Supplementary Materials for 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

This document contains the supplementary materials for "Class-Level Confidence Based 3D Semi-Supervised Learning".

1. Analysis on SSL Object Detection

In the main paper, we demonstrate the high correlation between class-level confidence of unlabeled data and test accuracy in 3D SSL classification task (line 100 to 104). Therefore, we hypothesize that class-level confidence of unlabeled data can be utilized to estimate learning status. To validate the generality of this correlation, we utilize 3DIoUMatch to conduct 3D SSL object detection experiment in SUN-RGBD dataset with 5 percent dataset as shown in the Fig 1. The results demonstrate that the class-level confidence also has high correlation with test accuracy in detection tasks, which supports the generality of our hypothesis.

2. Visualization of SSL Object Detection Results

To take a deeper look at the prediction results of our model, we compared the qualitative results of the supervised baseline, the 3DIoUMatch [7], and our results. The supervised baseline VoteNet [3] produces many false positive predictions due to the very limited labeled data during training. The results of 3DIoUMatch are much better than the supervised baseline but still have many false positive boxes. Compared to the baseline and 3DIoUMatch, the quality of our method is much higher and with fewer false-positive boxes demonstrating the effectiveness of our method.

2.1. Comparison with Class Imbalanced SSL Method

Most of current SSL imblanced method resamples 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. Unlike previous method,our re-sampling strategy directly increases the sampling probability of low learning status classes to balance the learning status. To further show the effectiveness of our method, we compare our method with recent state-of-the-art method BiS [2] that relies on class cardinality to sample. Table. 1 and Table. 2 indicate that our method still achieves better performance than BiS in both 3D detection and classification tasks when only the sampling part is utilized.

	ModelNet40 5%		ModelNet40 10%	
	Overall Acc	Mean Acc	Overall Acc	Mean Acc
Baseline	78.9	71.1	85.5	79.4
BiS [2] + Baseline	79.7	72.3	86.1	80.3
Confid-Sample + Baseline	81.0	72.8	86.7	81.1
Ours + Baseline	**82.1**	**74.3**	**87.8**	**82.5**

Table 1: Comparative studies with state-of-the-art class imbalanced SSL method for 3D object classification.

	SUN RGB-D 2%		SUN RGB-D 5%	
	mAP @0.25	mAP @0.5	mAP @0.25	mAP @0.5
Baseline	26.8 ± 1.1	10.6 ± 0.5	39.7 ± 0.9	20.6 ± 0.7
BiS [2] + Baseline	29.2 ± 0.7	11.5 ± 0.4	41.5 ± 1.1	22.4 ± 0.8
Confid-Sample + Baseline	31.9 ± 0.9	11.9 ± 0.7	42.4 ± 0.9	23.1 ± 0.6
Ours + Baseline	**32.7 ± 0.3**	**13.5 ± 0.4**	**43.1 ± 0.6**	**24.2 ± 0.5**

Table 2: Comparative studies with state-of-the-art class imbalanced SSL method for 3D object detection.

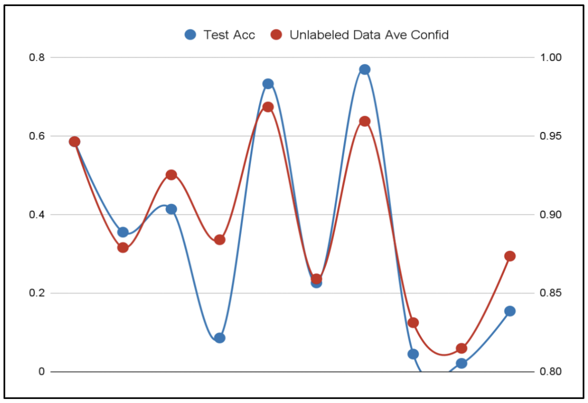


Figure 1: The results analysis of 3DIoUMatch trained in SUN-RGBD dataset with 5% labeled data. It indicates that the class-Level confidence has high correlation with test accuracy of each class in 3D SSL detection task.

