Class-Level Confidence Based 3D Semi-Supervised Learning

Zhimin Chen[1], Longlong Jing[2], Liang Yang[2], Yingwei Li[3], and Bing Li[1]

[1]Clemson University
[2]The City University of New York
[3]Johns Hopkins University

{zhiminc,bli4}@clemson.edu, ljing@gradcenter.cuny.edu, lyang1@ccny.cuny.edu,
yingwei.li@jhu.edu

Abstract

Recent state-of-the-art method FlexMatch firstly demonstrated that correctly estimating learning status is crucial for semi-supervised learning (SSL). However, the estimation method proposed by FlexMatch does not take into account imbalanced data, which is the common case for 3D semi-supervised learning. To address this problem, we practically demonstrate that unlabeled data class-level confidence can represent the learning status in the 3D imbalanced dataset. Based on this finding, we present a novel class-level confidence based 3D SSL method. Firstly, a dynamic thresholding strategy is proposed to utilize more unlabeled data, especially for low learning status classes. Then, a re-sampling strategy is designed to avoid biasing toward high learning status classes, which dynamically changes the sampling probability of each class. To show the effectiveness of our method in 3D SSL tasks, we conduct extensive experiments on 3D SSL classification and detection tasks. Our method significantly outperforms state-of-the-art counterparts for both 3D SSL classification and detection tasks in all datasets.

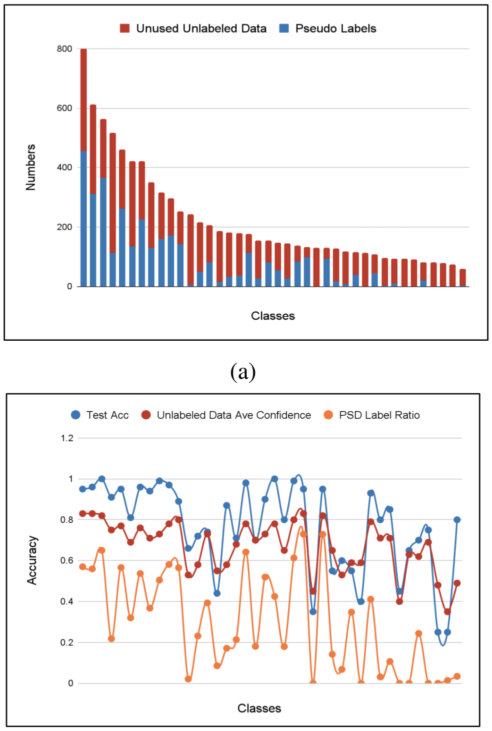
1. Introduction

As 3D point cloud data collection and annotation is expensive and time-consuming, 3D semi-supervised learning (SSL) has attracted increasing attention in recent years and shows its superiority in utilizing the unlabeled data [46, 31, 4, 25, 43]. Most existing semi-supervised learning methods, such as Pseudo-Labeling [12] and FixMatch [25], employ the pseudo-labeling strategy in which the network's high confidence predictions on the unlabeled data are used as labels to further optimize the network.

The pseudo-labeling strategy based methods were widely used and achieve significant improvement in performance for many different tasks [46, 31, 4, 25]. However, a non-negligible drawback of pseudo labeling is that it relies on a manually pre-defined fixed threshold to choose high-quality pseudo labels. For each data from the unlabeled data, no matter its category, the data will be used for training only when its confidence is higher than this fixed threshold; otherwise, this data will be ignored. The common practice uses a very high threshold (0.9) [31, 4] in 3D tasks to keep the pseudo-labels with high quality. This fixed threshold ignores the different learning statuses among classes and thus left a large number of unlabeled data unused, which compromises the model performance.

To solve this issue, an intuitive way is to estimate the learning status of each class and set dynamic threshold for each class accordingly. However, the key challenge is how to estimate the learning status. Recently SOTA method FlexMatch [43] leveraged curriculum learning approach to estimate learning status of each class and flexibly adjust thresholds. However, the FlexMatch does not well-define the terminology 'learning status'. In this work, we regard 'learning status' as how well the model learning for a class and can be reflected in the test accuracy. The test accuracy of each class is utilized to represent the learning status in our analysis. Furthermore, in FlexMatch, only learning difficulty is taken into account to estimate learning status but the imbalanced data condition is not included. We find that both learning difficulty and imbalanced data affect the learning status of network. For example, the network may have similar performance on high learning difficulty but majority classes and low learning difficulty but minority classes.

Hence, a more accurate and general estimation method of learning status is required for semi-supervised learning. Previous work [12, 25] utilizes instance-level confidence



(a)

(b)

Figure 1: The results analysis of FixMatch trained in ModelNet40 dataset with 10% labeled data. (a) The ratio of selected pseudo-labels to unlabeled data. (b) Test accuracy and the class-level confidence of unlabeled data. It is obvious to observe that (1) Only a small percentage of unlabeled data is used during training with high threshold setting. (2) Learning difficulty of each class classes has high variance. Some minority classes have higher accuracy than majority classes. (3) The class-Level confidence has high correlation with test accuracy of each class.

to represent the instance learning status even in imbalanced data and can be utilized to select well-learned pseudo labels. Therefore, it is intuitive to assume that the class-level learning status can be reflected by the class-level confidence. Furthermore, our analysis in Fig. 1b demonstrate that there is a high correlation between the average class-level confidence score and the test accuracy results in 3D imbalanced data. More analysis on detection task can be found in the supplementary materials. Hence, we hypothesize that the class-level confidence on the unlabeled data can be leveraged to estimate the learning status of each class in 3D imbalanced data.

