Learning with Noisy Data for Semi-Supervised 3D Object Detection

Zehui Chen[1] Zhenyu Li[2] Shuo Wang[1] Dengpan Fu[3] Feng Zhao[1*]
[1]University of Science and Technology of China
[2]Harbin Institute of Technology [3] NIO
{lovesnow,wangshuo}@mail.ustc.edu.cn zhenyuli17@hit.edu.cn
dengpan.fu@nio.com fzhao956@ustc.edu.cn

Abstract

Pseudo-Labeling (PL) is a critical approach in semi-supervised 3D object detection (SSOD). In PL, delicately selected pseudo-labels, generated by the teacher model, are provided for the student model to supervise the semi-supervised detection framework. However, such a paradigm may introduce misclassified labels or loose localized box predictions, resulting in a sub-optimal solution of detection performance. In this paper, we take PL from a noisy learning perspective: instead of directly applying vanilla pseudo-labels, we design a noise-resistant instance supervision module for better generalization. Specifically, we soften the classification targets by considering both the quality of pseudo labels and the network learning ability, and convert the regression task into a probabilistic modeling problem. Besides, considering that self-supervised learning works in the absence of labels, we incorporate dense pixel-wise feature consistency constraints to eliminate the negative impact of noisy labels. To this end, we propose NoiseDet, a simple yet effective framework for semi-supervised 3D object detection. Extensive experiments on competitive ONCE and Waymo benchmarks demonstrate that our method outperforms current semi-supervised approaches by a large margin. Notably, our NoiseDet achieves state-of-the-art performance under various dataset scales on ONCE dataset. For example, NoiseDet improves its NoiseyStudent baseline from 55.5 mAP to 58.0 mAP, and further reaches 60.2 mAP with enhanced pseudo-label generation. Code will be available at https://github.com/zehuichen123/NoiseDet.

1. Introduction

3D object detection, aiming at detecting instances in the 3D space, is of great importance in various applications. Thanks to the large amount of labeled data, it has achieved great success in the past few years [37, 29, 9, 56, 10, 11]. However, labeling a large-scale dataset is extremely time-consuming and labor-intensive [40]. This issue gets much more severe in the 3D object detection task, where 7 degree-of-freedom parameters need to be determined as well as the categorical labels. Therefore, how to further boost the supervised detectors with limited labeled datasets remains a practical and urgent problem.

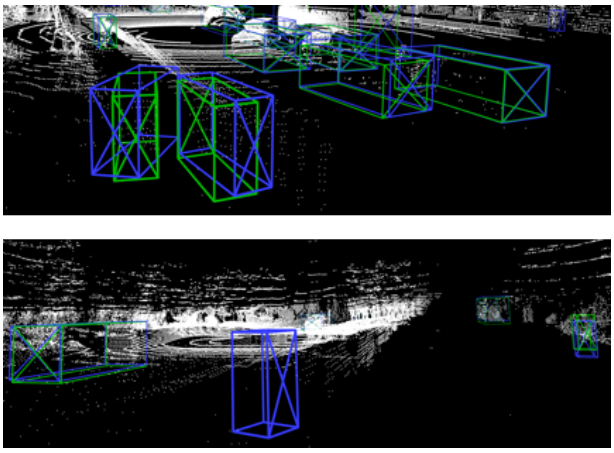

Figure 1. The noisy predictions generated by a well-trained 3D object detector. Despite carefully tuning the model and filtering low-confidence score predictions, there are still low-quality bounding boxes left. Directly enforcing the detector to learn from such noisy data on an unlabeled dataset can mislead the model convergence or even deteriorate its performance. In the top figure, we demonstrate an inaccurate heading regression case, and the bottom case shows a false positive prediction. We distinguish the ground truth and model predictions with green and blue, respectively.

Semi-supervised object detection (SSOD) demonstrates great prospects recently due to its simplicity and weak dependence on costly annotations [46, 20, 39]. By leveraging easily accessible unlabeled data, SSOD approaches enhance the model performance with little human effort [47, 43, 51, 25]. The strategies of current SSOD can be

*Corresponding Author

categorized into two main streamlines: Mean-Teacher [41] and Pseudo-Labeling [24]. Mean-Teacher adopts a teacher-student paradigm, where the teacher is initialized from the student model via Exponential Moving Average (EMA) to produce supervised signals on the unlabeled data in an end-to-end training fashion. Inspired by this, more advanced ways, including SESS [55], NoisyStudent [45], are built upon the teacher-student paradigm and achieve significant improvements. Despite the superior performance and elegant pipeline, they are not model-agnostic (*i.e.* the teacher and student should be the same model) and miss the chance to pursing high-quality pseudo labels, such as off-board label generation [35]. Another line of work is pseudo labeling [53, 36]: the model is first trained on labeled data and then generated pseudo labels on the unlabeled data for later training. Different from MT, it can be easily applied to any detector without adapting the framework to the teacher-student paradigm. These multi-stage approaches obtain reasonably good results, however, the final performance is often impeded by the quality of pseudo labels. To address this issue, 3DIoUMatch [42] proposes to estimate the localization accuracy predicted by the network and utilizes it as a metric to guide the label generation. ProficientTeacher [51] introduces a clustering-based box voting module to filter low-quality predictions without introducing any hand-crafted rule. Although the accuracy of pseudo labels gets greatly improved, it may still fail in certain cases, leading to unstable model convergence. Harsh confidence filtering is proven to be worked well [46], however, we observe that it is still hard to remove all *misleading* instances for generated labels, which can distract the model convergence, shown in Figure 1.

