Learning Object-level Point Augmentor for Semi-supervised 3D Object Detection

Cheng-Ju Ho*[1]
ace52751208@gmail.com
Chen-Hsuan Tai*[1]
derek.0417t@gmail.com
Yi-Hsuan Tsai‡[2]
wasidennis@gmail.com
Yen-Yu Lin[1]
lin@cs.nycu.edu.tw
Ming-Hsuan Yang[3]
mhyang@ucmerced.edu

[1] National Yang Ming Chiao Tung University, Taiwan
[2] Phiar Technologies, United States
[3] University of California at Merced, United States

Abstract

Semi-supervised object detection is important for 3D scene understanding because obtaining large-scale 3D bounding box annotations on point clouds is time-consuming and labor-intensive. Existing semi-supervised methods usually employ teacher-student knowledge distillation together with an augmentation strategy to leverage unlabeled point clouds. However, these methods adopt global augmentation with scene-level transformations and hence are sub-optimal for instance-level object detection. In this work, we propose an object-level point augmentor (OPA) that performs local transformations for semi-supervised 3D object detection. In this way, the resultant augmentor is derived to emphasize object instances rather than irrelevant backgrounds, making the augmented data more useful for object detector training. Extensive experiments on the ScanNet and SUN RGB-D datasets show that the proposed OPA performs favorably against the state-of-the-art methods under various experimental settings. The source code will be available at https://github.com/nomiaro/OPA.

1 Introduction

3D object detection aims to recognize and localize objects in a 3D scene by specifying them with their oriented bounding boxes and semantic classes. Compared to 2D images, 3D scenes provide rich geometric structure information and hence are crucial for many advanced 3D vision applications such as autonomous driving, AR/VR, and robot navigation. Recent research efforts [5, 16, 19, 20, 21, 33, 34, 38] have been made on 3D object detection and achieve significant progress. However, most existing methods are data-hungry and rely on

*The authors contribute equally to this paper
‡Currently at Google

large-scale labeled 3D objects, leading to a vast amount of costly manual efforts. To address this issue, it is favorable to develop semi-supervised learning (SSL) algorithms for 3D object detection where plenty of unlabeled 3D point clouds can be leveraged to compensate for the lack of labeled data and to improve detector training.

Several SSL approaches [9, 14, 22, 23, 27] for 2D object detection are developed based on teacher-student mutual learning, where pseudo-labels of unlabeled data are estimated and used as supervisory signals for detector training. For 3D object detection, 3DIoUMatch [29] employs two identical pre-trained networks to initialize a teacher-student model and applies asymmetric data augmentations to transform data samples. To be specific, the input data to the student model are globally transformed by strong augmentations for data regularization and variance enhancement, thus offering rich information to boost the capability of the student model. On the other hand, the input data to the teacher model are obtained by weak augmentations to generate pseudo-labels to supervise the student model. Prior work [1, 14, 23, 29, 36] shows that this asymmetric data augmentation mechanism is crucial for improving semi-supervised learning in a teacher-student model. However, most existing SSL methods for 3D object detection, such as 3DIoUMatch [29], adopt scene-level transformations, which is sub-optimal as augmenting irrelevant backgrounds may degrade the effectiveness of the augmented data. To address this issue, we present a method that takes both global and object-level data augmentations into consideration and thus generates more plausible augmented point cloud objects for SSL.

Compared to 3DIoUMatch which applies augmentations such as rotation and scaling to the entire point cloud *scene*, our method focuses on point cloud *object* augmentation, which better benefits the teacher-student framework. In this work, we present OPA based on a teacher-student mutual learning framework with an object-level augmentor for semi-supervised 3D object detection. To this end, we utilize a two-stage training procedure, including the pre-training and semi-supervised learning stages. First, we design an adversarial formulation to jointly pre-train a detector with an augmentor, where the augmentor takes point clouds within the object bounding box as the input, as well as the objectness guidance from the detector to control the learning pace in augmentation. Then, the augmentor outputs displacement values for each point as augmentation to improve data variations for the detector.

In the semi-supervised learning stage, we freeze the learned augmentor and use it to produce the object-level augmented point clouds. We leverage both ground-truth and pseudo-labeled bounding boxes inferred by the teacher model, respectively from the labeled and unlabeled data, to identify point cloud objects that serve as the input to the augmentor. As a result, the produced point clouds exhibit local variations and are complementary to those produced by global scene-level augmentations, thus improving the teacher-student model learning. In experiments, we show that our OPA performs favorably against the state-of-the-art methods for semi-supervised object detection on two benchmark datasets, including ScanNet [6] and SUN RGB-D [24]. In addition, we demonstrate that the proposed augmentor is effective when it is applied to labeled or unlabeled point clouds, and is beneficial from our designed augmentor loss function that is aware of the objectness score from the detector. The main contributions of this work are summarized as follows:

1. We propose a simple yet effective method for semi-supervised 3D object detection via introducing an object-level augmentation strategy in point cloud scenes.