3. Comparison with Our Method and Dash

Dash [9] proposed a dynamic threshold method based on cross entropy lose for all classes. However, the Latest SSL SOTA method FlexMatch [10] has demonstrated the benefit of class-level dynamic thresholding, which not only fully utilizes a large number of unlabeled data, but takes into account each class's learning status. Our method is inspired by FlexMatch and utilize class-level confidence to obtain dynamic threshold for each class. To further demonstrate the benefit of class-level threshold, we compare our method with Dash. Table. 3 shows that Dash only improves limited performance compared to baseline and our method achieves better performance than Dash in 3D tasks when only the dynamic threshold part is utilized.

	ModelNet40 10%		SUN RGB-D 5%	
	Overall Acc	Mean Acc	mAP @0.25	mAP @0.5
Baseline	85.5	79.4	39.7 ± 0.9	20.6 ± 0.7
Dash [38] + Baseline	85.9	80.1	40.5 ± 0.7	21.0 ± 0.6
Confid-Threshold + Baseline	86.9	81.7	42.0 ± 0.8	22.8 ± 0.5
Ours + Baseline	**87.8**	**82.5**	**43.1 ± 0.6**	**24.2 ± 0.5**

Table 3: Comparative studies with Dash.

4. Comparison with Our Method and FlexMatch

To utilize more unlabeled data at the early stage of the training, the Flexmatch [10] proposes a threshold warm-up strategy, which decreases the threshold according to the number of unused unlabeled data. However, due to the high learning difficulty of 3D data, a large number of unlabeled data remains unused during the training, which decreases the dynamic threshold of each class when FlexMatch is used. Furthermore, FlexMatch adjusts the threshold of each class according to the pseudo-labels numbers for each class. It works well for class-balanced dataset, but in commonly used 3D data [1, 6, 5, 8], the numbers of labeled data in each class is long-tail and thus the numbers of high confidence pseudo-labeled data is also tend to be long-tail. For example, in ModelNet40, 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. This makes the threshold value of airplane much larger than the bowl in FlexMatch. Hence, as demonstrated in Fig. 3a, utilizing FlexMatch generates low and significantly variant (long-tail) thresholds, which introduces much noise, especially for those minority classes, and thus achieves unsatisfied performances in 3D tasks. Unlike FlexMatch adjusting thresholds based on pseudo-label numbers, our method utilizes class-level confidence to adjust dynamic thresholds. As shown in the Fig. 3b our method produces appropriate and balanced dynamic thresholds, even when dataset is imbalanced. Hence, our method boosts the efficiency of utilizing unlabeled data without introducing much noise and have more generality than FlexMatch.

5. Per-class Accuracy for Classification

To understand the performance of our method on each class in the 3D SSL classification task, we report per-class accuracy on ScanObjectNN with 2 percent labeled data and ModelNet40 with 10 percent labeled data, respectively. The Table 6 indicates that in most classes, our method has better performance than FixMatch [4] and FlexMatch [10] . Moreover, In low learning status classes, the FlexMatch even degrades the performance of FixMatch due to the noise brought by low thresholds. Table 7 shows that in high learning status classes, performances of FixMatch, FlexMatch, and our method are similar. In low learning status classes, our method outperforms the FixMatch and FlexMatch by a large margin. This demonstrates our method's capability in improving low learning status classes and re-balance network's learning statuses.

6. Per-class Accuracy for Detection

To understand the performance of our method on each class in the 3D SSL detection task, we report per-class accuracy on SUN RGB-D with 2 percent labeled data and ScanNet with 5 percent labeled data, respectively. From Table 4 and 5 we can find that in most classes, our method has better performances than 3DIoUMatch [7] and backbone VoteNet [3].