In this paper, we acknowledge the existence of noise in the generated labels and take PL as a noisy learning task: instead of directly applying vanilla pseudo-labels, we design a noise-resistant instance supervision module for better model generalization. Specifically, for the classification branch, we leverage the confidence predictions from the teacher and the student model learning ability to construct soft categorical targets. Different from previous approaches [47, 28], which simply apply them for loss reweighting, we enforce the network to directly learns from such continual supervision. In this way, our strategy can adaptively learn more knowledge from the teacher's predictions and at the same time develop tolerance to low confidence score labels. As for the regression branch, we find that solely relying on classification scores or IoU predictions is not enough to reflect the learning quality of each attribute of the 3D box. To mitigate this problem, we convert the original smooth L1 regression into a probabilistic modeling task: by enabling the network to output two variables to represent the Gaussian distribution, we maximize the negative log-likelihood of the targets in the distribution. Apart from the label supervision on the unlabeled data, we also introduce a novel dense pixel-wise feature self-supervised consistency constraints to further eliminate the negative impact of noisy data. To this end, we propose *NoiseDet*, a simple semi-supervised 3D object detection framework to address the noise learning problem from the unlabeled data. Our NoiseDet is modal-agnostic and can be easily applied to any 3D object detector with little modification.

We summarize our key contributions as follows:

- Instead of directly improving the quality of pseudo-labels, we formulate SSOD as a noisy learning problem, where we acknowledge the existence of the noise in pseudo-labels and soften the network supervision for model generalization.
- Based on this thought, we propose NoiseDet, a simple yet effective SSOD framework to deal with the noisy data through two core modules: noise-resistant instance supervision and dense pixel-wise feature consistency constraint.
- With extensive experiments, we validate the effectiveness of our approach on two competitive benchmarks. Notably, it achieves state-of-the-art performance on the challenging ONCE dataset.

2. Related Work

2.1. LiDAR-based 3D Object Detection

3D object detection aims to detect instances in the 3D space, which is of great importance in various applications, such as autonomous driving, navigation robot, and augmented reality. Current LiDAR detectors [52, 59, 49] mainly voxelize the point cloud into a BEV or voxel representations and adopt traditional 2D convolutions to predict bounding boxes. To improve the inference speed, some works [15, 30] attempt to detect objects directly through the range view. [58] combines both BEV and range-view features together to achieve better performance. VoxelNet [59] adopts PointNet [34] to extract local features through raw point clouds and fill them into the predefined voxel space. PointPillar [22] directly processes points inside a pillar to naturally formulate BEV feature representations, establishing an ultra-fast baseline detector. Inspired by CenterNet [14], CenterPoint [52] introduces a center-based label assignment strategy in 3D object detection, achieving competitive detection accuracy among various approaches. In addition to dense one-stage detectors, many works [8, 38] apply an R-CNN-style two-stage detection paradigm. Point R-CNN [38] propose 3D RoIAlign to aggregate regional features based on the proposals and then refines the detections. PV-RCNN [37] and Voxel R-CNN [13] construct two-parallel branches to extract both point-level and voxel-

level features to enjoy the best of each. However, independently localizing moving instances can introduce misalignment noise, 3D-MAN [50] and MPPNet [7] leverage multi-frame information to further enhance predictions with temporal knowledge.

2.2. Semi-Supervised Learning (SSL)

Semi-supervised learning is an important task in leveraging easy-to-access unlabeled data to improve the supervised model. Most current SSL methods [53, 17, 5, 19] involve adding additional supervision on unlabeled data to regularize the learning of the model. Among them, pseudo-labeling is a popular pipeline [4, 24, 18], where unlabeled data is firstly labeled with a supervised model, and then acts in a common training paradigm. In order to guarantee the quality of the generated labels, [24] often filter them with a hard threshold based on the classification score. In addition to hard pseudo-labels, NoisyStudent [45] explores soft supervision to avoid ambiguity problems. Besides, it injects different augmentations to the student and teacher models, to encourage consistency regularization. In light of this, plenty of approaches [44, 2, 26] enforce the model to predict similar results when applying various input permutations. Such strategies are also verified on 2D object detection, where [32] borrows the idea from FixMatch to achieve promising performance. MUM [21] introduces Mix/UnMix augmentation, enforcing students to reconstruct unmixed features for the mixed input images.

2.3. Semi-Supervised 3D Object Detection

Semi-supervised learning (SSL) has been rapidly developed in both classification and object detection domains and obtains promising results in recent years. There are two main streams in SSOD: consistency learning and pseudo labeling. Consistency-based works [41, 25] apply data augmentations/perturbations to the input, which forms natural regularization for the network predictions. Such a consistency-learning target enforces the model to acquire valid information from the unlabeled data, therefore improving the performance. SESS [55] is the first work to attempt such a paradigm on 3D object detection, where the classification and regression predictions are matched through L2 distance and supervised with the similarity loss on the teacher and student. Inspired by Mean Teacher [41], it also adopts the exponential moving average (EMA) technique to further boost the performance. Apart from the consistency-based approaches, pseudo-labeling is another solution [46, 42]. Most PL works pay attention to the quality enhancement of pseudo labels. 3DIoUMatch [42] proposes to learn the IoU of the network predictions and utilize it to adaptively filter the low-quality pseudo labels. In order to remove the duplicate score threshold search, Proficient-Teacher [51] introduces a novel clustering-based box voting module, replacing the hand-crafted NMS process. Different from the previous approaches, our method views pseudo-labeling as a noisy learning problem, therefore delivering more generalization against noisy pseudo-data.

3. NoiseDet

It is acknowledged that predictions generated by the teacher model will inevitably encounter false positive (FP) or false negative (FN) problems, despite many efforts that have been devoted to overcoming these issues [51, 46]. Such noises can distract the convergence of the model during the semi-supervised training process, or even deteriorate the final performance.

The main intuition of NoiseDet is to improve the model generalization ability against noisy pseudo labels, which are generated by the baseline model trained on limited labeled data. We start with a vanilla two-stage pseudo-labeling pipeline: assuming that we have access to a (small) set of labeled set $L = \{x_i^l, y_i^l\}_{i=1}^{N_l}$, and a set of unlabeled set $U = \{x_i^u\}_{i=1}^{N_u}$, where N_l and N_u are the numbers of labeled and unlabeled datasets. A teacher model is firstly trained on the labeled set L, and then infer on unlabeled set U to get pseudo labels $\{y_i^u\}_{i=1}^{N_u}$ and formulate the pseudo-labeled set $U_p = \{x_i^u, y_i^u\}$. After that, we initialize the student model from the pretrained teacher and uniformly sample data from both L and U_p set to construct each batch for second-stage training. Different from previous works, which mainly focus on improving the quality of pseudo-labels, NoiseDet directly learns knowledge from noise labels $\{y_i^U\}_{i=1}^{N_u}$.

