Towards Robust Few-shot Point Cloud Semantic Segmentation

Yating Xu[1]
xu.yating@u.nus.edu
Na Zhao[2]
na_zhao@sutd.edu.sg
Gim Hee Lee[1]
gimhee.lee@nus.edu.sg

[1] Department of Computer Science
National University of Singapore
Singapore

[2] Singapore University of Technology and Design
Singapore

Abstract

Few-shot point cloud semantic segmentation aims to train a model to quickly adapt to new unseen classes with only a handful of support set samples. However, the noise-free assumption in the support set can be easily violated in many practical real-world settings. In this paper, we focus on improving the robustness of few-shot point cloud segmentation under the detrimental influence of noisy support sets during testing time. To this end, we first propose a Component-level Clean Noise Separation (CCNS) representation learning to learn discriminative feature representations that separates the clean samples of the target classes from the noisy samples. Leveraging the well-separated clean and noisy support samples from our CCNS, we further propose a Multi-scale Degree-based Noise Suppression (MDNS) scheme to remove the noisy shots from the support set. We conduct extensive experiments on various noise settings on two benchmark datasets. Our results show that the combination of CCNS and MDNS significantly improves the performance. Our code is available at https://github.com/Pixie8888/R3DFSSeg.

1 Introduction

Few-shot point cloud semantic segmentation (3DFSSeg) [23, 51] is a pragmatic direction as it is able to segment novel classes during testing stage with only few labeled samples. In contrast to the fully-supervised methods [27, 28, 39] which only work for close set, 3DFSSeg has better generalization ability. However, it assumes that the learning samples of the novel classes are correctly labeled during online testing time.

Unfortunately, the assumption of completely clean data could be violated in practice due to a variety of reasons. First, human labeling is error-prone. The irregular data structure, low-resolution, and subtle inter-class geometric difference make human annotators themselves hard to correctly recognize objects [37]. The crowdsourcing labeling further stresses the annotation quality [36]. As a consequence, ScanNet [6] still contains annotation mistakes [46] after manual refinement over an extended period of time. Second, the industry is actively seeking cheaper and more efficient annotation system to replace human labeling, *e.g.* semi-automatic labeling [16, 41] and fully automatic annotation [3, 5, 10]. It further challenges the curation of high-quality data.

© 2023. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

As shown in Fig. 1, we can refine the noisy annotations of the static base class dataset offline by either manual checking or data-driven algorithm [46] given enough time and budget. However, it is impossible to invest the same amount of human supervision to guarantee noise-free in every support set after model being deployed because the number of new classes in the real world is *infinite* [9, 26]. Neither can we use data-driven algorithm [46] to automatically clean the noise due to severe overfitting to the small number of training samples per new class (*cf.* Tab. 1).

To this end, we tackle with the noisy labels in the testing stage of 3DFSSeg, which is challenging but with high practical value. In 3DFSSeg, a few support point clouds are provided as learning samples for each new class during meta-testing. Each support sample (*i.e.* shot) is provided with a binary mask indicating the presence of the corresponding class. Based on the given support set, the model segments the new class in any unlabeled (*i.e.* query) point clouds. As pointed out by [10, 46] that the instance-level noise is most common in the annotation, objects of other classes are wrongly annotated as the target class and collected in the support set. We define shots with incorrectly labeled foreground object as noisy shots. Thus, the goal of robust few-shot point cloud semantic segmentation (R3DFSSeg) is to learn a robust few-shot segmentor that is less influenced by the noisy shots.

In this paper, we first propose a Component-level Clean Noise Separation (CCNS) representation learning to learn robust representation that is discriminative between features of clean and noisy points. Inspired by [51], we adopt the meta-learning paradigm for few-shot point cloud segmentation. During meta-training, we randomly inject noise into the support set by sampling point clouds containing foreground objects from other classes to mimic the noisy meta-testing environments. We introduce a class-wise supervised contrastive learning on the noisy support set to separate the clean samples of the target classes from the noisy samples. To obtain more fine-grained and diverse contrastive features, we further propose the use of farthest point sampling to decompose the masked points in the feature space into multiple components. Intuitively, our CCNS is designed to encourage features from different classes to be well-separated, such that the clean shots in the support set would form the largest cluster in the feature space when learning converges.

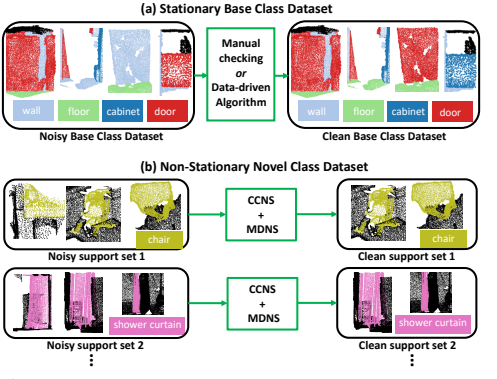

Figure 1: Comparison between noisy base and novel class dataset of 3DFSSeg. (a) Base class dataset is static with finite samples. (b) Novel class dataset is non-stationary as new classes are continuously collected in the online testing stage. An example where a sofa and a curtain are wrongly annotated in support set 1 and 2, respectively.