7. Result Analysis and Discussion

To take a deeper look at how our model improves the performance, we conduct analysis about the results of our

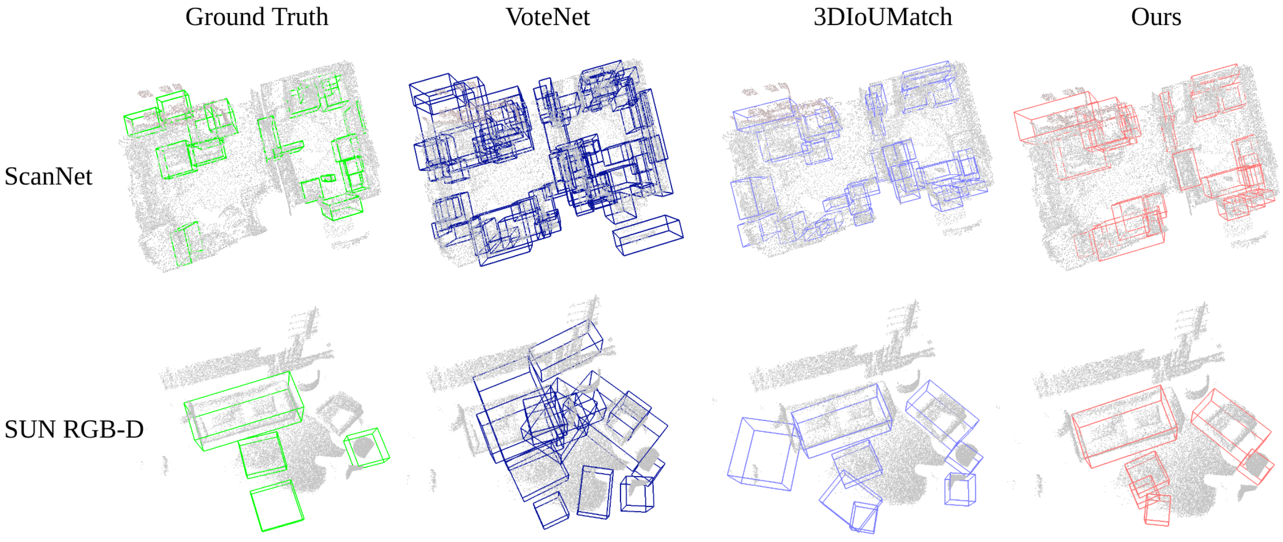
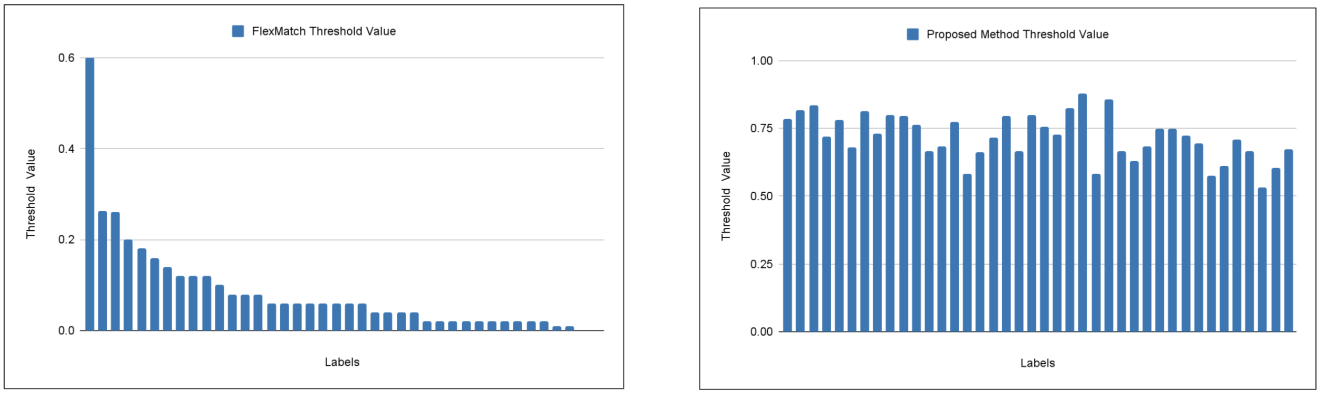


Figure 2: Detection results comparison between the supervised baseline VoteNet [3], 3DIoUMatch [7], and our method.