3.1. Noise-Resistant Instance Supervision

In contrast to standard fully supervised object detection, where labels are carefully human-annotated and verified, the supervisory signals on the unlabeled set are generated by the vanilla teacher model. It may contain unexpected noisy labels, leading to model performance degeneration. Therefore, we propose to transform such hard, deterministic instance supervision into the noise-resistant one to develop the model tolerance to the noisy labels, instead of directly eliminating them.

Recent works [16, 3] suggests that the confidence score c can be viewed as the noisy level of the network predictions, *i.e,* the model is more prone to yield a high score for easy-to-detect samples and low score for not-sure samples. Therefore, an intuitive idea is to filter the generated pseudo-labels with a pre-defined score threshold $U_p = \{(x_i^u, c_i^u) | c_i^u \geq c_{thres}\}$ to eliminate the noises for the unlabeled dataset. Though simple, such a strategy does not provide a clear divergence for the rest labels. For instance, the quality of a box with $c = 0.9$ can be different from the quality of the box with $c = 0.5$. However, they are treated equally with 1/0 labels in the classification branch. To fully exploit a reliable confidence score derived from the

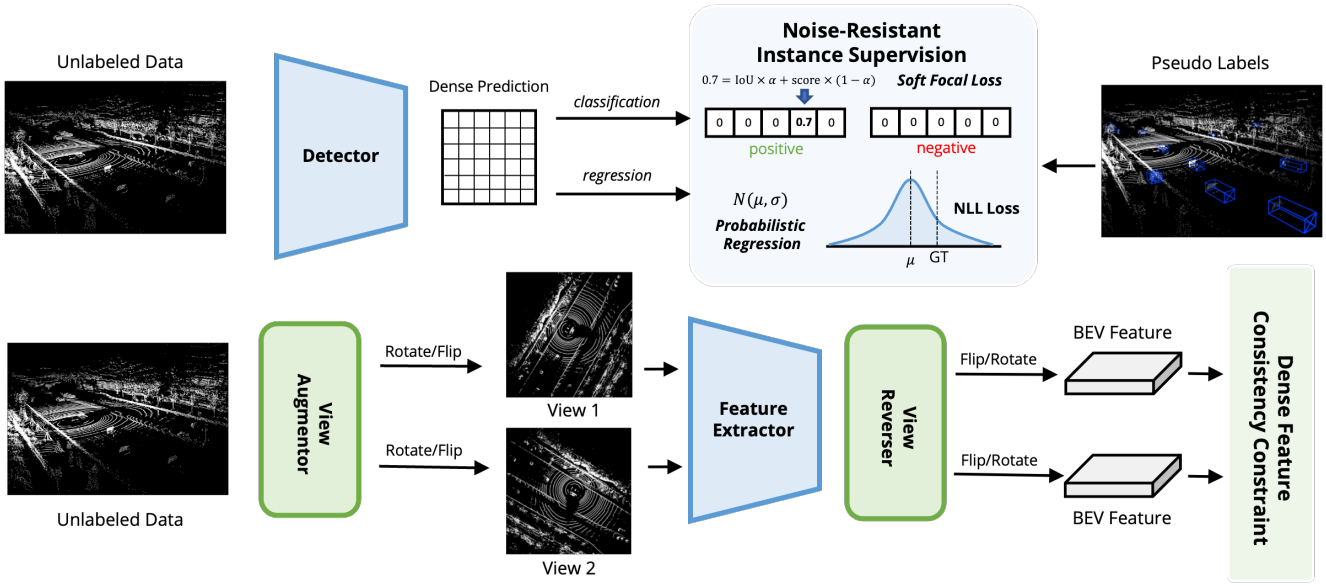

Figure 2. The overall architecture of the proposed NoiseDet. It consists of two main components: noise-resistant instance supervision and dense feature consistency constraint. It follows the pseudo-labeling practice with a two-stage training paradigm: (i) pseudo labels are pre-generated with a well-trained teacher model on the unlabeled dataset, and (ii) the student model is trained on both labeled (omitted in the figure) and unlabeled LiDAR points in a semi-supervised manner.

teacher model, we leverage the c as an indicator to measure the quality of the pseudo label. Specifically, instead of directly assigning each anchor with a deterministic binary label, we soften the categorical label into a value ranging from 0 to 1 given the confidence score c as well as the IoU τ between the student predictions and its matched pseudo labels. Different from the one in GFL [27] where the categorical labels are assigned based on the IoU between the network prediction and the ground-truth boxes, we view it as the combination of the quality of the GT boxes itself and the learning ability of the student. Since the standard Focal Loss can only handle discrete binary values, we adopt Quality Focal Loss [27] to conduct the classification supervision on the non-discrete categorical labels:

$$\hat{y} = \alpha c + (1-\alpha)\tau, \tag{1}$$

$$\mathcal{L}_{cls}^{U} = -|y-\hat{y}|^{\beta}((1-y)\log(1-\hat{y}) + y\log(\hat{y})), \tag{2}$$

where $\hat{y}$ is the quality score predicted by the teacher and y is the student predictions, α is set to 0.75. Note that our approach can be easily extended into other continual versions of cross-entropy loss, for instance, Gaussian Focal Loss [23].