Inspired by our intuitions and the analysis above, we propose a semi-supervised learning method that can dynamically adjust the threshold based on the learning status. The dynamic threshold is adjusted based on the learning status of each class, which allows more unlabeled data of low learning status to be utilized. Furthermore, our method can also utilize more unlabeled data at the early stage of the training process where only few data have predicted confidence larger than the fixed high thresholds. However, we find that improving numbers of selected pseudo-label with dynamic threshold cannot eliminate the learning status variance caused by data imbalance and learning difficulty variance. This makes network biased toward high learning status classes and thus overfitted. To avoid this issue, we further propose a novel re-sampling strategy to dynamically sample the data based on the learning status. Specifically, our re-sampling strategy increases the sample probability of instances belonging to low learning status classes and decreases the sample probability of high learning status classes. With the dynamical thresholding and re-sampling, our method can utilize the unlabeled data more effectively and balance the learning status of each class .

The goal of our method is to estimate the learning status of each class and further balance and improve the learning status. Compared to the other 3D semi-supervised learning methods, our method estimate the learning status of each class, dynamically adjust the threshold, and re-sampling the data based on the learning status. Our method can be easily applied to different semi-supervised tasks to improve performance even in imbalanced dataset. To demonstrate the generality of the proposed method, we evaluated our proposed method on two different tasks, including SSL 3D object recognition and SSL 3D object detection tasks. Our method outperforms the state-of-the-art methods by a large margin.

Our key contributions are summarized as follows:

1. We clarify the definition of learning status and practically demonstrate that the class-level confidence can represent the learning status of each class. Based on this finding, we propose a learning status estimation method that works well in the imbalanced 3D dataset.

2. We firstly incorporate learning difficult and imbalanced data problems together based on learning status. We propose a novel 3D semi-supervised learning method to dynamically adjust the thresholds and re-sample data based on each class learning status, which balances and improve learning status and thus solves the variance of learning difficult and imbalanced data problems at the same time.

3. Our proposed method outperforms the state-of-the-art semi-supervised 3D object detection and classification methods by a large margin.

2. Related Work

Semi-Supervised Learning: Semi-supervised learning methods have made a significant progress in recently years [2, 3, 6, 12, 16, 28, 17, 38]. Many existing SSL methods utilize pseudo-labeling [12] to minimizes the entropy of the predictions on unlabeled data. The performance of pseudo labeling heavily relies on the quality of pseudo-labels. To improve the quality of pseudo-labels, state-of-the-art SSL method FixMatch [25] and many FixMatch-liked methods usually set a fixed high confidence threshold to filter out low-confidence predictions from the strong augmented data. The high-value threshold can improve the quality of pseudo labels but ignores the learning status of each class, which not only engenders the bias toward high learning status classes but leaves a large number of unlabeled data unused. To address this issue, Dash [38] uses cross entropy loss to obtain dynamic threshold for all classes. FlexMatch [43] substitutes the pre-defined threshold with flexible thresholds based on curriculum learning to consider each class learning status. However, our experiments demonstrate that the FlexMatch does not generalize well in 3D dataset. This is because it is designed for class-balanced datasets, but the 3D dataset is inherently imbalanced. Furthermore, in 3D networks, the prediction confidence of some classes cannot achieve the pre-defined high threshold and make those classes thresholds extremely small. Therefore, the learning status estimation strategy in FlexMatch is not suitable to be applied in 3D SSL tasks. More comparison with FlexMatch can be found in supplemental materials.

Class-Imbalanced Semi-Supervised Learning: Recently, class imbalanced semi-supervised learning has attracted increasing attention as it more accurately describes the real-world data distribution [40, 9]. Wei et al [33] found that the raw SSL methods usually have high recall and low precision for head classes and proposed CReST to re-sampling unlabeled data based on the labeled data numbers. Recent state-of-the-art method BiS [8] deployed two different re-sampling strategies at the same time to decoupled train the model. All of those class-imbalanced SSL methods resample based on data numbers. However, we find that some minority classes may have better performance than majority classes due to their low learning difficulty. Sampling based on data numbers makes the model biased toward those low learning difficulty classes. Due to the page limitation, we add the comparison with state-of-the-art imbalanced SSL methods in supplemental materials.

3D Semi-Supervised Object Classification: Currently, many 3D object classification method have been proposed for 3D understanding [20, 21, 29, 34, 37, 44, 45, 36, 36]. Unlike 2D features, a 3D model is complex by nature and thus hard to extract. To better extract 3D features, many works [14, 20, 21, 32] have been proposed to extract features from 3D point clouds for the 3D classification tasks. However, 3D SSL classification is underexploited. Chen et al. [4] finds that directly implementing 2D SSL methods like FixMatch [25], S4L [42], and Pseudo label [12] cannot achieve comparable results in 2D and propose to utilize multi-modality information in 3D SSL classification. However, what downgrades those 2D SSL methods' performance in 3D is left to explore.