We further propose a Multi-scale Degree-based Noise Suppression (MDNS) scheme to remove the noisy shots from the support set during testing stage. Our MDNS separates clean from noisy samples by checking the degree of each sample in a fully connected pair-wise similarity graph. Clean samples tend to form well-defined clusters with higher degrees in the pair-wise similarity graph. In contrast, noisy samples are relatively scattered with lower degrees of connectivity in the feature space.

Our **main contributions** can be summarized as follows: **1)** To the best of our knowledge, we are the first to study the problem of robust few-shot point cloud semantic segmentation,

which is important in real-world applications since noisy labels are inevitable in practice. **2)** We propose a component-level clean noise separation method for representation learning to enhance the class-level discrimination in the embedding space. **3)** We propose a multi-scale degree-based noise suppression scheme that is able to effectively remove noisy samples from the small support set for each new class during testing. **4)** We conduct extensive experiments on two benchmark datasets (*i.e.* S3DIS and ScanNet) with various noise settings and show superior results over the baselines.

2 Related Work

Few-shot Learning. Few-shot learning aims to transfer knowledge learned from the abundant samples of the seen class to a set of unseen classes with only few labeled samples. One of the dominant approach is the metric-based [30, 35] methods, which meta-learns a transferable feature embedding that coincides with a fixed metric. The pioneer work ProtoNet [30] predicts query label by finding the nearest class prototype under the Euclidean distance. The key to the metric-based method is the discriminative feature embedding with compact class clusters [7, 22, 44, 45]. Ye *et al*. [45] apply the contrastive objective to align the training instances close to its own class center after the embedding adaptation. Although we also use contrastive learning in the episodic training, we adopt fine-grained contrastive objective (*i.e.* feature components) to better capture the diverse intra-class distribution of point cloud.

Few-shot Semantic Segmentation. Few-shot semantic segmentation segments semantic objects in an image [20, 38, 50] or a point cloud [13, 23, 51] with only few annotated samples. The 2D image semantic segmentation can be categorized into relation-based method [37, 48, 49, 50] and prototype-based method [20, 38, 43]. Zhao *et al*. [51] propose the first work on 3D few-shot point cloud semantic segmentation. They generate multi-prototypes via farthest point sampling to better capture the complex data distribution of the point cloud. The transductive inference is conducted between multi-prototypes and query points to infer the label for each query point. However, all these works assume that the annotation in the given support are accurate during testing time. In practice, this is a very strong assumption given that the pixel-level and point-level annotation are extremely tedious and error-prone. In view of this limitation, this paper studies the problem of robust few-shot point cloud semantic segmentation and proposes a effective model that can better adapt to real world applications.

Learning with Noisy Labels. Learning with noisy labels is gaining increasing attention as the deep neural networks are shown to be extremely vulnerable to the noisy labels [2, 8, 14]. There are three major approaches: label correction using the prediction of the model as the new label [14, 19, 29, 31], sample selection using small loss criterion to selectively update model [8, 40, 47] and learning robust representation [11, 15, 17, 25, 42]. PNAL [46] proposes robust point cloud semantic segmentation (R3DSeg) that studies label noise in the fully-supervised setting. It stores the prediction history of every point in the training dataset and corrects point labels in the cluster-wise manner epoch by epoch. All these noise-robust methods deal with static dataset in the offline training stage and require massive samples to train (data-driven). In contrast, R3DFSSeg addresses noise in the online testing stage, where new classes appear continuously with small support set. The data-driven algorithms would thus overfit in R3DFSSeg.

Existing methods dealing with noisy label of novel classes in the few-shot learning are only for 2D image classification (R2DFSL) [18, 21, 24]. RNNP [24] proposes a non-parametric test method by combining data augmentation with k-means to refine the class prototype. Liang *et al*. [18] learn a robust prototype generater by relying on the self-attention

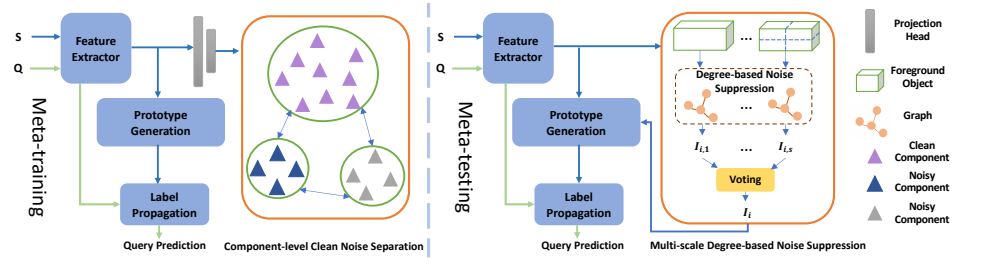

Figure 2: **The architecture of our framework**. 'S' represents the support point cloud and 'Q' represents the query point cloud. The left figure shows the pipeline during meta-training, where we conduct component-level clean noise separation representation learning for each episode class. Components of different classes are pushed away from each other. The right figure shows the pipeline during meta-testing, where we perform multi-scale degree-based noise suppression to remove the noisy shots.