2. We integrate the proposed augmentor into the teacher-student mutual learning framework and jointly train the entire model to make use of labeled and unlabeled data.

3. We design a learning mechanism to make augmentor aware of the objectness from the detector and thus generate appropriate augmentations to improve 3D object detection.

2 Related Work

Semi-supervised Learning. Semi-supervised learning (SSL) aims to train a model using few labeled data and abundant unlabeled data. Numerous SSL strategies have been developed in the literature. 1) *Consistency regularization*: Methods of this category such as [1, 9, 18, 31] apply different transformations to a data sample and enforce consistency of model predictions among the transformed samples. 2) *Teacher-student framework*: It often employs two identical networks, one for a teacher model and the other for a student model [22, 27]. The teacher model is first frozen to guide the student model and is then updated from the student model. 3) *Pseudo-labeling*: It usually works in a self-supervised manner and derives the model using unlabeled data with their estimated pseudo-labels [10]. Fixmatch [22] combines the teacher-student framework and pseudo-labeling. It utilizes both student's and the teacher's predictions to enhance the quality of pseudo-labels. One key component of this method is asymmetric data augmentation. The strongly augmented inputs, e.g., those via Mixup [35], to the student model enrich data variance for model training, while the weakly augmented inputs to the teacher model ensure more accurate pseudo-labels for supervision. Based on the teacher-student framework, we propose an effective object-level augmentation method that focuses on point cloud instances in a scene.

Semi-supervised Object Detection. For 2D object detection in SSL, consistency-based methods [7, 25, 26] enforce the prediction consensus over different augmentations. Moreover, self-supervised approaches [13, 23, 30] apply a teacher-student framework with pseudo-label supervisions [11, 14, 23, 26, 32, 37]. For instance, STAC [23] and Unbiased Teacher [14] apply the teacher-student framework with asymmetric data augmentation to enlarge data variance and filter pseudo-labels to keep high-confidence object proposals. However, for the 3D scenario, there are fewer explorations of SSL for 3D object detection. SESS [36] enforces consistency over different augmentations as regularization. Furthermore, 3DIoUMatch [29] designs a 3D IoU estimation module based on VoteNet [16] as an IoU-aware Votenet, which calculates the IoU score of object proposals. Then, it takes IoU scores into account to filter out low-confidence pseudo-labels, with a selective mechanism to supervise unlabeled data using filtered high-quality pseudo-labels. In contrast, the proposed OPA introduces object-level point augmentations, which is an essential step towards a successful teacher-student framework for SSL, and has not been widely studied in 3D object detection.

Data Augmentation on Point Clouds. Data augmentation is important for deep learning. Because training data cannot cover all kinds of scenarios in the complex world, data augmentation is utilized to enlarge the diversity of training data. In 3D point cloud tasks, global augmentation operations like rotation, scaling, and translation with point-wise jittering [15, 17] are commonly used. However, those augmentation methods cannot transform the local structure in a point cloud. Therefore, recent works aim to improve the augmentation strategies for point clouds. The method in [4] divides an object and applies different augmented operations in each partition. Moreover, PointAugment [12] trains an auto-augmentor network that can learn to augment point cloud samples for better point cloud classification. PointWOLF [8] presents another method for the classification task where a convex combination of multiple transformations with smoothly varying weights carries out the local structure augmentation.

Table 1: Results of pre-defined object-level augmentations.

Setting	ScanNet 10%		SUN RGB-D 5%	
	mAP@0.25	mAP@0.5	mAP@0.25	mAP@0.5
Without Object-level Aug.	47.1	28.3	39.0	21.1
Pre-defined Object-level Aug. (scale, flip, rotation)	42.7	24.2	24.9	13.6
Pre-defined Object-level Aug. (displacement, range at 0.5%)	48.4	29.1	40.6	20.4
Pre-defined Object-level Aug. (displacement, range at 1%)	49.0	29.3	40.5	20.9
Pre-defined Object-level Aug. (displacement, range at 5%)	47.3	27.4	39.5	20.5

Based on the Mixup [35] idea in images, PointMixup [2] interpolates two point cloud objects to create an augmented point cloud, and the model is trained to predict the ratio of two mixed classes with a soft label. PointMixSwap [28] further explores the structural variance across multiple point clouds and generates more diverse point clouds for training data enrichment. For 3D object detection, PPBA [3] iteratively finds the best augmentation parameters of specific operations and applies them to the entire scene.

Compared to the above-mentioned methods that focus on the classification task or the combination of pre-defined augmentation operations, we study the SSL setting for 3D object detection by introducing a simple yet effective augmentation method. We focus on learning an augmentor that can synthesize object-level point clouds for foreground objects, serving as a better asymmetric augmentation module that is jointly trained in a teacher-student framework to achieve better SSL performance.