3D Semi-Supervised Object Detection: According to the input data formats, current methods for 3D object detection task can be summarized into three different types: 2D projection [13, 24, 39], voxel grid [11, 22, 27], and point cloud [10, 18, 23, 19]. Although those methods have achieved impressive results, the high-quality 3D ground truths are expensive and time-consuming to collect. Due to the capacity of alleviating the dependency on labeled data, semi-supervised 3D object detection has drawn wide attention from researchers [46, 31]. However, current 3D SSL detection works are all FixMatch-like and utilize a fixed threshold to select pseudo labels. None of them takes into account the variance of learning difficulty and class-imbalance situation, which leads to the sub-optimal results. In this work, a general class-level confidence based 3D SSL method is proposed to dynamically set the thresholds and re-sample the data according to the learning status.

3. Method

The overview of our proposed method is shown in Fig 2. It contains three main parts: (1) learning status estimation, which is obtained by the class-level confidence based on the model predictions on the unlabeled data, (2) dynamic thresholding based on the learning status of each lass, and (3) dynamically re-sampling for each class based on the learning status. The formulation for each component is introduced in the following sections.

3.1. Problem Formulation

The goal of the 3D semi-supervised learning is to jointly train the model based on limited labeled samples and a large number of unlabeled samples. Let $X_L = (x_i, y_i)_{i=1}^{N_L}$ be a limited labeled dataset with N_l samples, where $x_i \in \mathbb{R}^{N\times 3}$ is a 3D point cloud representation of an object or a scene, and y_i is the corresponding label. Let $X_U = (x_i)_{i=1}^{N_U}$ be an unlabeled set with N_U samples, which does not contain labels. Our model is trained on both X_L and X_U for semi-supervised learning with the proposed class-level confidence based dynamic threshold and re-sampling strategy.

3.2. Baseline: FixMatch

Consistency regularization is a commonly used constraint in recent SSL algorithms [2, 1, 28]. It forces prediction results from different augmentation of same instance being consistent:

$$\sum_{b=1}^{\mu B} ||p_m(y|\alpha(u_b)) - p_m(y|\alpha(u_b))||_2^2 \tag{1}$$

, where B represents the batch size of labeled data, μ is the labeled data and unlabeled data ratio, α is a stochastic data

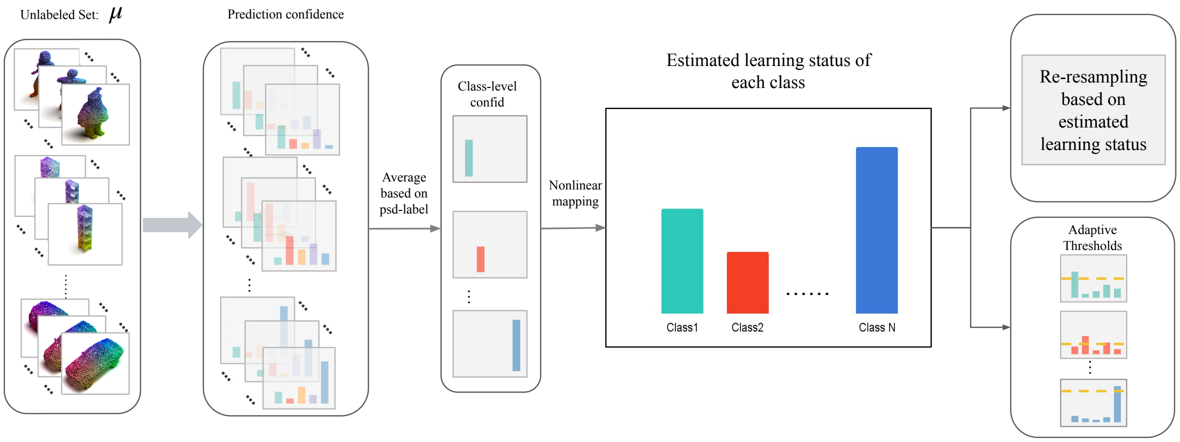


Figure 2: **An overview of our proposed method**. Our model consists of three main parts: (1) Obtaining the learning status for each class through the class-level confidence from unlabeled data, (2) Leveraging learning status to dynamically adjusting the threshold of each class , and (3) Re-sampling the dataset based on the learning status.

augmentation function, u_b is the unlabeled data from X_U, and p_m is the output probability of the model. Another popular method in SSL is pseudo-labeling, which obtain pseudo labels from unlabeled data prediction results. It defines a fixed threshold to cut off high-confidence unlabeled data to render pseudo labels. The cross-entropy loss is utilized to minimize the difference of the predictions and hard pseudo-labels:

$$\frac{1}{\mu B}\sum_{b=1}^{\mu B}\mathbb{1}(max(p_m(y|\alpha(u_b)) \geq \tau) \\ \cdot H(\hat{p}_m(y|\alpha(u_b), p_m(y|\alpha(u_b)) \tag{2}$$

, where $\hat{p}_m(y|\alpha(u_b) = \arg\max(p_m(y|\alpha(u_b))$ and τ is the threshold. $H(p, q)$ represents the cross entropy loss between p and q. Usually, a high threshold τ will be used to filter out low quality pseudo labels that have low prediction confidence.