module inside the Transformer [34] to weigh down the noisy shots. Compared to 2D classification, 3D point cloud segmentation is more challenging as it requires per-point classification and point cloud has much larger intra-class variance. Thus, the 2D methods, which only generate one robust prototype per class, fail on the R3DFSSeg.

3 Our Method

Problem Formulation. The few-shot point cloud segmentation consists of two datasets: $\mathcal{T}_{base}$ and $\mathcal{T}_{novel}$ sampled from disjoint classes $\mathcal{C}_{base}$ and $\mathcal{C}_{novel}$, respectively. The goal is to learn a model from $\mathcal{C}_{base}$ and generalize to the $\mathcal{C}_{novel}$. Following previous work [51], we adopt the episodic training on the $\mathcal{C}_{base}$ to emulate the few-shot setting during testing. In each N-way K-shot episode, N is the number of classes to be learned, and K is the number of labeled samples per class. The labeled samples are termed as the support set: $S = \left\{ \left(P_k^1, M_k^1\right)_{k=1}^K, \ldots, \left(P_k^N, M_k^N\right)_{k=1}^K \right\}$. Each point cloud $P_k^n \in \mathbb{R}^{m \times f_0}$ contains m points with input feature dimension of f_0. The $M_k^n \in \mathbb{R}^{m \times 1}$ is the corresponding binary mask indicating the presence of class n.

We are also given a set of T unlabeled point clouds, termed as the query set: $Q = \{(R_i, L_i)\}_{i=1}^T$. Each query point cloud $R_i \in \mathbb{R}^{m \times f_0}$ is associated with the ground truth label $L_i \in \mathbb{R}^{m \times 1}$ only available in the training stage. During testing, M_k^n can wrongly assign object of another class to class n due to the instance-level labeling error [46]. We denote the noisy mask $\tilde{M}_k^n$ and the corresponding point cloud $\tilde{P}_k^n$ as the noisy sample, and its correct class assignment as Y_k. Consequently, the support set S becomes the mixture of clean and noisy shots. The goal of robust few-shot point cloud semantic segmentation is to correctly predict the query label by learning from the noisy support set S.

Framework Overview. Fig. 2 illustrates our proposed framework. We choose AttMPTI [51] as our few-shot segmentor since it achieves state-of-the-art performance in the few-shot point cloud segmentation. In addition, AttMPTI is potentially robust to the noise when a good feature embedding is guaranteed (Sec. 3.1). In view of this, we propose the Component-level Clean Noise Separation (CCNS) representation learning during meta-training to enhance the discrimination and generalization of the feature embedding for AttMPTI (Sec. 3.2). We further propose the multi-scale degree-based noise suppression (MDNS) to remove the noisy shots during meta-testing based on their similarity graph (Sec. 3.3).

3.1 Why Choose AttMPTI?

AttMPTI [51] is the state-of-the-art few-shot point cloud segmentation method. It consists of a feature extractor to embed the support and query point cloud into the same metric space, a multi-prototype generation module to generate prototypes from support set, and a label propagation module to infer query label. Compared to ProtoNet [30], AttMPTI has several unique components that gives it the potential to be robust, in addition to showing more superior performance. **First**, AttMPTI generates multi-prototypes via FPS [28], while ProtoNet uses mean aggregation of all the relevant class feature. The sampled seed points via FPS are able to represent the diversity of the feature space, and the local prototype is generated by clustering each point to the nearest seed point based on the Euclidean distance in the feature space. In this way, the multi-prototypes can inherently separate the clean and noisy points in the prototype-level. As shown in Fig. 3, the clean ratio of local prototypes is either 1 (100% clean) or 0 (100% noise), but it seldom produces a half-clean prototype. In comparison, the global prototype used in the ProtoNet leads to a clean-noise compound. **Second**, AttMPTI infers query labels via label propagation [52] in a transductive fashion, while ProtoNet infers each query point independently with the set of class prototypes. The label propagation is based on the manifold smoothness, *i.e.* nearby samples in the feature space share the same label, and it has the ability to correct the noisy label [11, 42]. In contrast, ProtoNet independently and identically predicts the label for each query point based on the global prototypes that are potentially noisy. The lack of reasoning the relationships among the support and query prevents the model from being able to correct the support noise. Although the design of AttMPTI shows a better potential than ProtoNet in resisting the noise existing in the support set, the performance of both multi-prototype generation and label propagation are subjected to the discriminativity of the feature embeddings. To enhance the representation learning, we propose to perform component-level clean-noise separation.