3 Proposed Method

This section elaborates the proposed method OPA. We give the problem definition and method overview in Section 3.1, and then describe our object-level augmentor and its training pipeline in Section 3.2.

3.1 Problem Definition and Algorithmic Overview

Given a 3D point cloud scene of S points $\mathbf{x} \in \mathbb{R}^{S\times 3}$, 3D object detection aims to recognize and locate objects of interest in $\mathbf{x}$ and describe them by their semantic classes and oriented bounding boxes. For learning a 3D object detector under the semi-supervised setting, we are given N_l labeled scenes $\{\mathbf{x}_i^l, \mathbf{y}_i^l\}_{i=1}^{N_l}$ and N_u unlabeled scenes $\{\mathbf{x}_i^u\}_{i=1}^{N_u}$, where $N_l << N_u$ in practice. The ground-truth annotation $\mathbf{y}_i^l$ stores the oriented bounding boxes $\{b_k\}$ and semantic labels $\{c_k\}$ of the objects of interest $\{o_k\}$ in $\mathbf{x}_i^l$.

Teacher-student knowledge distillation with asymmetric data augmentation has shown its effectiveness for semi-supervised 3D object detection. However, previous works [29, 36] focus on scene-level augmentation and ignore that object-level variances are crucial for detection. One way to address this issue is to apply augmentations, e.g., a random rotation, flip, and scale, to the point clouds within each object bounding box. However, such a method is sub-optimal, and its performance depends on proper augmentation settings. In Table 1, we find that pre-defined random augmentations, especially rotations, may confuse model learning and even harm the performance significantly. For scene-level augmentation on 3D object detection, rotation is widely used to enhance data variance without changing geometric relationships between foreground objects and background. However, for object-level augmentation in a scene, each object has its own orientation with respect to the global scene. Thus, changing the object-scene context during augmentation may lead to negative effects.

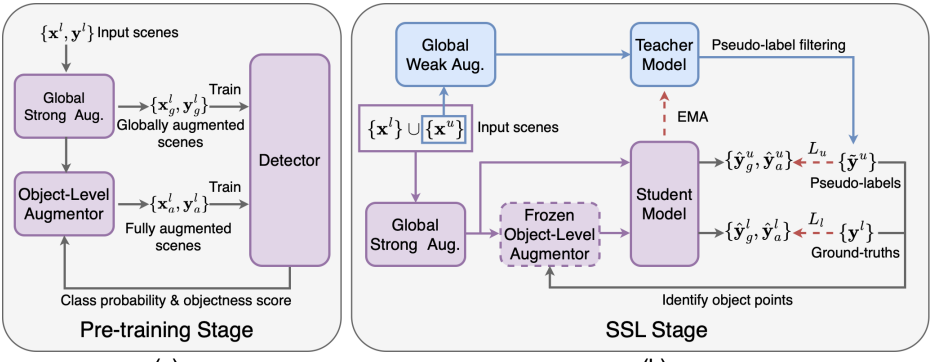

Figure 1: **OPA pipeline at (a) the pre-training stage and (b) the SSL stage.** In the pre-training stage, we utilize globally and fully augmented labeled scenes to jointly train the detector and augmentor using an adversarial strategy. In the SSL stage, we leverage the teacher-student framework with our frozen object-level augmentor. The teacher model consumes unlabeled data to generate high-quality pseudo-labels. Both labeled and unlabeled data are globally augmented and fully augmented to train the student model, where the augmentor takes points within each object bounding box as input and outputs the augmented points. Finally, the teacher model is updated from student model via EMA.

We instead consider point displacement for augmentaiton since it can enhance object-level data variance while keeping object orientations. As shown in Table 1, we try different ranges of displacement. Although using random displacements slightly improves the performance, it requires to pre-define a proper range of displacement, e.g., using too large or too small displacements may not be optimal. These issues motivate us to develop a better strategy via learning an augmentor for object-level augmentation that benefits 3D object detection. By learning an augmentor to generate proper displacement values, we preserve the intrinsic characteristics of an object and avoid over-deforming it.

Teacher-Student Framework in SSL. We aim to learn an augmentor that can synthesize plausible object instances while excluding irrelevant backgrounds, without twisting any augmentation parameters. Moreover, the augmentor can be integrated into the teacher-student framework and supports SSL. Fig. 1(b) shows the training pipeline. The teacher and student models are initialized from the same model. The teacher model is updated from the student model using the exponential moving average (EMA) mechanism, while pseudo-labels of unlabeled data are generated by the teacher model and are filtered to provide high-quality labels to the student model. The ground-truth and pseudo-labeled bounding boxes respectively from the labeled and unlabeled data are used to supervise the student model.