(a)

(b)

Figure 3: (a) Thresholds of our method for each class in the last epoch under ModelNet40 dataset with 10 percent labeled data. (b) Thresholds of FlexMatch for each class in the last epoch. The FlexMatch leads to long-tail thresholds, which introduces much noise in pseudo labels and thus degrades the performance. Thresholds of our method is appropriate and balanced, which boosts the efficiency of utilizing unlabeled data.

	bathtub	bed	bookshelf	chair	desk	dresser	nightstand	sofa	table	toilet
					mAP@0.25					
VoteNet [3]	52.8	28.8	24.8	48.1	21.5	9.0	0.3	2.8	0.8	24.4
3DIoUMatch [7]	60.1	**33.5**	**36.5**	55.9	36.2	6.7	0.3	15.9	3.6	32.5
Ours	**62.8**	30.3	32.6	**58.9**	**54.2**	**10.9**	**0.8**	**22.4**	**3.8**	**47.7**
					mAP@0.5					
VoteNet [3]	16.7	4.1	7.1	14.8	3.1	0.4	0	0.3	0.2	4.4
3DIoUMatch [7]	20.0	**9.7**	18.8	30.8	**5.1**	0.4	0	0.5	0.8	10.8
Ours	**25.3**	8.7	**19.3**	**34.8**	2.2	**1.1**	0	**9.4**	**1.2**	**24.2**

Table 4: Per-class performance comparison for the 3D object detection task with the state-of-the-art semi-supervised learning methods on the SUN RGB-D dataset with 2 percent labeled data.

model and compare with FixMatch, which uses a fixed threshold 0.90 and FlexMatch. We calculated the average class-level confidence of FixMatch, FlexMatch, and our method on ModelNet40 datasets.

Fig. 4a shows that the class-level confidence of FixMatch model is imbalanced and relatively lower. This is prob-

	cab	bed	chair	sofa	table	door	wind	bkshf	pic	cntr	desk	curt	fridg	showr	toil	sink	bath	ofurn
mAP@0.25																		
VoteNet [3]	12.5	65.9	70.2	69.2	37.6	14.0	8.5	15.6	0.5	10.0	48.2	16.8	20.2	20.6	72.1	28.1	44.4	8.6
3DIoUMatch [7]	**27.0**	**71.5**	78.4	72.3	48.0	**22.9**	**17.8**	14.1	1.6	40.0	51.6	25.0	29.8	43.6	81.7	33.5	75.1	16.0
Ours	24.6	69.6	**79.0**	**73.6**	**49.1**	18.5	16.5	**23.9**	**3.5**	**41.7**	**62.5**	**32.3**	**33.6**	**44.1**	**93.2**	**33.8**	**78.6**	**16.3**
mAP@0.5																		
VoteNet [3]	0.1	50.5	34.3	37.7	16.1	2.7	1.2	5.3	0	1.2	13.1	0.5	6.7	0	49.0	8.3	27.5	0.9
3DIoUMatch [7]	3.2	**57.0**	56.1	**53.2**	29.5	**8.6**	**4.9**	4.7	0	2.1	28.3	3.1	15.7	7.3	59.8	6.2	60.9	3.6
Ours	**4.1**	56.6	**56.9**	50.9	**30.7**	6.2	4.2	**8.3**	0	**3.2**	**31.6**	**11.1**	**23.2**	**7.6**	**60.0**	**9.0**	**62.9**	**4.6**

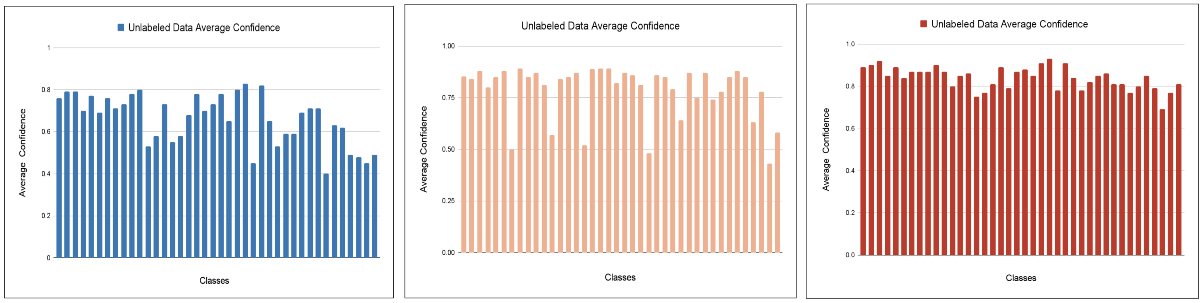
Table 5: Per-class performance comparison for the 3D object detection task with the state-of-the-art semi-supervised learning methods on the ScanNet dataset with 5 percent labeled data.