In addition to the classification loss, the boundary targets of the bounding box can present more ambiguities since it contains 7 degree-of-freedom and presents fewer training samples (only positive samples receive the regression supervision). Therefore, how to effectively deal with misleading regression targets is non-trivial. To address this problem, we convert the deterministic regression into the probability optimization task. Concretely, we model the network prediction of each bounding box as a Gaussian distribution h given the feature vector x:

$$\hat{h} = \mathcal{N}(\mu(x), \sigma(x)), \tag{3}$$

where $\mu(x)$ and $\sigma(x)$ denotes the mean and variance of each regression term predicted by the network. After that, the regression loss can be converted into a negative log-likelihood (NLL) loss, where the objective is to maximize the likelihood of each GT h in the predicted distribution:

$$\mathcal{L}_{reg} = -\log\mathcal{N}(h; \mu(x), \sigma(x)) \tag{4}$$

$$= -\log\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{h-\mu}{\sigma})^2}. \tag{5}$$

By converting the deterministic regression task to a probabilistic estimation problem, the model develop more tolerance against noisy information in the training data, therefore achieving better performance.

3.2. Dense Feature Consistency Constraint

Considering that labels can contain noise information that deteriorates the model performance, we further exploit the incorporation of unsupervised learning to obtain useful knowledge about label-independent features. However, different from the common-studied image classification task that mostly cares about the overall feature representation,

3D object detection requires estimating instances with various scales and directions. Therefore, learning a scale and rotation-invariant feature extractor for 3D detector is of vital importance. By applying various data augmentation, such as rescaling, rotating, and global transforming, into the input point clouds, and supervising them with the transformed ground-truth boxes, the model can develop robustness to such transformations. However, when the label itself is not so accurate enough, such a strategy may be more harmful.

To avoid this problem, we propose a dense pixel-wise consistency constraint to enforce the network to acquire such property. Previous literature mainly involves label-level consistency [55, 42, 1, 54] to ensure the model consistency, where the predicted bounding boxes are reversed back to the original space and regularized with Euclidean distance matching. Despite being simple and straightforward, this low-dimensional prediction distribution (number of classes and 7 regression targets) means that only a few amounts of knowledge are encoded, thus limiting the knowledge that can be transferred. Hence, we apply the consistency constraint on the feature level by reversing the BEV features according to the data transformation in the data augmentation and then conducting dense pixel-wise regularization.

More formally, given one frame of point clouds P and a set of data augmentation policies $\mathcal{A}$, we randomly sample two transformations $\mathcal{A}_1, \mathcal{A}_2$ from $\mathcal{A}$ and apply them to the point P to produce two different views of point clouds P_1, P_2. The augmented point clouds are fed into the point feature extractor $\mathcal{F}$ to generate the BEV feature maps F:

$$F_1 = \mathcal{F}(P_1), F_2 = \mathcal{F}(P_2). \tag{6}$$

Once obtaining the BEV features, we simply reverse them back to the original space with the recorded transformation flow:

$$\hat{F}_1 = \mathcal{A}_1^{-1}(F_1), \hat{F}_2 = \mathcal{A}_2^{-1}(F_2), \tag{7}$$

where A^{-1} denotes the reverse transformation of A. To this end, we can derive the pixel-wise feature consistency constraint with standard L2 loss:

$$\mathcal{L}_{consist} = ||\hat{F}_1 - \hat{F}_2||_2. \tag{8}$$

Considering that the point-based 3D features can hold meaningful information only if the point exists, mainly due to the characteristics of the LiDAR points, we further introduce a foreground-focusing mask to selectively regularize the augmented BEV features. Different from previous approaches [12] that directly set each foreground GT region with hard supervision, we generate a soft mask to smooth the regularization effect at the boundary regions, similar in [57]. Specifically, we draw a Gaussian distribution for each GT center (x_i, y_i) in the BEV space,

$$\phi_{i,x,y} = \exp(\frac{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}{2\sigma_i^2}), \tag{9}$$

where σ_i is a constant (set to 2), indicating the object size standard deviation. Since feature maps are class-agnostic, we merge all $\phi_{i,x,y}$ into one single mask Φ by taking the maximum value across the i dimension. Therefore, the final dense feature consistency constraint is derived as:

$$\mathcal{L}_{consist} = \frac{1}{HW\Phi} \sum_{i}^{H} \sum_{j}^{W} max(\phi_{ij}) ||\hat{F}_1 - \hat{F}_2||_2. \tag{10}$$

By aligning dense pixel-wise features on the BEV space, the model can gradually learn the ability to extract transform-invariant feature extraction and fully utilize the unlabeled data in a self-supervised manner.

4. Experiments

4.1. Datasets

ONCE dataset [33] is one of the largest autonomous driving datasets with 1 million LiDAR point clouds and 7 million paired images. Only 15,000 samples are annotated with 3D bounding boxes, which are divided into training, validation and testing split with a ratio of 33%/20%/47%, respectively. Among all instances, five classes are labeled as interested categories: Car, Bus, Truck, Pedestrian, and Cyclist. According to the official SSL setting, 5,000 samples are selected as labeled set and all the unlabeled sets are divided into 3 scales of subsets: `Small`, `Medium`, and `Large`. The small set contains 70 sequences (100k samples), the medium set contains 321 sequences (500k samples), and the large set contains 560 sequences (about 1M samples). Following the official evaluation tools, we set 0.7, 0.3, and 0.5 for Vehicle, Pedestrian, and Cyclist as the mAP IoU threshold, respectively. Besides, similar to Waymo Open Dataset [40], the evaluation protocol provides metrics based on different perception ranges: '0-30m', '30-50m', and /'50m-inf' to fully reflect the detection ability of the model.

4.2. Implementation Details

NoiseDet follows the two-stage training pipeline: we first train the teacher network with 80 epochs for ONCE dataset. Then we utilize the STE module proposed in ProficientTeacher [51] to obtain pseudo-labels on the unlabeled dataset. During this process, we can easily incorporate other techniques such as camera/Radar detection or object tracking to further refine the labels. We do not consider such implementations since it is out of the scope of this paper, but it is proven to be effective in [35]. For the semi-supervised training configuration, we uniformly sampled 1 labeled and

Table 1. Comparison with other state-of-the-art semi-supervised 3D object detection approaches on ONCE validation set with different amounts of unlabeled samples (*e.g.*, "Small", "Medium" and "Large"). We bold the best overall results and list the relative gains based on the baseline model (*i.e.*, SECOND [48] trained on labeled training set only) for better illustration. Most experimental results are borrowed from the official implementations in ONCE [33].