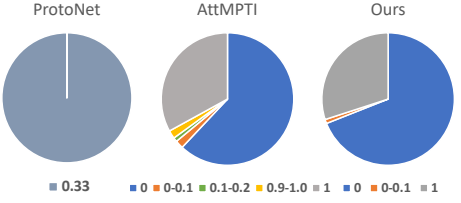

Figure 3: Comparison of prototype cleanness from different methods on a 5-shot with 40% out-episode noise setting. '1' means the prototype only containing clean-labeled points, and '0' means the prototype only containing points that are incorrectly labeled as the target class. Values in between 0-1 represent the portion of clean-labeled points in the prototype.

3.2 Component-level Clean Noise Separation

Our component-level clean noise separation (CCNS) representation learning aims to enhance the class-wise discrimination in the feature space. We randomly replace some of the K support shots with shots sampled from other classes during episodic training and induce the model to differentiate clean and noisy shots in the feature space. With these synthesized support sets with noisy labels, we perform a clean-noise separation representation learning for each way (*i.e.* class) by optimizing the model with the class-wise contrastive learning among the K support shots as follow:

$$\mathcal{L}_{\text{CNS}} = \frac{1}{K}\sum_{k=1}^{K}\left(\frac{-1}{|A(z_k)|}\sum_{z_g \in A(z_k)} \log \frac{\exp\left(z_k \cdot z_g/\tau\right)}{\sum_{h \setminus k} \exp\left(z_k \cdot z_h/\tau\right)}\right), \tag{1}$$

where $z_k \in \mathbb{R}^d$ is the L2 normalized average foreground feature of the support point cloud P_k in the projection space. $A(z_k) = \{z_g \mid Y_g = Y_k\}$ is the set of positive samples z_g with its semantic label Y_g the same as the semantic label Y_k of z_k. $|A(z_k)|$ is the cardinality and τ is the temperature. By training with $\mathcal{L}_{\text{CNS}}$, the shots with same foreground class are encouraged to stay together while staying away from samples of other classes.

Unfortunately, a simple mean aggregation of the foreground area tends to be sub-optimal in representing the class distribution since the distribution of point features of each class is very large as shown in Fig. 4. To this end, we conduct class-wise contrastive learning in a more fine-grained way by dividing the features in each foreground area into local components. The feature components aggregate local patterns that exhibit similar fine-grained semantics, and have better coverage of the feature space compared to the naive mean aggregation. Specifically, we first perform FPS in the feature space and then locally aggregate the point features into a set of feature components $\{z_k^1, \cdots, z_k^R\}$, to replace the original holistic z_k. Consequently, the component-level clean noise separation $\mathcal{L}_{\text{CCNS}}$ is formulated as:

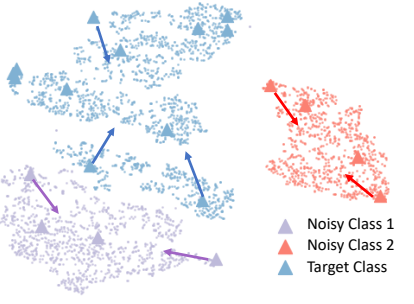

Figure 4: t-SNE [33] visualization of the CCNS on a 5-shot support set with 2 noisy shots. Each dot represents a point in the feature space and each triangle represents a feature component. Different colors represent different classes with blue indicating the target class. The arrow shows the direction to pull the feature components.

$$\mathcal{L}_{\text{CCNS}} = \frac{1}{KR}\sum_{k=1}^{K}\sum_{i=1}^{R}\left(\frac{-1}{|A(z_k^i)|}\sum_{z_g^j \in A(z_k^i)} \log \frac{\exp\left(z_k^i \cdot z_g^j/\tau\right)}{\sum_{h,b\setminus(k,i)} \exp\left(z_k^i \cdot z_h^b/\tau\right)}\right), \tag{2}$$

where the $A(z_k^i) = \left\{z_g^j \mid Y_g = Y_k\right\}$ is the set of positive samples with the same semantic label Y_g as Y_k, and the $|A(z_k^i)|$ is the cardinality. As shown in Fig. 4, each component represents a different aspect of its corresponding shot in the feature space. Essentially, it forms a multi-view self-supervised contrastive learning for each shot, where the 'view' is a local component in the feature space. Correspondingly, the components at the boarder of the class distribution automatically serve as the hard negative samples to other classes and hard positive samples to its own class, which are the key to a successful contrastive learning [4, 12].

The final optimization objective during the training stage is given by:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda \mathcal{L}_{\text{CCNS}}, \tag{3}$$

where λ is a hyper-parameter to weigh the contribution of $\mathcal{L}_{\text{CCNS}}$. $\mathcal{L}_{CE}$ is the original cross-entropy loss in AttMPTI.