A key component making the teacher-student framework effective is data augmentation. We first follow [29] to apply the global transformations (e.g., rotation, flip, scale) to point cloud scenes, where the weak and strong augmentations are used for the teacher and student models, respectively. More details can be referred to [29]. To integrate our object-level augmentor, after global augmentation, we apply our augmentor to points within each object bounding box. Note that we only use the augmentor for the student model (see Fig. 1(b)), as the student model is the main model for updating parameters from loss functions. In practice, we also have tried to apply our augmentor to the teacher model but it does not show significant differences. To train our augmentor, we utilize a pre-training stage, shown

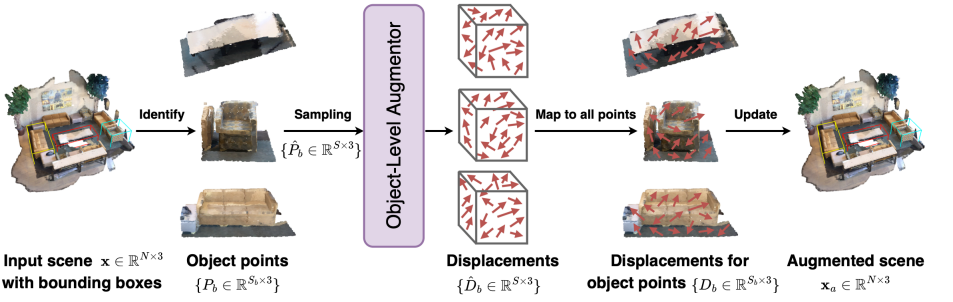

Figure 2: Given a point cloud scene $\mathbf{x} \in \mathbb{R}^{N\times 3}$ with M objects, we identify the object points $\{P_b \in \mathbb{R}^{S_b\times 3}\}_{b=1}^{M}$ in the M bounding boxes, where S_b is the number of points within the bth bounding box. Point sampling is applied to $\{P_b\}$ and makes each of the resultant sampled objects $\{\hat{P}_b\}$ have S points, which then serve as the input to the augmentor. The augmentor outputs the displacements $\{\hat{D}_b \in \mathbb{R}^{S\times 3}\}$. We map them back to their original szies $\{D_b \in \mathbb{R}^{S_b\times 3}\}$ via reverse sampling. Finally, $\{D_b\}$ is added back to the scene to obtain the object-level augmented scene $\mathbf{x}_a \in \mathbb{R}^{N\times 3}$.

in Fig. 1(a), to jointly train the augmentor and detector using only the labeled data. The reason is that, in the SSL stage, we find that using the unlabeled data with noisy pseudo-labeled bounding boxes would cause instability in training the augmentor. More details are described in the following section.

3.2 Object-level Point Augmentor

In this work, we aim to train an object-level augmentor that can determine point-wise parameters for foreground points and increase the variation of local structure in a scene. Different from PointAugment [17], we use only the point-wise displacement $\mathbb{D}$ to transform object points since we observe that random rotation is not helpful in 3D object detection as mentioned in Section 3.1. In addition, we dynamically learn the augmentor that controls the appropriate magnitude of point displacement based on objectness scores of the detector. Lastly, we leverage both labeled and unlabeled data to mutually update both the detector and the augmentor via an adversarial learning strategy.

Augmentation Process. The augmentation processing is illustrated in Fig. 2. Given a globally augmented 3D point cloud scene containing objects and their bounding boxes $\{\mathbf{x}_g^l, \mathbf{y}_g^l\}$, we sample M foreground objects from this scene to apply object-level augmentations. For unlabeled scene, we utilize its pseudo-labeled bounding boxes $\{\mathbf{x}_g^u, \tilde{\mathbf{y}}_g^u\}$. For each scene, the points inside the M bounding box proposals are collected, i.e., $\{P_b \in \mathbb{R}^{S_b\times 3}\}_{b=1}^{M}$, where S_b is the number of points inside the bth proposal. Then, we either up-sample by padding or down-sample by farthest point sampling (FPS) to make each object have exactly S points $\{\hat{P}_b \in \mathbb{R}^{S\times 3}\}_{b=1}^{M}$, while keeping each object structure unchanged. The augmentor takes the sampled objects $\{\hat{P}_b\}_{b=1}^{M}$ as input and outputs point-wise displacements $\{\hat{D}_b \in \mathbb{R}^{S\times 3}\}_{b=1}^{M}$ for point clouds $\{\hat{P}_b\}_{b=1}^{M}$. To match displacement $\{\hat{D}_b\}_{b=1}^{M}$ back to the point clouds of the original sizes $\{D_b \in \mathbb{R}^{S_b\times 3}\}_{b=1}^{M}$, we record the mapping from P_b to $\hat{P}_b$ and apply the reverse mapping. The point-wise displacement $\{D_b\}_{b=1}^{M}$ is added to the object points $\{P_b\}_{b=1}^{M}$ as our object-level augmentation. The fully augmented scene $\mathbf{x}_a$ is obtained by replacing the original object points with the augmented points. Note that after obtaining augmented points,

we restrict the displacement not to exceed the original bounding box. The original object points within the bounding box can be replaced by the augmented points that fit the original background while not affecting other objects.