Methods	**Vehicle AP** (%)				**Pedestrian AP** (%)				**Cyclist AP** (%)				**mAP** (%)
	overall	0-30m	30-50m	50m-inf	overall	0-30m	30-50m	50m-inf	overall	0-30m	30-50m	50m-inf	
Baseline [48]	71.19	84.04	63.02	47.25	26.44	29.33	24.05	18.05	58.04	69.96	52.43	34.61	51.89
Small (100K unlabeled samples)													
Pseudo Label	72.80	84.46	64.97	51.46	25.50	28.36	22.66	18.51	55.37	65.95	50.34	34.42	51.22 (- 0.67)
Noisy Student [45]	73.69	84.69	67.72	53.41	28.81	33.23	23.42	16.93	54.67	65.58	50.43	32.65	52.39 (+ **0.50**)
Mean Teacher [41]	74.46	86.65	68.44	53.59	30.54	34.24	26.31	20.12	61.02	72.51	55.24	39.11	55.34 (+ **3.45**)
SESS [55]	73.33	84.52	66.22	52.83	27.31	31.11	23.94	19.01	59.52	71.03	53.93	36.68	53.39 (+ **1.50**)
3DIoUMatch [42]	73.81	84.61	68.11	54.48	30.86	35.87	25.55	18.30	56.77	68.02	51.80	35.91	53.81 (+ **1.92**)
NoiseDet (Ours)	**75.26**	86.36	67.52	55.29	**37.96**	42.36	32.78	23.28	**60.77**	72.31	55.03	38.87	**58.00** (+ **6.11**)
Medium (500K unlabeled samples)													
Pseudo Label	73.03	86.06	65.96	51.42	24.56	27.28	20.81	17.00	53.61	65.26	48.44	33.58	50.40 (- 1.49)
Noisy Student [45]	75.53	86.52	69.78	55.05	31.56	35.80	26.24	21.21	58.93	69.61	53.73	36.94	55.34 (+ **3.45**)
Mean Teacher [41]	76.01	86.47	70.34	55.92	35.58	40.86	30.44	19.82	63.21	74.89	56.77	40.29	58.27 (+ **6.38**)
SESS [55]	72.11	84.06	66.44	53.61	33.44	38.58	28.10	18.67	61.82	73.20	56.60	38.73	55.79 (+ **3.90**)
3DIoUMatch [42]	75.69	86.46	70.22	56.06	34.14	38.84	29.19	19.62	58.93	69.08	54.16	38.87	56.25 (+ **4.36**)
NoiseDet (Ours)	**77.14**	87.21	**69.94**	58.44	**40.45**	44.91	35.71	24.11	**62.59**	73.04	57.98	42.67	**60.06** (+ **8.17**)
Large (1M unlabeled samples)													
Pseudo Label	72.41	84.06	64.54	50.05	23.62	26.80	20.13	16.66	53.25	64.69	48.52	33.47	49.76 (- 2.13)
Noisy Student [45]	75.53	86.52	69.78	55.05	31.56	35.80	26.24	21.21	58.93	69.61	53.73	36.94	55.34 (+ **3.45**)
Mean Teacher [41]	76.38	86.45	70.99	57.48	35.95	41.76	29.05	18.81	**65.50**	75.72	60.07	43.66	59.28 (+ **7.39**)
SESS [55]	75.95	86.83	70.45	55.76	34.43	40.00	27.92	19.20	63.58	74.85	58.88	39.51	57.99 (+ **6.10**)
3DIoUMatch [42]	75.81	86.11	71.82	57.84	35.70	40.68	30.34	21.15	59.69	70.69	54.92	39.08	57.07 (+ **5.18**)
NoiseDet (Ours)	**78.02**	87.00	72.55	59.49	**42.89**	46.52	38.21	26.60	62.74	73.19	58.03	42.88	**61.16** (+ **9.27**)

Table 2. Comparison of detection results based on SECOND and CenterPoint with and without NoiseDet on ONCE validation set.

Detector	Method	mAP
SECOND	NoisyStudent	55.50
	NoiseDet (Ours)	**58.01**
CenterPoint	NoisyStudent	63.77
	NoiseDet (Ours)	**65.57**

4 unlabeled samples for each batch to avoid training collapse. Following the official ONCE benchmark [33], we initialize the student from a pretrained checkpoint on the full labeled set. The student is trained for 25, 50, 75 epochs for small, medium, and large settings on ONCE dataset. The learning rate is set to 1e-4 and the pseudo-labels are updated every 25 epochs. All models are trained on an 8 NVIDIA V100 GPUs machine.

4.3. Main Results

We first implement NoiseDet with the standard anchor-based 3D detector, SECOND, following the official benchmark [33], on the `Small` protocol setting of ONCE dataset. The final performance is shown in Table 2. Our NoiseDet greatly boosts its enhanced baseline NoisyStudent [45] by 2.5 mAP. Then, we also report it on a stronger anchor-free-based detector, CenterPoint [52], which also obtains an improvement of 1.8 mAP, validating the effectiveness and generalization of the proposed method under different 3D detection frameworks. Besides, we also validate NoiseDet on the competitive Waymo Open Dataset [40] in Appendix, which outperforms 3DIoUMatch [42] by a large margin.

4.4. Comparison with State-of-the-Arts

To further validate the superiority of our method, we compare NoiseDet with various state-of-the-art semi-supervised approaches, including Pseudo-Label, NoisyStudent, Mean Teacher, SESS, and 3DIouMatch. The experiments are conducted on the three different protocols (`Small`, `Medium`, and `Large`) to fully exploit the extensibility of various strategies and the results are reported in Table 1. We select SECOND as our baseline to keep consistent with other works [33]. NoiseDet surpasses all other counterparts, achieving new state-of-the-art in this competitive benchmark. With the increase of the unlabeled dataset size, NoiseDet can consistently improve the detectors with significant gains: with 6.1 mAP, 8.2 mAP, and 9.3 mAP under `Small`, `Medium`, and `Large` protocols, respectively.