3.3 Multi-scale Degree-based Noise Suppression

Although the clean and noisy points can separate under the well-learned embedding space, the prototype generation and label propagation module are still exposed to the mislabeled shots during testing time. To reduce their negative influence during testing, we design a degree-based noise suppression scheme to automatically remove the suspicious noisy shots. Specifically, we build a fully connected graph G on the K support shots for each way. We

average the foreground feature $x_i \in \mathbb{R}^d$ of the i-th shot as the feature of node i. The weight W_{ij} of the edge encodes the affinity between the two end nodes i and j as follow:

$$W_{ij} := \begin{cases} \left[x_i^\top x_j\right]_+^\gamma, & \text{if } i \neq j \\ 0, & \text{otherwise} \end{cases}. \tag{4}$$

We then compute the degree $d_i = \sum_j W_{ij}$ for each node i. Essentially, the degree reflects the nodes connection in the graph. The noisy shots tend to have lower degree since the clean shots usually form a cluster with the largest size and the noisy shots are scattered in the feature space. Consequently, we identify them based on the clean indicator:

$$I_i := \begin{cases} 1 & \text{if } d_i > thr \\ 0, & \text{otherwise} \end{cases}, \tag{5}$$

where we set the thr as the mean of the $\{d_i\}_{i=1}^K$. The shots with $I = 0$ are treated as noise and removed.

Some point clouds may have complex data distribution that cannot be sufficiently represented by a global representation. To mitigate this problem, we extend the single-level degree-based noise suppression scheme to multi-level, thus yielding the Multi-scale Degree-based Noise Suppression (MDNS). Our MDNS can be more robust to some complex samples and consequently improve the accuracy of clean sample identification. Specifically, we add an additional level to perform noise suppression. We evenly split the foreground object along the x/y/z coordinates, and denote the number of cuts along the x/y/z coordinates as $n_x/n_y/n_z$. The foreground feature in each sub-shot is locally aggregated and the feature set for each shot is enlarged to $\left\{x_{i,s}^1, \cdots, x_{i,s}^e\right\}$, where $e = n_x \times n_y \times n_z$. The single representation x_i is the case of $\left\{n_x = 1, n_y = 1, n_z = 1\right\}$ and is considered as the coarsest scale with $s = 1$. We then send them into the noise suppression module to get the clean indicator $\left\{I_{i,s}^1, \cdots, I_{i,s}^e\right\}$, where the majority voting is performed get the shot-level indicator $I_{i,s}$. Lastly, we assemble the final prediction I_i as the majority voting of the prediction at each scale $\{I_{i,1}, \ldots, I_{i,s}\}$.

4 Experiments

4.1 Datasets and Noise Settings

Datasets. We conduct experiments on **S3DIS** [1] and **ScanNet** [6]. S3DIS contains point clouds of 272 rooms collected from six indoor areas with annotation of 12 semantic classes. ScanNet contains point clouds of 1,513 scans from 707 unique indoor scenes with annotation of 20 semantic classes. Following [51], we split each room into non-overlapping blocks with size of 1m $\times$ 1m on the xy plane. Consequently, S3DIS and ScanNet contains 7,547 and 36,350 blocks, respectively. We sample $m = 2,048$ points as the input point cloud from a block. The input feature f_0 corresponds to XYZ, RGB and normalized XYZ values. During training, we randomly sample one episode by first sampling N classes from $\mathcal{C}_{base}$ and then sampling NK point clouds as the support set and T point clouds as the query set. The support mask M and the query label L are modified from its original annotation to only indicate the presence of the target classes with irrelevant classes as the background. The testing episodes are formed in a similar way, except for that we exhaustively sample 100 episodes for each combination of N classes from the $\mathcal{C}_{novel}$. We use the data split 0 of [51] as the test classes on both datasets. We adopt the mean Intersection over Union (mIoU) as the evaluation metric.

model	0%		In-episode Noise				Out-episode Noise			
			20%		40%		40%		60%	
	2-way	3-way	2-way	3-way	2-way	3-way	2-way	3-way	2-way	3-way
PNAL [46]	13.67	8.12	8.94	5.45	5.95	3.13	8.08	4.28	4.77	2.87
Tra-NFS [18]	44.98	31.67	43.44	30.68	37.27	27.39	41.72	28.43	35.67	23.20
ProtoNet [30]	57.02	46.78	54.21	43.57	42.57	36.71	50.01	39.31	44.96	36.08
AttMPTI [51]	65.90	51.71	60.01	47.96	38.81	37.56	58.60	44.76	51.18	40.32
Ours	**68.21**	**54.79**	**66.02**	**52.91**	**58.01**	**48.72**	**66.09**	**50.71**	**58.84**	**46.19**

