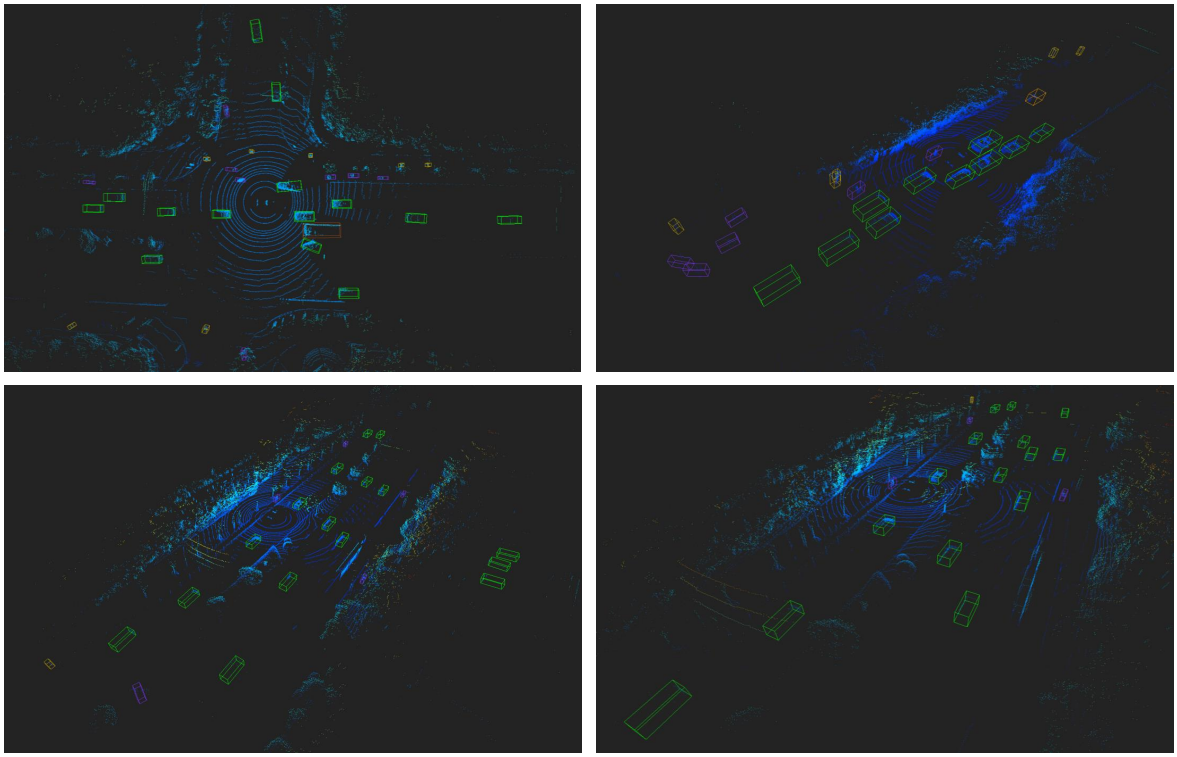

Figure 2: Visualization results predicted by our proposed *SemiDet3D* framework.

find the detailed performance with different distance ranges over the three classes.

3.5. Evaluation Results on Testing Server

We submit our result with *SemiDet3D* framework to the official testing server to attend the *ICCV 2021 Workshop SSLAD Track 2 - 3D Object Detection Challenge*, the top-5 results can be found in Tab. 3. Our result is highlighted with blue, which finally ranks third. Noted that our public result is from an early version, the final performance of the whole *SemiDet3D* framework still needs a few days to run out.

3.6. Visualization Results

In Fig. 2, we show some visualization results of our proposed *SemiDet3D* framework.

4. Conclusions

In this report, we proposed "SemiDet3D" framework which combines both self-supervised learning and semi-supervised learning to improve the performance of 3D object detection with the help of the large-scale unlabeled point cloud provided by ONCE Benchmark. Two state-of-the-art detectors, PV-RCNN and CenterPoint, were employed in the final model. Further integrating with the post-processing techniques and model ensemble, we finally get 79.55% mAP in the testing split of ONCE Benchmark and 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. *arXiv preprint arXiv:1903.11027*, 2019. 1

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. 2020. 1

[3] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 2

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 3

[5] Jin Fang, Dingfu Zhou, Feilong Yan, Tongtong Zhao, Feihu Zhang, Yu Ma, Liang Wang, and Ruigang Yang. Augmented lidar simulator for autonomous driving. *IEEE Robotics and Automation Letters*, 5(2):1931–1938, 2020. 1

[6] Jin Fang, Xinxin Zuo, Dingfu Zhou, Shengze Jin, Sen Wang, and Liangjun Zhang. Lidar-aug: A general rendering-based augmentation framework for 3d object detection. In *CVPR*, pages 4710–4720, 2021. 1

[7] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012. 1, 3

[8] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words. In *CVPR*, 2020. 1

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1

[10] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its application. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2702–2719, 2019. 1

[11] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. *arXiv preprint arXiv:1812.05784*, 2018. 1

[12] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, Jie Yu, Chunjing Xu, et al. One million scenes for autonomous driving: Once dataset. 2021. 2, 3, 4

[13] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Yunde Jia, and Luc Van Gool. Towards a weakly supervised framework for 3d point cloud object detection and annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[14] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3

[15] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10529–10538, 2020. 1, 2

[16] 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. In *CVPR*, pages 10529–10538, 2020. 4

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

[18] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: ensembling boxes for object detection models. *arXiv e-prints*, 2019. 3

[19] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, pages 1–6, 2021. 4

[20] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020. 1

[21] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020. 1

[22] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4604–4612, 2020. 4

[23] He Wang, Yezhen Cong, Or Litany, Yue Gao, and Leonidas J Guibas. 3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection. In *CVPR*, pages 14615–14624, 2021. 2

[24] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, 2020. 3

[25] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *ECCV*, pages 574–591, 2020. 2

[26] Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. St3d: Self-training for unsupervised domain adaptation on 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10368–10378, 2021. 2

[27] Junbo Yin, Jianbing Shen, Chenye Guan, Dingfu Zhou, and Ruigang Yang. Lidar-based online 3d video object detection with graph-based message passing and spatiotemporal transformer attention. In *CVPR*, pages 11495–11504, 2020. 1

[28] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11784–11793, 2021. 1, 2

[29] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *CVPR*, pages 11784–11793, 2021. 4

[30] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Sess: Self-ensembling semi-supervised 3d object detection. In *CVPR*, pages 11079–11087, 2020. 2

[31] Dingfu Zhou, Jin Fang, Xibin Song, Chenye Guan, Junbo Yin, Yuchao Dai, and Ruigang Yang. Iou loss for 2d/3d object detection. In *2019 International Conference on 3D Vision (3DV)*. IEEE, 2019. 1

[32] Dingfu Zhou, Jin Fang, Xibin Song, Liu Liu, Junbo Yin, Yuchao Dai, Hongdong Li, and Ruigang Yang. Joint 3d instance segmentation and object detection for autonomous driving. In *CVPR*, pages 1839–1849, 2020. 1

[33] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018. 1