	bag	bin	box	cabinet	chair	desk	display	door
FixMatch [4]	2.4	41.7	**14.3**	49.5	83.1	**12.7**	52.5	94.8
FlexMatch [10]	1.1	36.7	0.5	49.5	85.9	3.3	45.1	**97.1**
Ours	**12.1**	**42.7**	12.8	**63.4**	**86.4**	6.0	**66.7**	95.7
	shelf	table	bed	pillow	sink	sofa	toilet	
FixMatch [4]	38.6	41.9	46.4	29.5	41.7	39.5	10.6	
FlexMatch [10]	35.7	**61.1**	**76.3**	3.9	51.2	50.5	2.6	
Ours	**41.5**	55.9	60.9	**34.8**	**53.7**	**87.6**	**12.9**	

Table 6: Per-class performance comparison for the 3D object classification task with the state-of-the-art semi-supervised learning methods on the ScabObjectNN dataset with 2 percent labeled data.

	airplane	bathtub	bed	bench	bookshelf	bottle	bowl	car	chair	cone
FixMatch [4]	100	**78**	**99**	65	98	95	85	94	98	**100**
FlexMatch [10]	100	62	99	60	97	**97**	**90**	**99**	**100**	100
Ours	**100**	76	98	**65**	**98**	96	86	98	98	95
	cup	curtain	desk	door	dresser	flower pot	glass box	guitar	keyboard	lamp
FixMatch [4]	30	55	**81**	80	**88**	20	91	99	95	65
FlexMatch [10]	40	50	73	85	76	25	91	100	100	65
Ours	**50**	**70**	74	**90**	83	**30**	**92**	**100**	**100**	**85**
	laptop	mantel	monitor	night stand	person	piano	plant	radio	range hood	sink
FixMatch [4]	100	92	95	44	65	85	**85**	50	85	60
FlexMatch [10]	100	93	99	47	75	93	80	60	72	**80**
Ours	**100**	**93**	**99**	**53**	**75**	**95**	78	**60**	**86**	75
	sofa	stairs	stool	table	tent	toilet	tv stand	vase	wardrobe	xbox
FixMatch [4]	**100**	65	**80**	90	95	99	61	81	35	75
FlexMatch [10]	98	90	55	**91**	95	99	**85**	**88**	25	65
Ours	99	**90**	70	90	**95**	**99**	82	85	**45**	**77**

Table 7: Per-Class performance comparison for the 3D object classification task with the state-of-the-art semi-supervised learning methods on the ModelNet40 dataset with 10 percent labeled data. In high learning status classes, performances of FixMatch, FlexMatch, and our method are similar. In low learning status classes, our method outperforms the FixMatch and FlexMatch by a large margin.



(a) FixMatch average confidence

(b) FlexMatch average confidence.

(c) Our method average confidence.

Figure 4: The comparison of unlabeled data class-level confidence between FixMatch and our method in ModelNet40 datasets trained with 10 percent labeled data. The result demonstrates that our method not only improves the learning status of each class but makes the learning status more balanced.

ably caused by the fixed threshold since it does not consider the learning difficulty and status of different classes. The Fig. 4b shows that although the FlexMatch improve confidence of some classes, the class-level confidence still remains imbalanced. This is because FlexMatch is not designed for data-imbalanced dataset and thus cannot rebalance the learning status. The class-level confidence of our proposed method is shown in the Fig. 4c. Benefiting from the dynamic threshold and re-sampling strategy, our proposed method not only improves classes' average confidence but also makes learning status balanced compared to the FixMatch and FlexMatch. This analysis confirms the advantage of using the dynamic threshold for each class.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.