Joint Augmentor and Detector Training. We use labeled data, including globally augmented samples $\{\mathbf{x}_g^l\}$ and fully augmented samples $\{\mathbf{x}_a^l\}$ via our augmentor, to jointly train the detector and augmentor in the pre-training stage. The augmentor is optimized to generate proper augmented scene $\mathbf{x}_a^l$ and to maximize the detector capability, while the detector is derived to localize and recognize the augmented data accurately.

Detector Loss. For training the detector, we formulate the loss function $\mathcal{L}_D$ as follows:

$$\mathcal{L}_D = \mathcal{L}_d(\mathbf{x}_g^l, \mathbf{y}_g^l) + \mathcal{L}_d(\mathbf{x}_a^l, \mathbf{y}_a^l), \tag{1}$$

where $\mathcal{L}_d$ is the detection loss used in [29].

Augmentor Loss. Similar to PointAugment [12], the fully augmented sample $\mathbf{x}_a^l$ should satisfy the following two requirements: 1) Predicting $\mathbf{x}_a^l$ should be more challenging than $\mathbf{x}_g^l$, i.e., $\mathcal{L}_d(\mathbf{x}_a^l, \mathbf{y}_a^l) \geq \mathcal{L}_d(\mathbf{x}_g^l, \mathbf{y}_g^l)$; 2) $\mathbf{x}_a^l$ and $\mathbf{x}_g^l$ should be similar to some degree by enforcing that they are predicted as the same class. To satisfy the two requirements, we use a dynamic variable ρ to control the augmentation magnitude. $\mathcal{L}_d(\mathbf{x}_a^l, \mathbf{y}_a^l)$ should be larger than $\mathcal{L}_d(\mathbf{x}_g^l, \mathbf{y}_g^l)$ for the first requirement and should not become too far for the second requirement. Thus, we make $\rho\mathcal{L}_d(\mathbf{x}_g^l, \mathbf{y}_g^l)$ be the upper bound of $\mathcal{L}_d(\mathbf{x}_a^l, \mathbf{y}_a^l)$. With a larger value of ρ, the augmentor generates more challenging augmented samples. On the other hand, the smaller value of ρ can avoid over-deforming the augmented samples. The augmentor loss $\mathcal{L}_A$ is formulated as

$$\mathcal{L}_A = \mathcal{L}_d(\mathbf{x}_a^l, \mathbf{y}_a^l) + \lambda |1 - \exp(\mathcal{L}_d(\mathbf{x}_a^l, \mathbf{y}_a^l) - \rho\mathcal{L}_d(\mathbf{x}_g^l, \mathbf{y}_g^l))|, \tag{2}$$

where λ is a pre-defined constant used to balance the importance between the object detection term and the augmentation magnitude term. $\rho \geq 1$ is set to ensure $\mathcal{L}_d(\mathbf{x}_a^l, \mathbf{y}_a^l) \geq \mathcal{L}_d(\mathbf{x}_g^l, \mathbf{y}_g^l)$, while ρ cannot be too high otherwise the augmented samples become too challenging. To balance it, we follow [12] and bound ρ between 1 and a value based on the classification probability. Different from [12], we further include a term $\hat{\mathbf{y}}_o$ to make ρ aware of the objectness for our object detection task:

$$\rho = \max(1, \exp(\hat{\mathbf{y}}_o \cdot \sum_{c=1}^{C} \hat{\mathbf{y}}_c \cdot \mathbf{y}_c)), \tag{3}$$

where C is the number of classes. $\mathbf{y}_c$, $\hat{\mathbf{y}}_c$, and $\hat{\mathbf{y}}_o$ are the class label, classification probability, and objectness score, respectively.

We find that our introduced $\hat{\mathbf{y}}_o$ term is critical to our task. As a metric to evaluate the objectness ability, $\hat{\mathbf{y}}_o$ is more suitable than the IoU score which is too sensitive to the bounding box location. When the class probability or the objectness score, *i.e.* $\hat{\mathbf{y}}_o$, of a sample is higher, it implies that this sample can be well classified by the detector, so we may use a larger value of ρ to allow more augmentations and make the augmented sample more challenging. Since the augmentor is learned in a class-agnostic fashion, the objectness score provides class-agnostic guidance to control the difficulty of the augmented samples, which in turn improves the learning of object detector. Finally, we alternatively train $\mathcal{L}_D$ and $\mathcal{L}_A$ in the pre-training stage.