4.5. Ablation Studies

In this section, we conduct a series of ablation studies to gain a deeper understanding of NoiseDet. For efficiency, all experiments are conducted on the `Small` protocol of the ONCE training set and evaluated on the validation set.

Table 3. Effectiveness of each component in our NoiseDet. Results are reported on ONCE validation subset with SECOND.

Noise-Resistant Instance Supervision		Dense Feature Consistency Constraint	mAP			
Classification	*Regression*		AP_{veh}	AP_{ped}	AP_{cyc}	$AP_{overall}$
			72.03	36.80	57.73	55.50
✓			72.78	36.41	60.73	56.80
	✓		73.87	36.74	58.34	56.39
✓	✓		74.05	36.62	61.03	57.27
✓	✓	✓	75.26	37.96	60.77	**58.01**

4.5.1 Main Ablations

In order to understand the effectiveness of each module contributing to the detection performance in NoiseDet, we test each component separately on the baseline detector SECOND and report the results in Table 3. Our baseline is the enhanced implementation of NoisyStudent [45], where its performance starts from 55.5 mAP on the validation dataset. When we add the noise-resistant instance supervision on the classification branch, the mAP score is raised by 1.1 mAP. And then we test the proposed component on the regression branch, which yields another 0.8 mAP. Such a huge improvement validates the correctness of the noise learning strategy on the unknown quality of pseudo-labels and the effectiveness of the proposed noise-resistant instance supervision module. When the dense feature consistency constraint module is applied, the accuracy is promoted by 1.2 mAP, suggesting that consistent regularization provides useful unsupervised hints for model convergence. Finally, we put all techniques together, and NoiseDet achieves 58.0 mAP on the validation subset, indicating a 2.5 mAP performance enhancement.

4.5.2 Noise-Resistant Instance Supervision

Comparison with other classification supervision. We first compare our proposed noise-resistant classification loss with other supervision approaches. The vanilla strategy is to directly apply Focal Loss on the pseudo labels, similar to the labeled data. Then, we attempt to utilize the predicted score from the teacher to adjust the supervision on the pseudo-labels by reweighting the classification loss. In doing this, we can alleviate the training distraction from low-quality samples by lowering their penalties. Inspired by 3DIoUMatch [42], we consider IoU as another measurement to help decide the loss penalty on the pseudo samples (denoted as IoU Reweight). We also consider Quality Focal Loss proposed in [27] to further integrate IoU into the classification supervision. The final results are shown in Table 4. Incorporating reweighting mechanisms, either IoU or classification score, to balance the supervision across different samples brings about 0.5 mAP improvements. When adapting to QFL, the model improves to 56.2 mAP. Finally, by both considering localization quality (IoU) and the sample quality (classification score), we obtains the best performance, 56.8 mAP on the validation subset.

Table 4. Comparison with different classification supervision approaches on ONCE validation dataset.

Classification Loss	$AP_{overall}$
Focal Loss [31]	55.45 $\pm$ 0.07
Score Reweight	56.09 $\pm$ 0.10
IoU Reweight	55.90 $\pm$ 0.17
QFocal Loss [27]	56.20 $\pm$ 0.05
Ours	**56.80** $\pm$ 0.08

Strategies on $\hat{y}$ generation. In this part, we explore the best strategy to generate the classification targets $\hat{y}$ for the Focal Loss. We first consider utilizing the teacher's predicted categorical confidence to replace the original hard pseudo labels, based on the observation that the quality of the generated pseudo-labels is highly correlated with the classification score. On top of this, we also consider the localization quality between the student and teacher's predictions. The main intuition is that when the student model can not perfectly learn the pseudo-targets well, we infer them as noisy labels and downgrade their categorical targets. Actually, this strategy comes to the same formulation as Quality Focal Loss. In order to further improve the representation ability of $\hat{y}$, we adopt the combination of IoU and confidence score to fully leverage the information. We consider two types of combinations: multiplication and weighted sum. The experimental results are shown in Table 5. By combining IoU and confidence score with weighted sum, the detection accuracy gets the best performance.

Comparison with other regression supervision. In addition to classification targets $\hat{y}$, properly softening the regression supervision on noisy pseudo labels is also non-trivial. Therefore, we compare different kinds of regression losses and their tolerance against noisy data. The original loss is Smooth L1 loss, which provides more penalty on more accurate predictions (close to GT targets), encouraging the model to pay more attention to high-precision boxes. However, such a strategy is counter to the learning on noisy

Table 5. Ablations on the strategies to generate the classification target $\hat{y}$ for categorical supervision.

Cls Score	IoU	Operation	$AP_{overall}$
		-	55.45 ± 0.07
✓		-	56.45 ± 0.17
	✓	-	56.11 ± 0.21
✓	✓	Multiply	56.52 ± 0.05
✓	✓	Weighted-sum	**56.80** ± 0.08

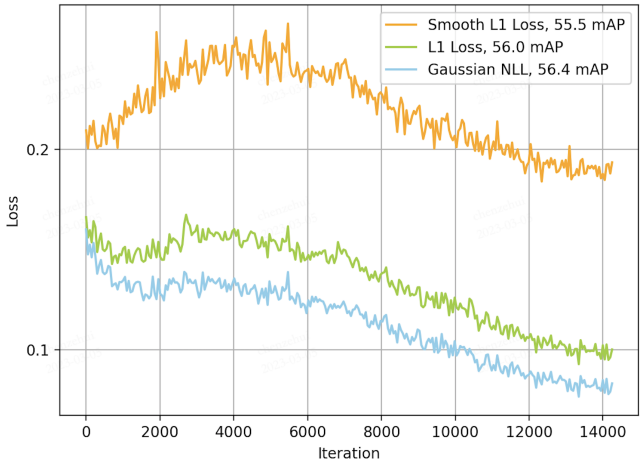

Figure 3. Visualization of the loss curve of different regression approaches on the labeled split in each batch. We also plot the final detection performance (mAP) at the *end* iteration of each loss. Note that there exits occasional fluctuations during training probably due to the noisy annotations in the training data, empirically it will not affect the final model performance.

labels: if the network fails to regress the target well, it probably belongs to low-quality samples. Therefore, we directly employ the L1 loss to avoid large misleading gradients on noisy bounding box regression. Another option to improve the tolerance against noisy labels for regression is to convert the regression task into a probabilistic optimization problem. As a result, we consider the Gaussian NLL as an alternative regression objective. We report the results in Figure 3. Gaussian NLL outperforms all other losses, achieving 56.4 mAP on the validation subset, demonstrating the superiority of the probabilistic model on noisy data learning.