Table 1: Results on the S3DIS using mIoU metric on 2-way 5-shot and 3-way 5-shot.

model	0%		In-episode Noise				Out-episode Noise			
			20%		40%		40%		60%	
	2-way	3-way	2-way	3-way	2-way	3-way	2-way	3-way	2-way	3-way
Tra-NFS [18]	41.89	31.56	39.72	29.20	34.25	25.07	38.42	27.29	34.68	23.78
ProtoNet [30]	47.55	38.97	44.19	36.46	34.57	30.23	42.47	33.88	36.64	28.55
AttMPTI [51]	**54.16**	**44.52**	46.63	38.83	31.57	27.62	43.31	34.33	36.45	26.79
Ours	53.50	43.84	**49.78**	**41.01**	**38.70**	**34.03**	**47.90**	**38.93**	**38.42**	**28.81**

Table 2: Results on the ScanNet using mIoU metric on 2-way 5-shot and 3-way 5-shot.

Noise Settings. We explore two types of label noise: 1) **In-episode noise** samples noisy shots from other N-1 classes of the current episode. It studies how the mix of the N foreground classes affects the prediction of query point. We test the models on in-episode noise ratio of 20% and 40%. 2) **Out-episode noise** samples noisy shots from outside of the N classes in the $\mathcal{C}_{novel}$. It studies how the outliers affect the prediction of the query point. We test the models on out-episode noise ratio of 40% and 60%.

The noise rate is defined as the percentage of the K support shots. Following existing literature of learning with noisy labels [8, 14, 18], we define the noise ratio with the restriction that the percentage of clean labeled samples is larger than any noisy class. We thus can only consider up to 40% noise for the in-episode noise and up to 60% noise for the out-episode noise in both 2-way 5-shot and 3-way 5-shot point cloud segmentation.

4.2 Implementation Details

We adopt the AttMPTI [51] as the few-shot segmentor and follow the same training procedure as AttMPTI. We first pre-train the feature extractor 100 epochs on the $\mathcal{C}_{base}$ with learning rate of 0.001 and Adam optimizer. In the episodic training, the feature extractor is fine-tuned with learning rate of 0.0001 and other learnable modules are optimized with learning rate of 0.001. The projection head consists of one fully-connected layer with the output dimension d as 128. Both λ and τ are set to 0.1, and R is set 4 in $\mathcal{L}_{\text{CCNS}}$. We randomly generate noisy shots in each episode during training by sampling shots from $\mathcal{C}_{base}$. The noise ratio is randomly chosen from $\{0, 0.2, 0.4\}$. In MDNS, the γ is set to 3 for scale s=1 and to 1 for any other scale. The MDNS is conducted in two scales: $\{n_x = 1, n_y = 1, n_z = 1\}$ and $\{n_x = 2, n_y = 2, n_z = 1\}$. All the experiments are done using one GTX 3090 GPU.

4.3 Main Results

We compare with 3DFSSeg methods AttMPTI [51] and ProtoNet [30], R2DFSL method Tra-NFS [18] and R3DSeg method PNAL [46]. All methods use the same feature extractor as AttMPTI for fair comparison.

Tab. 1 and Tab. 2 presents the experiment results on the noisy 2-way 5-shot and 3-way 5-shot point cloud segmentation on S3DIS and ScanNet, respectively. AttMPTI [51] usually has better performance than ProtoNet [30] in terms of various noise setting, but both methods suffer considerably with increasing noise ratio. It suggests that few-shot segmentor is indeed vulnerable to the support noise. On the other hand, our method is able to largely improve the robustness of the AttMPTI over all noise settings on both datasets. Fig. 5 presents the qualitative results of a 2-way 5-shot point cloud segmentation with 40% out-episode noise on the S3DIS. It shows our method can correctly segments the target classes in the query

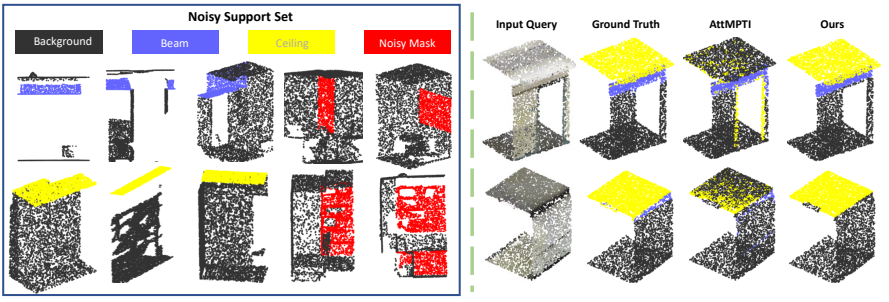

Figure 5: Qualitative comparison of a 2-way 5-shot point cloud segmentation with 40% out-episode noise on S3DIS.

point while AttMPTI fails. We notice that our model is slightly worse than AttMPTI in the 0% setting in Tab. 2. We postulate that our method can predict correct labels, but the noisy ground truths of ScanNet [46] cannot reflect the true performance of our method. This postulation is evidenced by the great superiority of our method over baseline methods on S3DIS, which is a dataset with clean ground truths. It suggests that our method can adapt to the unknown test environment (both clean and noise test), which is important for model deployment in real world.