Overall Loss Functions for SSL. In the SSL stage, we initialize the student and teacher models from the pre-trained detector and freeze the augmentor. The training pipeline is illustrated in Fig. 1(b). In each training batch, there are labeled samples $\{\mathbf{x}^l, \mathbf{y}^l\}$ and unlabeled samples $\{\mathbf{x}^u\}$. After applying global augmentation and our augmentor, we collect four kinds of data for the student model to learn: globally augmented labeled data $\{\mathbf{x}^l_g\}$ and unlabeled data $\{\mathbf{x}^u_g\}$, fully augmented labeled data $\{\mathbf{x}^l_a\}$ and unlabeled data $\{\mathbf{x}^u_a\}$. The student model outputs corresponding predictions: $\hat{\mathbf{y}}^l_g$, $\hat{\mathbf{y}}^u_g$, $\hat{\mathbf{y}}^l_a$, and $\hat{\mathbf{y}}^u_a$. For labeled data, $\hat{\mathbf{y}}^l_g$ and $\hat{\mathbf{y}}^l_a$ are supervised with the ground truths via

$$\mathcal{L}_l = \mathcal{L}_d(\mathbf{x}^l_g, \mathbf{y}^l_g) + \mathcal{L}_d(\mathbf{x}^l_a, \mathbf{y}^l_a). \tag{4}$$

For unlabeled data, $\hat{\mathbf{y}}^u_g$ and $\hat{\mathbf{y}}^u_a$ are supervised by filtered pseudo-labels $\tilde{\mathbf{y}}^u$:

$$\mathcal{L}_u = \mathcal{L}_d(\mathbf{x}^u_g, \tilde{\mathbf{y}}^u_g) + \mathcal{L}_d(\mathbf{x}^u_a, \tilde{\mathbf{y}}^u_a). \tag{5}$$

The overall loss in SSL for both labeled and unlabeled data is $\mathcal{L}_{SSL} = \mathcal{L}_l + \mathcal{L}_u$. The teacher model is updated by Exponential Moving Average (EMA) from the student model.

4 Experiments

Datasets. We follow the settings in the prior work [29, 36] for semi-supervised 3D object detection. ScanNet [6] is a 3D indoor benchmark dataset. It contains 1,201 training and 312 validation scenes with the reconstructed meshes. We focus on the 18 semantic classes. SUN RGB-D [24] is another 3D indoor benchmark dataset. It is composed of 5,285 training and 5,050 validation scenes. We use 10 object classes to evaluate our model.

Evaluation Metrics. For both benchmarks, we split them into the labeled and unlabeled data to perform semi-supervised learning. We apply 5%, 10%, and 20% labeled data ratio settings to conduct our experiments. We adopt mAP (mean average precision) as the evaluate metrics and report mAP@0.25 (mAP with 3D IoU threshold at 0.25) and mAP@0.5 scores.

Implementation Details. For pre-training, we use a batch size as 4 to train the augmentor. We use $M = 3$ foreground objects in one scene and sample $S = 1024$ points using either FPS or point padding according to the original point size. We train the detector and the augmentor for 900 epochs and use the Adam optimizer with an initial learning rate of 0.001. The learning rate decay by 0.1 occurs in the 400^{th}, 600^{th}, and 800^{th} epoch. To further stabilize the training, we leverage a warm-up mechanism that does not train the augmentor for the first 100 epochs. For the augmentor loss (2), we set $\lambda = 0.1$.

In the SSL stage, a batch is composed of two labeled data and four unlabeled data. We leverage ground truth bounding boxes to identify $S = 3$ foreground objects in labeled data, while for unlabeled data, we randomly pick $S = 3$ foreground objects from top-6 pseudo-labels with the highest confidence calculated by the IoU and objectness scores from the detector outputs. The IoU score represents the localization quality of the proposals and the objectness score shows the classification quality. We take both them into account and select the pseudo-labels of high quality. This mechanism avoids some easy samples with high confidence being selected all the time, which increases the chance that the model can observe more data variations. We train the detector for 1,000 epochs and use the Adam optimizer with an initial learning rate of 0.002. The learning rate decays 0.3, 0.3, 0.1, 0.1 at 400th, 600th, 800th, and 900th epochs, respectively. We conduct experiments on a single

Table 2: Results on ScanNet val set and SUN RGB-D val set for 5%, 10%, 20% labeled data ratio. We run the experiments under 3 random data splits and report our result in mean±standard deviation for the mAP@0.25 and mAP@0.50 metric.