4.5.3 Dense Feature Consistency Constraint

Strategies on consistency regularization. Consistency learning is a crucial component in semi-supervised pipelines. In most previous work [55], such regularization is often conducted on the label level. A key reason is that label encoding is agnostic to any transformation, which is easy to apply to various object detection tasks. Luckily, most input permutations in point clouds can be easily reversed and differentiable. Therefore, we can also explore the feature-level consistency regularization with L2 loss. Followed by the idea of SimSiam [6], we add a projection layer for one branch and detach the gradient in the other branch, and then regularize them with L2 loss. We also consider distance correlation to consider the inner structural similarity in the latent feature space. The final results are shown in Table 6. The vanilla L2 loss on the feature level obtains the best performance. We infer the reason that mutual L2 introduces additional flexibility through the linear projection layer and further promotes the effect of consistent regularization.

Table 6. Comparison with different approaches for dense feature consistency constraints. "DC" denotes distance correlation.

Input-level	Loss	AP
Label-level	L2	55.45 ± 0.07
Feature-level	DC	56.07 ± 0.14
Feature-level	L2	**56.70** ± 0.06

4.6. Extension to Auto-Labeling Strategy

Auto-Labeling is a simple and effective strategy to improve model prediction. By combing with additional techniques, such as temporal information or object tracking, we can obtain high-quality pseudo-labels [35]. Different from the Mean-Teacher paradigm, the pseudo-labeling pipeline allows the network to pursue more precious labels offline and greatly improves its potential in real-world applications. In this section, we imitate this paradigm and validate the superiority of our framework. Specifically, we generate the pseudo-labels based on two well-tuned CenterPoint models, (CenterPoint and CenterPoint-E), which achieve 62.5 and 70.1 mAP on the ONCE validation subset, respectively, to stimulate the auto-labeling process and utilize it to supervise a SECOND model. The final results are shown in Table 7. With the quality-improved label, our NoiseDet still consistently outperforms the competitive NoisyStudent baseline, yielding a 1.4 and 2.2 mAP on the ONCE validation subset, demonstrating its effectiveness and generalization on semi-supervised learning tasks.

Table 7. Detection results on different pseudo-label generation with respective teacher models. The results are reported on ONCE validation dataset. CenterPoint-E denotes for the enhanced version of CenterPoint.

PL Source	AP_{veh}	AP_{ped}	AP_{cyc}	$AP_{overall}$
SECOND [48]	75.26	37.96	60.77	58.00
CenterPoint [52]	75.42	40.19	62.73	59.44
CenterPoint-E [52]	76.01	41.56	63.01	**60.19**

5. Conclusion

In this paper, we introduce a pseudo-labeling-based semi-supervised 3D object detection framework, namely NoiseDet. By viewing the semi-supervised learning as a noisy learning task, we propose two core modules to overcome the ambiguity detection problem: noise-resistant instance supervision and dense feature consistency regularization. With soft task supervision on the unlabeled data and unsupervised feature consistency regularization, our model develops tolerance towards noisy pseudo-labels and improves the model generalization. Extensive experiments on the ONCE dataset demonstrate the effectiveness and generalization of our approach. We hope NoiseDet can provide a new perspective in dealing with in-sufficient accuracy pseudo labels for semi-supervised 3D object detection.

Acknowledgments

This work was supported by the JKW Research Funds under Grant 20-163-14-LZ-001-004-01, and the Anhui Provincial Natural Science Foundation under Grant 2108085UD12. We acknowledge the support of GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

References

[1] Amir Bar, Xin Wang, Vadim Kantorov, Colorado J Reed, Roei Herzig, Gal Chechik, Anna Rohrbach, Trevor Darrell, and Amir Globerson. DETReg: Unsupervised pretraining with region priors for object detection. In *CVPR*, pages 14605–14615, 2022. 5

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *NeurIPS*, 32:1–11, 2019. 3

[3] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-NMS–improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer vision*, pages 5561–5569, 2017. 3

[4] Paola Cascante-Bonilla, Fuwen Tan, Yanjun Qi, and Vicente Ordonez. Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. In *AAAI*, volume 35, pages 6912–6920, 2021. 3

[5] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *NeurIPS*, 33:22243–22255, 2020. 3

[6] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, pages 15750–15758, 2021. 8

[7] Xuesong Chen, Shaoshuai Shi, Benjin Zhu, Ka Chun Cheung, Hang Xu, and Hongsheng Li. Mppnet: Multi-frame feature intertwining with proxy points for 3D temporal object detection. *arXiv preprint arXiv:2205.05979*, 2022. 3

[8] Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Fast Point R-CNN. In *ICCV*, pages 9775–9784, 2019. 2

[9] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Autoalignv2: Deformable feature aggregation for dynamic multi-modal 3D object detection. *ECCV*, 2022. 1

[10] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Bevdistill: Cross-modal bev distillation for multi-view 3D object detection. *ICLR*, pages 1–17, 2022. 1

[11] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Graph-DETR3D: rethinking overlapping regions for multi-view 3D object detection. In *ACM MM*, pages 5999–6008, 2022. 1

[12] Zhiyu Chong, Xinzhu Ma, Hong Zhang, Yuxin Yue, Haojie Li, Zhihui Wang, and Wanli Ouyang. Monodistill: Learning spatial features for monocular 3D object detection. *arXiv preprint arXiv:2201.10830*, 2022. 5

[13] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3D object detection. In *AAAI*, volume 35, pages 1201–1209, 2021. 2