Recently, FixMatch [25] combines consistency regularization and pseudo-labeling together and achieved state-of-the-art performance on many tasks [31, 46, 4]. FixMatch contains a supervised loss ℓ_s and a unsupervised loss ℓ_u. The supervised loss is defined as:

$$\ell_s = \frac{1}{B}\sum_{b=1}^{B} H(y_b, p_m(y|\alpha(x_b)) \tag{3}$$

, where y_b is the label of labeled instance x_b. For unsupervised loss ℓ_u, FixMatch chooses the confident predictions (larger than the threshold) from weak augmented data as pseudo labels. Then, a cross-entropy loss is minimized based on the network prediction from the strong augmented views of the data and this pseudo label. The unsupervised loss is formulated as:

$$\ell_u = \frac{1}{\mu B}\sum_{b=1}^{\mu B}\mathbb{1}(max(p_m(y|\alpha(u_b)) \geq \tau) \\ \cdot H(\hat{p}_m(y|\alpha(u_b), p_m(y|\mathcal{A}(u_b)) \tag{4}$$

Where $\mathcal{A}$ is a strong augmentation function. Due to the simplicity and high performance of FixMatch, currently, most SSL methods are FixMatch-liked for many different tasks. However, in FixMatch and FixMatch-liked methods, the threshold τ is normally a high constant value. Although such a high threshold can improve the quality of pseudo labels, it will decrease the number of pseudo-labels that are actually used to optimize the network and leaves a large number of unlabeled data unused.

3.3. Class-Level Confidence Based Dynamic Threshold

Inspired by FlexMatch [43], we proposed a dynamic threshold based on the class-level confidence to leverage the learning status of each class. In existing SSL methods, instance-level prediction confidence is leveraged to evaluate the quality of the instance. However, class-level confidence remains to be un-utilized. Our analysis shows that for each class, their class-level confidence on the unlabeled data can be leveraged to represent the learning status. We can use the estimated learning status to dynamically adjust each class threshold and thus boost the effectiveness of SSL. For each class, we obtain its unlabeled set from prediction probability $argmax$: $\{C_c|u_b \in C_c, \arg\max(p_m(y|\alpha(u_b))) = c, b = 1, 2, ..., \mu B\}$. Then, each unlabeled class set is averaged to obtain class-level confidence.

$$P_c = \frac{1}{|C_c|}\sum_{j=0}^{|C_c|} \max(p_m(y|\alpha(u_j)), u_j \in C_c \qquad (5)$$

Then, a non-linear function is utilized to map class-level confidence to the learning status. The entire process to obtain the dynamic formulated as:

$$\tau_e(c) = \begin{cases} 1-\tau, & if\ M(P_c) < 1-\tau \\ \tau, & elif\ M(P_c) > \tau \\ M(P_c), & else \end{cases} \qquad (6)$$

where P_c is the class-level confidence for class c, $M(x) = \frac{x}{2-x}$ is a nonlinear mapping function, $\tau_e(c)$ is the dynamic threshold for class c at epoch e. As there are no labels for unlabeled data, for each unlabeled instance u_i, the $argmax$ of the network prediction is utilized as its class: $c_i = \arg\max(p_m(y|\alpha(u_i)))$. The high-quality pseudo-labels are filtered by dynamic thresholds:

$$\hat{y}_c^i = \mathbb{1}\left[p_i \geq \tau_e(c_i)\right] \qquad (7)$$

, where $p_i = \max(p_m(y|\alpha(u_i)))$. The threshold for low learning statuses classes will be reduced to increase the pseudo label numbers and improve the learning status. As our proposed method adjusts thresholds method only based on class-level confidence, it can be applied to any kind of dataset and have higher generalization ability.The unsupervised loss of proposed dynamic thresholds method is formulated as:

$$\ell_{u,e} = \frac{1}{\mu B}\sum_{b=1}^{\mu B} H(\hat{y}_c^i, p_m(y|\mathcal{A}(u_b)) \qquad (8)$$

3.4. Class-Level Confidence based Data Re-sampling

While our proposed dynamic threshold can increase the pseudo-label numbers of low learning status classes, it still cannot completely balance each class's learning status due to data imbalance and the variance of learning difficulty. For example: in ModelNet40, the label numbers of airplane and bowl are 563 and 59 separately. Even if the dynamic threshold filters half pseudo-labels of airplane and utilizes all pseudo-labels of bowls, the airplane's selected unlabeled data numbers are at least four times larger than the bowl's selected unlabeled data numbers. Furthermore, the airplane has lower learning difficulty than the bowl due to its unique shape. Hence, even with dynamic threshold, classes like airplane still have high learning and tend to be overfitted due to their abundant data numbers, low learning difficulty, or both. This compromises the network performance.