Dataset	Model	5%		10%		20%	
		mAP @0.25	mAP @0.5	mAP @0.25	mAP @0.5	mAP @0.25	mAP @0.5
ScanNet	VoteNet [16]	27.9±0.5	10.8±0.6	36.9±1.6	18.2±1.0	46.9±1.9	27.5±1.2
	SESS [36]	NA	NA	39.7±0.9	18.6	47.9±0.4	26.9
	3DIoUMatch [29]	40.0±0.9	22.5±0.5	47.2±0.4	28.3±1.5	52.8±1.2	35.2±1.1
	OPA	**41.9**±1.5	**25.0**±0.4	**50.5**±0.2	**32.7**±1.0	**54.7**±0.3	**36.8**±0.8
	Gain (%)	1.9↑	2.5↑	3.3↑	4.4↑	1.9↑	1.6↑
SUN RGB-D	VoteNet [16]	29.9±1.5	10.5±0.5	38.9±0.8	17.2±1.3	45.7±0.6	22.5±0.8
	SESS [36]	NA	NA	42.9±1.0	14.4	47.9±0.5	20.6
	3DIoUMatch [29]	39.0±1.9	21.1±1.7	45.5±1.5	28.8±0.7	49.7±0.4	30.9±0.2
	OPA	**41.6**±0.1	**23.1**±0.5	**47.2**±0.7	**29.6**±0.8	**50.8**±1.0	**31.5**±0.6
	Gain (%)	2.6↑	2.0↑	1.7↑	0.8↑	1.1↑	0.6↑

GTX 2080-Ti GPU. For fair comparisons, we follow the procedure in [29] to use the student model for inference, along with a post-processing step on final predictions.

4.1 Experimental Results

4.1.1 Main Results

Table 2 shows the result of our method on ScanNet and SUN RGB-D, under different labeled data ratios compared with state-of-the-art methods for 3D object detection in SSL, including VoteNet [16], SESS [36], and 3DIoUMatch [29]. The proposed OPA method consistently performs favorably against existing approaches in all the settings. Moreover, our method performs better in settings with lower labeled data ratios, e.g., SUN RGB-D 5% and ScanNet 10%, which shows the advantage of the proposed augmentor. Note that, since the total number of scenes in ScanNet is five times less than the one in SUN RGB-D, we find that the performance gain of 5% ScanNet is slightly less than the 10% ScanNet setting, which can be caused by the less data to train the augmentor. More results and analysis are provided in the supplementary material.

4.1.2 Ablation Study

Augmentation on Labeled and Unlabeled Data. We first study the effect of our augmentor trained on labeled or unlabeled data. In Table 3, comparing to ID (5) using our augmentor on both labeled and unlabeled data (i.e., our full model), we show the benefit by removing labeled or unlabeled data in ID (2) and (3), respectively. Moreover, comparing ID (1) with ID (2) and ID (3), where we include unlabeled and labeled data in our proposed augmentor with $\hat{y}_o$, the performance gains (ScanNet 10% mAP@0.5) are 3.1% and 3.4%, respectively. In ID (2), the augmentor helps unlabeled data to produce better data variance for student model training. In ID (3), the augmentor provides more diverse supervised samples in the pre-training stage. This shows that our augmentor can take advantage of different data and improve performance. Note that, experiments are conducted in one of the same data splits for fair comparisons, and thus the numbers of our full model are slightly different from the averaged numbers in Table 2.

Table 3: We study the affect of proposed components in our augmentor in settings of ScanNet 10% and SUN RGB-D 5% labeled data ratio.

ID	Aug. (labeled)	Aug. (unlabeled)	$\hat{y}_o$ in (3)	ScanNet 10% mAP @0.25	ScanNet 10% mAP @0.5	SUN RGB-D 5% mAP @0.25	SUN RGB-D 5% mAP @0.5
(1)				47.1	28.3	38.1	21.3
(2)		√	√	50.4	31.4	40.1	23.2
(3)	√		√	50.4	31.7	40.1	22.5
(4)	√	√		48.5	29.3	38.1	22.1
(5)	√	√	√	50.7	32.4	41.8	23.5

Table 4: Sensitivity analysis of λ in (2) on ScanNet 10% labeled data ratio.

λ in (2)	ScanNet 10% mAP@0.25	ScanNet 10% mAP@0.5
0.01	49.8	32.1
0.05	50.5	31.4
0.1	50.7	32.4
0.5	50.1	31.7
1.0	48.9	29.5

Objectness Term $\hat{y}_o$ in (3). Different from PointAugment [17], we introduce an objectness term in (3) that controls the magnitude of augmentation, so that the augmentor is aware of the quality of class-agnostic detection results and learns how to generate appropriate augmentations with challenging variations. In Table 3, ID (4) without using this objectness term performs worse than our full model in ID (5), which indicates that this term is essential to generate augmentations that are helpful in our SSL setting.

Sensitivity on λ in (2). In Table 4, we test the sensitivity on λ in (2) when training the augmentor using 10% labeled data on ScanNet. The higher lambda values (e.g., 1.0) accelerate the training processing of our augmentor to become more aggressive (i.e., generating more challenging samples), which may harm the stability in the early training stage, thus leading to worse performance. On the other hand, the lower lambda values control the pace for training the augmentor in an appropriate step, stabilizing the training and leading to better performance. Overall, Table 4 shows that our method is robust to the λ value when it is in a reasonable range (e.g., from 0.01 to 0.5). In all the experiments, we choose $\lambda = 0.1$.