[14] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *ICCV*, pages 6569–6578, 2019. 2

[15] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. RangeDet: In defense of range view for LiDAR-based 3D object detection. In *ICCV*, pages 2918–2927, 2021. 2

[16] Ross Girshick. Fast R-CNN. In *ICCV*, pages 1440–1448, 2015. 3

[17] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *NeurIPS*, 17:1–8, 2004. 3

[18] Sheng Guo, Weilin Huang, Haozhi Zhang, Chenfan Zhuang, Dengke Dong, Matthew R Scott, and Dinglong Huang. Curriculumnet: Weakly supervised learning from large-scale web images. In *ECCV*, pages 135–150, 2018. 3

[19] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *NeurIPS*, 31, 2018. 3

[20] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. *NeurIPS*, 32, 2019. 1

[21] JongMok Kim, Jooyoung Jang, Seunghyeon Seo, Jisoo Jeong, Jongkeun Na, and Nojun Kwak. Mum: Mix image tiles and unmix feature tiles for semi-supervised object detection. In *CVPR*, pages 14512–14521, 2022. 3

[22] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, pages 12697–12705, 2019. 2

[23] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *ECCV*, pages 734–750, 2018. 4

[24] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013. 2, 3

[25] Gang Li, Xiang Li, Yujie Wang, Shanshan Zhang, Yichao Wu, and Ding Liang. Pseco: Pseudo labeling and consistency training for semi-supervised object detection. *ECCV*, 2022. 1, 3

[26] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020. 3

[27] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *NeurIPS*, 33:21002–21012, 2020. 4, 7

[28] Zhenyu Li, Zehui Chen, Ang Li, Liangji Fang, Qinhong Jiang, Xianming Liu, and Junjun Jiang. Unsupervised domain adaptation for monocular 3D object detection via self-training. *ECCV*, pages 1–17, 2022. 2

[29] Zhichao Li, Feng Wang, and Naiyan Wang. LiDAR R-CNN: An efficient and universal 3D object detector. In *CVPR*, pages 7546–7555, 2021. 1

[30] Zhidong Liang, Ming Zhang, Zehan Zhang, Xian Zhao, and Shiliang Pu. Rangercnn: Towards fast and accurate 3D object detection with range image representation. *arXiv preprint arXiv:2009.00206*, 2020. 2

[31] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 7

[32] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. *ICLR*, 2021. 3

[33] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Hanxue Liang, Jingheng Chen, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, et al. One million scenes for autonomous driving: Once dataset. *arXiv preprint arXiv:2106.11037*, 2021. 5, 6

[34] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, pages 652–660, 2017. 2

[35] Charles R Qi, Yin Zhou, Mahyar Najibi, Pei Sun, Khoa Vo, Boyang Deng, and Dragomir Anguelov. Offboard 3D object detection from point cloud sequences. In *CVPR*, pages 6134–6144, 2021. 2, 5, 8

[36] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *ICLR*, 2021. 2

[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. In *CVPR*, pages 10529–10538, 2020. 1, 2

[38] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3D object proposal generation and detection from point cloud. In *CVPR*, pages 770–779, 2019. 2

[39] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020. 1

[40] 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 *CVPR*, pages 2446–2454, 2020. 1, 5, 6

[41] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *NeurIPS*, 30, 2017. 2, 3, 6

[42] 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, 3, 5, 6, 7

[43] Jianren Wang, Haiming Gang, Siddarth Ancha, Yi-Ting Chen, and David Held. Semi-supervised 3D object detection via temporal graph neural networks. In *3DV*, pages 413–422. IEEE, 2021. 1

[44] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *NeurIPS*, 33:6256–6268, 2020. 3

[45] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, pages 10687–10698, 2020. 2, 3, 6, 7

[46] Hongyi Xu, Fengqi Liu, Qianyu Zhou, Jinkun Hao, Zhijie Cao, Zhengyang Feng, and Lizhuang Ma. Semi-supervised 3D object detection via adaptive pseudo-labeling. In *ICIP*, pages 3183–3187. IEEE, 2021. 1, 2, 3

[47] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *ICCV*, pages 3060–3069, 2021. 1, 2

[48] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 6, 8

[49] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3DSSD: Point-based 3D single stage object detector. In *CVPR*, pages 11040–11048, 2020. 2

[50] Zetong Yang, Yin Zhou, Zhifeng Chen, and Jiquan Ngiam. 3D-MAN: 3D multi-frame attention network for object detection. In *CVPR*, pages 1863–1872, 2021. 3

[51] Junbo Yin, Jin Fang, Dingfu Zhou, Liangjun Zhang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. Semi-supervised 3D object detection with proficient teachers. In *ECCV*, pages 727–743. Springer, 2022. 1, 2, 3, 5

[52] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3D object detection and tracking. In *CVPR*, pages 11784–11793, 2021. 2, 6, 8

[53] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *NeurIPS*, 34:18408–18419, 2021. 2, 3

[54] Jinnian Zhang, Houwen Peng, Kan Wu, Mengchen Liu, Bin Xiao, Jianlong Fu, and Lu Yuan. MiniViT: Compressing vision transformers with weight multiplexing. In *CVPR*, pages 12145–12154, 2022. 5

[55] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Sess: Self-ensembling semi-supervised 3D object detection. In *CVPR*, pages 11079–11087, 2020. 2, 3, 5, 6, 8

[56] Wu Zheng, Weiliang Tang, Li Jiang, and Chi-Wing Fu. Se-ssd: Self-ensembling single-stage object detector from point cloud. In *CVPR*, pages 14494–14503, 2021. 1

[57] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 5

[58] Yin Zhou, Pei Sun, Yu Zhang, Dragomir Anguelov, Jiyang Gao, Tom Ouyang, James Guo, Jiquan Ngiam, and Vijay Vasudevan. End-to-end multi-view fusion for 3D object detection in LiDAR point clouds. In *CoRL*, pages 923–932. PMLR, 2020. 2

[59] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3D object detection. In *CVPR*, pages 4490–4499, 2018. 2