To alleviate this problem, we resort to the re-sampling strategy which has been proved to be effective in SSL class-imbalanced datasets. Most of current SSL imblanced method samples based on data numbers [33, 8]. However, we find that some minority classes may have better performance than majority classes due to their low learning difficulty. Sampling based on data numbers makes the model biased toward those low learning difficulty classes. Hence, we propose a class-level confidence based re-sampling strategy to directly increase the sampling probability of low learning status classes, which takes into account both learning difficulty and data imbalance. The sample probability for each class is formulated as:

$$\begin{cases} 1 - W(e)\cdot P_c \cdot p_i & ,if\ P_c > \tau\ , \\ 2 - W(e)\cdot P_c \cdot p_i & ,if\ P_c \leq \tau\ , \end{cases} \qquad (9)$$

, where the warm-function $W(e) = exp(-5 \times (1 - e/E_{max})^2)$ following previous works [41, 15] to avoid aggressively sampling. p_i is the prediction confidence for instance i, c is the prediction class for instance i, P_c is the class-level confidence for class c, e is the current epoch, and E_{max} is the max epoch.

3.5. Final Objective Function

The core idea of our method is to dynamically adjust the pseudo-label threshold and re-sample the data based on learning status. Therefore, the proposed method can be easily applied to other pseudo-label based methods. Our entire model is jointly trained with two loss functions, including supervised loss ℓ_s on the labeled data and the $\ell_{u,e}$ on the unlabeled data as:

$$\ell = \lambda_s\ell_s + \lambda_u\ell_{u,e} \qquad (10)$$

,where λ_s and λ_u are weights for labeled and unlabeled losses.

4. Experimental Results

4.1. Datasets

Classification Datasets: Following the state-of-the-art SSL 3D object classification methods, we evaluate our method on two benchmarks including ScanObjectNN [30] and ModelNet40 [35]. The ModleNet40 is a widely used benchmark and it contains $12,311$ meshed CAD models from 40 categories, and there are $9,843$ models in training and $2,468$ in testing. The ScanObjectNN is a more realistic point cloud object dataset. It contains $15,000$ objects belongs to 15 classes that sampled from $2,902$ unique object instances from real world.

Detection Datasets: Following the previous state-of-the-art SSL 3D object detection methods [31], we evaluate our method on two widely used detection benchmark

Dataset	Method	2%		5%		10%	
		Overall Acc	Mean Acc	Overall Acc	Mean Acc	Overall Acc	Mean Acc
ModelNet40	Point Transformer[7]	71.1	61.0	77.1	69.2	84.6	77.2
	PL[12]	69.7	59.6	78.3	69.0	85.1	77.7
	Flex-PL[43]	66.7	54.9	74.2	62.3	83.2	70.3
	Confid-PL(Ours)	**74.4**	**61.9**	**80.6**	**73.5**	**86.5**	**80.4**
	FixMatch[25](NeurIPS 2020)	70.8	62.7	78.9	71.1	85.5	79.4
	Dash[38](ICML 2021)	71.5	63.0	79.7	71.8	85.9	80.1
	FlexMatch[43](NeurIPS 2021)	70.1	61.2	80.5	70.4	86.2	78.7
	Confid-Match(Ours)	**73.8**	**64.1**	**82.1**	**74.3**	**87.8**	**82.5**

Table 1: Comparison with state-of-the-art methods. The results on the ModelNet40 dataest for 3D semi-supervised object classification task.

Dataset	Method	1%		2%		5%	
		Overall Acc	Mean Acc	Overall Acc	Mean Acc	Overall Acc	Mean Acc
ScanObjectNN	Point Transformer[7]	32.1	26.1	44.7	36.5	56.6	50.0
	PL[12]	31.2	25.8	47.5	38.6	58.1	51.5
	Flex-PL[43]	29.2	24.2	47.2	37.6	60.1	51.9
	Confid-PL(Ours)	**32.6**	**27.1**	**48.8**	**41.5**	**63.1**	**55.2**
	FixMatch[25](NeurIPS 2020)	33.5	27.6	47.4	39.9	59.4	52.4
	Dash[38](ICML 2021)	35.1	29.3	50.3	44.1	62.8	60.3
	FlexMatch[43](NeurIPS 2021)	34.2	26.2	48.5	39.7	63.4	57.2
	Confid-Match(Ours)	**38.2**	**32.7**	**57.0**	**48.6**	**69.4**	**65.5**

Table 2: Comparison with state-of-the-art methods. The results on the ScanObjectNN dataest for 3D semi-supervised object classification task.

including SUN RGB-D [26] and ScanNet [5]. The ScanNet [5] is an indoor scene dataset consisting of 1513 reconstructed meshes, among which 1201 are training samples, and 312 are validation samples. SUN RGB-D [26] contains more than $10,000$ indoor scenes while 5285 for training and 5050 for validation.

4.2. Implementation Details

Semi-Supervised 3D Object Classification: On the ModelNet40 and ScanObjectNN datasets, we use SGD optimizer with a learning rate of 0.01, and the learned rate is scheduled with CosineAnnealingLR decay with a minimum learning rate of 0.0001. All the models are optimized with a total epoch of 500. The weak augmentation contains rotation and random scale, and the strong augmentation leverages random scale, translation, jittering, and rotation. The batch size is set to 240, while 48 of them are labeled data and the rest are unlabeled. Weights of supervised loss and unsupervised loss are both 1. The threshold is set to be $\tau = 0.8$. The re-sampling strategy updates the dataloader every 50 epoch. The PointTransformer [7] is utilized as the backbone for the SSL classification task.