5 Conclusions

In this paper, we propose OPA, a novel teacher-student mutual learning framework with object-level augmentor, which benefits semi-supervised learning on both labeled and unlabeled data for 3D object detection. We show that the existing methods using only global transformations is sub-optimal, and thus we propose to adopt both global and local augmentations. To this end, we propose to learn an object-level augmentor that is jointly trained with the object detector in an adversarial learning manner, in which the objectness score from the detector provides the guidance to the augmentor. In this way, our object-level augmentor is able to increase the variance within object points and thus boost the detector's capability in SSL. We conduct extensive experiments on the ScanNet and SUN RGB-D benchmarks, in which OPA achieves consistent performance gains against state-of-the-art approaches on all the settings with different ratios of labeled data.

Acknowledgment. This work was supported in part by National Science and Technology Council (NSTC) under grants 111-2628-E-A49-025-MY3, 109-2221-E-009-113-MY3, 110-2634-F-006-022, 110-2634-F-002-050, and 111-2634-F-007-002. This work was funded in part by Qualcomm and MediaTek.

References

[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[2] Yunlu Chen, Vincent Tao Hu, Efstratios Gavves, Thomas Mensink, Pascal Mettes, Pengwan Yang, and Cees GM Snoek. Pointmixup: Augmentation for point clouds. In *European Conference on Computer Vision*, pages 330–345, 2020.

[3] Shuyang Cheng, Zhaoqi Leng, Ekin Dogus Cubuk, Barret Zoph, Chunyan Bai, Jiquan Ngiam, Yang Song, Benjamin Caine, Vijay Vasudevan, Congcong Li, et al. Improving 3d object detection through progressive population based augmentation. In *European Conference on Computer Vision*, pages 279–294, 2020.

[4] Jaeseok Choi, Yeji Song, and Nojun Kwak. Part-aware data augmentation for 3d object detection in point cloud. *arXiv preprint arXiv:2007.13373*, 2020.

[5] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019.

[6] 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*, 2017.

[7] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in Neural Information Processing Systems*, 2019.

[8] Sihyeon Kim, Sanghyeok Lee, Dasol Hwang, Jaewon Lee, Seong Jae Hwang, and Hyunwoo J. Kim. Point cloud augmentation with weighted local transformations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 548–557, 2021.

[9] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016.

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

[11] Hengduo Li, Zuxuan Wu, Abhinav Shrivastava, and Larry S Davis. Rethinking pseudo labels for semi-supervised object detection. *arXiv preprint arXiv:2106.00168*, 3(5), 2021.

[12] Ruihui Li, Xianzhi Li, Pheng-Ann Heng, and Chi-Wing Fu. Pointaugment: an auto-augmentation framework for point cloud classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6378–6387, 2020.

[13] Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, and Boqing Gong. Improving object detection with selective self-supervised self-training. In *European Conference on Computer Vision*, pages 589–607, 2020.

[14] 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. *arXiv preprint arXiv:2102.09480*, 2021.

[15] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017.

[16] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE International Conference on Computer Vision*, pages 9277–9286, 2019.

[17] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in Neural Information Processing Systems*, 30, 2017.

[18] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in Neural Information Processing Systems*, 29, 2016.

[19] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–779, 2019.

[20] 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 Conference on Computer Vision and Pattern Recognition*, pages 10529–10538, 2020.

[21] 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. *IEEE transactions on pattern analysis and machine intelligence*, 43(8):2647–2664, 2020.

[22] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems*, 2020.

[23] 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.

[24] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[25] Peng Tang, Chetan Ramaiah, Yan Wang, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 2291–2301, 2021.

[26] Yihe Tang, Weifeng Chen, Yijun Luo, and Yuting Zhang. Humble teachers teach better students for semi-supervised object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3132–3141, 2021.

[27] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, 2017.

[28] Ardian Umam, Cheng-Kun Yang, Yung-Yu Chuang, Jen-Hui Chuang, and Yen-Yu Lin. Point mixswap: Attentional point cloud mixing via swapping matched structural divisions. 2022.

[29] He Wang, Yezhen Cong, Or Litany, Yue Gao, and Leonidas J Guibas. 3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 14615–14624, 2021.

[30] Keze Wang, Xiaopeng Yan, Dongyu Zhang, Lei Zhang, and Liang Lin. Towards human-machine cooperation: Self-supervised sample mining for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[31] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33:6256–6268, 2020.

[32] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 3060–3069, 2021.

[33] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10), 2018. ISSN 1424-8220.

[34] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11040–11048, 2020.

[35] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. Mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[36] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Sess: Self-ensembling semi-supervised 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11079–11087, 2020.

[37] Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, and Hao Li. Instant-teaching: An end-to-end semi-supervised object detection framework. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4081–4090, 2021.

[38] 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.