2D robust few-shot learner Tra-NFS [18] performs poorly on R3DFSSeg due to severe modality gap, *i.e.* point cloud has larger intra-class viriance than 2D images, making Tra-NFS hard to detect clean shots. 3D robust point cloud segmentor PNAL [46] also fails in the few-shot setting due to small support set in each episode.

We further notice that the in-episode noise has larger negative influence than the out-episode noise, *e.g.* 40% in-episode noise vs 40% out-episode noise. We believe the reason is that the features in each foreground class usually form a compact cluster. The in-episode noise causes the labels in this compact cluster to be different, which severely confuses the model of which class this cluster belongs to. In contrast, the out-episode noise are usually separated from the foregrond classes in the feature space, and is less likely to influence them.

4.4 Ablation Study

Effectiveness of CCNS and MDNS. We analyze the effectiveness of our proposed component-level clean noise suppression (CCNS) and multi-scale degree-based noise suppression (MDNS) on S3DIS in Tab. 3. It is worth noting that the robustness of AttMPTI is improved by simplely adding our feature representation learning, *i.e.* CCNS. It verifies our claim that AttMPTI has the potential to be noise robust (by FPS based multi-prototype generation and label propagation), yet is subject to how discriminative the feature embedding is. MDNS improves performance on most settings and its performance gains are also subjected to the discriminativity of the feature space. By further equipping with CCNS, our final model achieves consistent and significant improvements under all settings. The ablation study on hyperparameter choosing is provided in the **supplementary material**.

model	0%	In-episode Noise		Out-episode Noise	
		20%	40%	40%	60%
AttMPTI	65.90	60.01	38.81	58.60	51.11
AttMPTI+CCNS	**68.50**	63.10	41.75	63.77	56.79
AttMPTI+MDNS	64.80	63.03	52.78	61.73	52.98
Ours	68.21	**66.02**	**58.01**	**66.09**	**58.84**

Table 3: Effectiveness of CCNS and MDNS on the S3DIS on 2-way 5-shot. 'Ours' consists of both CCNS and MDNS.

High way setting. Tab. 4 shows results of 5-way 5-shot setting on ScanNet. Our model again can significantly outperform AttMPTI on all noise settings.

model	0%	In-episode Noise		Out-episode Noise	
		20%	40%	40%	60%
AttMPTI	32.75	27.96	20.72	23.89	17.54
Ours	32.74	**30.79**	**26.73**	**28.13**	**21.22**

Table 4: 5-way 5-shot setting on ScanNet.

5 Conclusion

In this paper, we address the new task of robust few-shot point cloud segmentation, which is a more general setting that considers label noise in the support set. We design the Component-level Clean Noise Separation (CCNS) representation learning to learn a discriminative feature embedding. Our CCNS encourages the features from different classes to stay away from each other, and concurrently induces the clean shots to form the largest cluster in the feature space. Leveraging the clean samples identified from our CCNS, we further propose the Multi-scale Degree-based Noise Suppression (MDNS) to remove the noisy shots before the prototype generation based on their affinity with other samples in the support set. Experiment results that outperform the baselines show the feasibility of our proposed method.

Acknowledgement. This research is supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG2-RP-2021-024), and the Tier 2 grant MOE-T2EP20120-0011 from the Singapore Ministry of Education. This research is also supported by the SUTD-ZJU Thematic Research Grant RS-MEZJU-00031. The work is fully done at the National University of Singapore.

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016.

[2] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pages 233–242. PMLR, 2017.

[3] Daan Bloembergen and Chris Eijgenstein. Automatic labelling of urban point clouds using data fusion. *arXiv preprint arXiv:2108.13757*, 2021.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[5] Zhengyong Chen, Qinghai Liao, Zhe Wang, Yang Liu, and Ming Liu. Image detector based automatic 3d data labeling and training for vehicle detection on point cloud. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1408–1413. IEEE, 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, pages 5828–5839, 2017.

[7] Yizhao Gao, Nanyi Fei, Guangzhen Liu, Zhiwu Lu, and Tao Xiang. Contrastive prototype learning with augmented embeddings for few-shot learning. In *Uncertainty in Artificial Intelligence*, pages 140–150. PMLR, 2021.

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

[9] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Proceedings of the IEEE international conference on computer vision*, pages 3018–3027, 2017.

[10] Galadrielle Humblot-Renaux, Simon Buus Jensen, and Andreas Møgelmose. From cad models to soft point cloud labels: An automatic annotation pipeline for cheaply supervised 3d semantic segmentation. *arXiv preprint arXiv:2302.03114*, 2023.