Semi-Supervised 3D Object Detection: We apply our method on state-of-the-art work 3DIoUMatch and follow the same setting. Unlike 3DIoUMatch uses VoteNet pre-trained model, we utilize the proposed re-sampling strategy to re-sample the labeled data in the pre-training process. Then the pre-trained weights are utilized to initialize the student and teacher networks. For those multiple objects sceneries, the object that has minimal confidence is leveraged in the re-sampling process. Like classification task, the re-sampling strategy updates the dataloader every 50 epoch. For a fair comparison, the pre-processing data methods and labels are the same as previous works [18, 31] and mean average precision (mAP) under IoU thresholds of 0.25 and 0.5 are utilized as evaluation metrics. The threshold is also set to be $\tau = 0.8$.

4.3. Performance on Semi-Supervised 3D Object Classification

To demonstrate the capability and potential of our proposed method, we compare the performance of our method with other semi-supervised learning methods including Pseudo-Labeling (PL) [12], FixMatch [25], Dash [38], and FlexMatch [43]) under the same setting in ModelNet40 [35] and ScanObjectNN [30]. The FlexMatch [43] is the most recent state-of-the-art method that was proposed to overcome the fixed-threshold drawback of FixMatch by proposing curriculum pseudo labeling to dynamically adjust the threshold. For a fair comparison, all the methods use the same backbone, data augmentation, and hyper-parameters.

To extensively compare with the state-of-the-art methods, we apply our method in FixMatch and Pseudo-labeling and report the results for all the methods under different percentages of labeled data. Two evaluation metrics are used to indicate the performance, including overall accuracy and

Dataset	Method	1%		2%		5%	
		mAP @ 0.25	mAP @0.50	mAP @0.25	mAP @0.50	mAP @0.25	mAP @0.50
SUN RGB-D	VoteNet [18](ICCV 2019)	16.7±1.2	3.9±0.9	21.8±1.6	5.1±0. 8	33.9±1.9	13.1±1.7
	SESS [46](CVPR 2020)	19.9±1.6	6.3±1.2	23.3±1.1	7.9±0.8	36.1±1.1	16.9±0.9
	3DIoUMatch [31] (CVPRR 2021)	25.6±0.6	9.4±0.7	26.8±0.7	10.6±0.5	39.7±0.9	20.6±0.7
	Confid-3DIoUMatch(Ours)	**27.8±0.8**	**11.3±0.6**	**32.7±0.3**	**13.5±0.4**	**43.1±0.6**	**24.2±0.5**

Table 3: Comparative studies with state-of-the-art methods on the SUN RGB-D dataest for 3D SSL object detection.

Dataset	Method	1%		2%		5%	
		mAP @ 0.25	mAP @0.50	mAP @0.25	mAP @0.50	mAP @0.25	mAP @0.50
ScanNet	VoteNet [18](ICCV 2019)	8.9 ± 1.1	1.5 ± 0.5	16.9 ± 1.3	4.7 ± 0.8	31.2 ± 1.1	14.7 ± 0.7
	SESS [46](CVPR 2020)	11.3 ± 1.6	2.7 ± 0.6	21.1 ± 1.5	8.4 ± 1.1	35.5 ± 2.0	17.2 ± 0.9
	3DIoUMatch [31] (CVPRR 2021)	14.6 ± 1.4	3.9 ± 0.5	24.5 ± 1.9	11.2 ± 1.4	40.4 ± 0.8	21.0 ± 0.6
	Confid-3DIoUMatch(Ours)	**19.0±0.4**	**6.4±0.4**	**29.5±1.5**	**15.2±0.6**	**43.6±0.5**	**24.3±0.4**

Table 4: Comparative studies with state-of-the-art methods on the ScanNet dataest for 3D SSL object detection.

class mean accuracy, which is the average of the accuracy of all the classes. The Table 1 and Table 2 indicate that in the 3D SSL clasification task, the current state-of-the-art work FlexMatch only has limited improvement or even **decreases** the performance when applied in FixMatch and Pseudo-Labeling with limited labeled data. This is because FlexMatch is only designed for class balanced dataset but 3D datasets are all inherently data-imbalanced. As shown in Table 1, our model outperforms all the state-of-the-art methods with two evaluation metrics on the ModelNet40 dataset. As shown in Table 2, on more realistic and challenging dataset ScanObjectNN [30], our model also significantly outperforms all the other methods by a large margin. For both ModelNet40 [35] and ScanObjectNN [30] dataset, the improvement of our model on the mean class accuracy is more significant, which demonstrates the effectiveness of our model on balancing the learning status of each class.

4.4. Performance on Semi-Supervised 3D Object Detection

To demonstrate the generalization ability of our method, we further evaluated our proposed method on the semi-supervised 3D object detection benchmark and compared it with the state-of-the-art methods. We extend the state-of-the-art method 3DIoUMatch [31] and apply our class-based thresholding and re-sampling during training. We report the performance comparison with the state-of-the-art methods including VoteNet [18], SESS [46] and 3DIoUMatch [31] on two benchmark including SUN RGB-D and ScanNet datasets. Following the convention, the mean average precision (mAP) under two different thresholds, including 0.25 and 0.5 are reported.

As shown in Table 3 and 4, our model significantly outperforms all the other state-of-the-art methods on both SUN RGB-D and ScanNet datasets with different settings. The most significant improvement is under 2% labeled setting in which our method outperforms 3DIoUMatch by **5.9** and **5.0** on mAP@0.25 on ScanNet and SUN RGB-D, respectively. The results on those two benchmarks demonstrate that our proposed confidence-based dynamic threshold and learning status balance re-sampling strategy can be easily integrated into other semi-supervised methods to significantly boost the performance.

4.5. Ablation Study for Dynamic Threshold and Re-sampling

Our proposed method contains two major components: confidence-based dynamic threshold and dynamic re-sampling strategy. To analyze the effect of each component, we conduct ablation studies about the combinations of different components on both SSL 3D object classification and detection tasks. The FixMatch [25] is used as baseline for classification task while 3DIoUMatch [31] is used as baseline for detection task. The Table 5 contains the results for classification while Table 6 is for detection task.

Dynamic Threshold: For both the classification and detection tasks, the baseline uses the fixed threshold to select unlabeled data with high-quality predictions. After applying our class-level dynamic threshold strategy to the two different tasks, the performance is improved for both tasks under different settings. The improvement for some settings is huge, i.e. the mean accuracy for ScanObjectNN dataset is improved by 10.1% under the 5% labeled set. These results confirm the effectiveness of our class-level dynamic threshold and show that our method can be easily integrated into other semi-supervised learning methods.

Dynamic Re-sampling: For both the classification and detection, the baselines do not use any re-sampling strategy, therefore, the sampling probability for each data sample are same. For the SSL 3D classification task, all the perfor-

Confidence Re-sample	Dynamic Threshold	ModelNet40 10%		ObjectNN 5%	
		Overall Acc	Mean Acc	Overall Acc	Mean Acc
		85.5	79.4	59.4	52.4
✓		86.7	81.1	64.1	60.5
	✓	86.9	81.7	66.9	62.5
✓	✓	**87.8**	**82.5**	**69.4**	**65.5**

Table 5: Ablation study for components effect on the 3D SSL object classification task in ModelNet40 and ScanObjctNN dataset.

Re-sample Pre-train	Confidence Re-sample	Dynamic Threshold	ScanNet 5%		SUN-RGBD 2%	
			mAP @0.25	mAP @0.5	mAP @0.25	mAP @0.5
			40.4±0.8	21.0±0.6	26.8±1.1	10.6±0.5
✓			41.8±0.6	22.7±0.5	31.0±0.7	11.7±0.6
✓		✓	42.1±0.8	23.2±0.5	31.7±0.8	12.4±0.5
✓	✓		42.4±0.4	23.0±0.6	31.9±0.9	11.9±0.7
✓	✓	✓	**43.6±0.5**	**24.3±0.4**	**32.7±0.3**	**13.5±0.4**

Table 6: Ablation study for components effect on the 3D SSL object detection task in ScanNet and Sun RGB-D dataset.

mances are improved after applying our proposed dynamic re-sampling strategy, and the improvement on the realistic dataset ObjectNN is the most significant. For the SSL 3D detection task, performing the re-sampling strategy during the pre-train stage to re-sample labeled data only can significantly improve the performance, while the performance can be further improved by also applying the re-sampling during the semi-supervised training stage. The results on both SSL 3D classification and detection benchmarks show the effectiveness of the proposed re-sampling strategy.

4.6. Ablation Study to Other Design Choice

To better understand our method, we conduct ablation studies to evaluate the impact of upper limit of thresholds and mapping functions. To comprehensively evaluate our method, we provide the ablation results on both SSL 3D classification task with 10 percent labeled data in ModelNet40 dataset and SSL 3D object detection task with 5 percent labeled data in ScanNet dataset.

Upper Limit Threshold: To investigate the impact of the upper limit threshold on our proposed method, we conduct ablation experiments on both classification and detection tasks. The results for SSL 3D classification and detection tasks are shown in Table. 7a and Table. 7c respectively. The 0.8 threshold achieves the best performance for both classification and detection tasks.

Learning Status Mapping Function: To better understand the effect of learning status mapping function, we verified the results of three different mapping functions including: (1) exponential: $M(x_c) = exp(-5 \times (1 - P_c)^2)$ (2) linear: $M(x_c) = P_c$, and (3) concave: $M(_c) = P_c/(2 - P_c)$. Where P_c is the class-level confidence for class c. The results for SSL 3D classification and detection tasks are shown in Table. 7b and Table. 7d respectively. We can see that the concave function leads to best performance for both classification and detection tasks. Besides, the performance of three mapping functions are similar for both tasks, which indicates the robustness of our method.

τ	Overall Acc	Mean Acc
0.75	87.0	81.5
0.8	**87.8**	82.5
0.85	87.5	**83.1**

(a) Upper limit threshold for classification.

Mapping function	Overall Acc	Mean Acc
Concave	**87.8**	**82.5**
Linear	87.1	81.4
Exp	87.4	82.1

(b) Mapping function for classification.