[11] Ahmet Iscen, Jack Valmadre, Anurag Arnab, and Cordelia Schmid. Learning with neighbor consistency for noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4672–4681, 2022.

[12] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

[13] Lvlong Lai, Jian Chen, Chi Zhang, Zehong Zhang, Guosheng Lin, and Qingyao Wu. Tackling background ambiguities in multi-class few-shot point cloud semantic segmentation. *Knowledge-Based Systems*, 253:109508, 2022.

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

[15] Junnan Li, Caiming Xiong, and Steven CH Hoi. Learning from noisy data with robust representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9485–9494, 2021.

[16] MingHui Li and Yanning Zhang. 3d point cloud labeling tool for driving automatically. In *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1666–1672. IEEE, 2020.

[17] Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 316–325, 2022.

[18] Kevin J Liang, Samrudhdhi B Rangrej, Vladan Petrovic, and Tal Hassner. Few-shot learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9089–9098, 2022.

[19] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

[20] Yongfei Liu, Xiangyi Zhang, Songyang Zhang, and Xuming He. Part-aware prototype network for few-shot semantic segmentation. In *European Conference on Computer Vision*, pages 142–158. Springer, 2020.

[21] Jiang Lu, Sheng Jin, Jian Liang, and Changshui Zhang. Robust few-shot learning for user-provided data. *IEEE transactions on neural networks and learning systems*, 32 (4):1433–1447, 2020.

[22] Jiawei Ma, Hanchen Xie, Guangxing Han, Shih-Fu Chang, Aram Galstyan, and Wael Abd-Almageed. Partner-assisted learning for few-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10573–10582, 2021.

[23] Yongqiang Mao, Zonghao Guo, LU Xiaonan, Zhiqiang Yuan, and Haowen Guo. Bidirectional feature globalization for few-shot semantic segmentation of 3d point cloud scenes. In *2022 International Conference on 3D Vision (3DV)*, pages 505–514. IEEE, 2022.

[24] Pratik Mazumder, Pravendra Singh, and Vinay P Namboodiri. Rnnp: A robust few-shot learning approach. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2664–2673, 2021.

[25] Diego Ortego, Eric Arazo, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Multi-objective interpolation training for robustness to label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6606–6615, 2021.

[26] Senthil Purushwalkam, Pedro Morgado, and Abhinav Gupta. The challenges of continuous self-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, pages 702–721. Springer, 2022.

[27] 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.

[28] 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.

[29] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[30] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[31] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In *International Conference on Machine Learning*, pages 5907–5915. PMLR, 2019.

[32] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[33] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[35] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

[36] Volker Walter, Michael Kölle, and Yifey Yin. Evaluation and optimisation of crowd-based collection of trees from 3d point clouds. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:49–56, 2020.

[37] Bernie Wang, Virginia Wu, Bichen Wu, and Kurt Keutzer. Latte: accelerating lidar point cloud annotation via sensor fusion, one-click annotation, and tracking. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 265–272. IEEE, 2019.

[38] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9197–9206, 2019.

[39] 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)*, 38(5):1–12, 2019.

[40] Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13726–13735, 2020.

[41] Florian Wirth, Jannik Quehl, Jeffrey Ota, and Christoph Stiller. Pointatme: efficient 3d point cloud labeling in virtual reality. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1693–1698. IEEE, 2019.

[42] Zhi-Fan Wu, Tong Wei, Jianwen Jiang, Chaojie Mao, Mingqian Tang, and Yu-Feng Li. Ngc: a unified framework for learning with open-world noisy data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 62–71, 2021.

[43] Boyu Yang, Chang Liu, Bohao Li, Jianbin Jiao, and Qixiang Ye. Prototype mixture models for few-shot semantic segmentation. In *European Conference on Computer Vision*, pages 763–778. Springer, 2020.

[44] Zhanyuan Yang, Jinghua Wang, and Yingying Zhu. Few-shot classification with contrastive learning. In *European Conference on Computer Vision*, pages 293–309. Springer, 2022.

[45] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8808–8817, 2020.

[46] Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao. Learning with noisy labels for robust point cloud segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6443–6452, 2021.

[47] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pages 7164–7173. PMLR, 2019.

[48] Bingfeng Zhang, Jimin Xiao, and Terry Qin. Self-guided and cross-guided learning for few-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8312–8321, 2021.

[49] Chi Zhang, Guosheng Lin, Fayao Liu, Jiushuang Guo, Qingyao Wu, and Rui Yao. Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9587–9595, 2019.

[50] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5217–5226, 2019.

[51] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Few-shot 3d point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8873–8882, 2021.

[52] Dengyong Zhou, Olivier Bousquet, Thomas Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. *Advances in neural information processing systems*, 16, 2003.