τ	mAP @0.25	mAP @0.5
0.75	42.7	23.6
0.8	**43.6**	**24.3**
0.85	43.2	24.1

(c) Upper limit threshold for detection.

Mapping function	mAP @0.25	mAP @0.5
Concave	43.6	**24.3**
Linear	**43.8**	23.5
Exp	43.0	24.0

(d) Mapping function for detection.

Table 7: Ablation study for upper limit threshold and mapping function in 3D SSL classification and detection task. (a) and (b) are classification task. (d) and (e) are results from detection task. All the results confirm that our method is very robust to those design choices.

5. Conclusion

In this work, we propose a novel class-level confidence based dynamic threshold method and re-sampling strategy. The proposed method not only improves the performance, but makes the prediction results balanced among classes. Our proposed method remarkably outperforms the state-of-the-art SSL classification and detection methods. These results demonstrate the effectiveness and generality of the proposed method in 3D SSL tasks. In the future, we will extend our method to 2D SSL tasks.

References

[1] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019.

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019.

[3] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020.

[4] Zhimin Chen, Longlong Jing, Yang Liang, YingLi Tian, and Bing Li. Multimodal semi-supervised learning for 3d objects. *arXiv preprint arXiv:2110.11601*, 2021.

[5] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

[6] Atin Ghosh and Alexandre H Thiery. On data-augmentation and consistency-based semi-supervised learning. *arXiv preprint arXiv:2101.06967*, 2021.

[7] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7(2):187–199, 2021.

[8] Ju He, Adam Kortylewski, Shaokang Yang, Shuai Liu, Cheng Yang, Changhu Wang, and Alan Yuille. Rethinking re-sampling in imbalanced semi-supervised learning. *arXiv preprint arXiv:2106.00209*, 2021.

[9] Minsung Hyun, Jisoo Jeong, and Nojun Kwak. Class-imbalanced semi-supervised learning. *arXiv preprint arXiv:2002.06815*, 2020.

[10] Jean Lahoud and Bernard Ghanem. 2d-driven 3d object detection in rgb-d images. In *Proceedings of the IEEE international conference on computer vision*, pages 4622–4630, 2017.

[11] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019.

[12] 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, 2013.

[13] Buyu Li, Wanli Ouyang, Lu Sheng, Xingyu Zeng, and Xiaogang Wang. Gs3d: An efficient 3d object detection framework for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1019–1028, 2019.

[14] Guohao Li, Matthias Muller, Ali Thabet, and Bernard Ghanem. Deepgcns: Can gcns go as deep as cnns? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9267–9276, 2019.

[15] Jerry Ma and Denis Yarats. On the adequacy of untuned warmup for adaptive optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8828–8836, 2021.

[16] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.

[17] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, and Ian J Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *arXiv preprint arXiv:1804.09170*, 2018.

[18] 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/CVF International Conference on Computer Vision*, pages 9277–9286, 2019.

[19] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[20] 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.

[21] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017.

[22] Zhile Ren and Erik B Sudderth. Three-dimensional object detection and layout prediction using clouds of oriented gradients. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1525–1533, 2016.

[23] S Shi, X Wang, H PointRCNN Li, et al. 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA*, pages 16–20, 2019.

[24] Martin Simony, Stefan Milzy, Karl Amendey, and Horst-Michael Gross. Complex-yolo: An euler-region-proposal for real-time 3d object detection on point clouds. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[25] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

[26] 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*, pages 567–576, 2015.

[27] Shuran Song and Jianxiong Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 808–816, 2016.

[28] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *arXiv preprint arXiv:1703.01780*, 2017.

[29] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6411–6420, 2019.

[30] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Duc Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *International Conference on Computer Vision (ICCV)*, 2019.

[31] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 14615–14624, 2021.

[32] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019.

[33] Chen Wei, Kihyuk Sohn, Clayton Mellina, Alan Yuille, and Fan Yang. Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10857–10866, 2021.

[34] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9621–9630, 2019.

[35] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.

[36] Xun Xu and Gim Hee Lee. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13706–13715, 2020.

[37] Yifan Xu, Tianqi Fan, Mingye Xu, Long Zeng, and Yu Qiao. Spidercnn: Deep learning on point sets with parameterized convolutional filters. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 87–102, 2018.

[38] Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised learning with dynamic thresholding. In *International Conference on Machine Learning*, pages 11525–11536. PMLR, 2021.

[39] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 7652–7660, 2018.

[40] Yuzhe Yang and Zhi Xu. Rethinking the value of labels for improving class-imbalanced learning. *arXiv preprint arXiv:2006.07529*, 2020.

[41] Tao Yu, Cuiling Lan, Wenjun Zeng, Mingxiao Feng, Zhizheng Zhang, and Zhibo Chen. Playvirtual: Augmenting cycle-consistent virtual trajectories for reinforcement learning. *Advances in Neural Information Processing Systems*, 34:5276–5289, 2021.

[42] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *Proceedings of the IEEE international conference on computer vision*, pages 1476–1485, 2019.

[43] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34, 2021.

[44] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5565–5573, 2019.

[45] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer. *arXiv preprint arXiv:2012.09164*, 2020.

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