White Aggregation and Restoration for Few-shot 3D Point Cloud Semantic Segmentation

Jiyun Im, SuBeen Lee, Miso Lee, and Jae-Pil Heo*

Sungkyunkwan University
{bbangsil0110, leesb7426, dlalth557, jaepilheo}@skku.edu

Abstract. Few-shot 3D Point Cloud Semantic Segmentation (FS-PCS) aims to predict per-point labels for an unlabeled point cloud, given only a few labeled examples. To extract representations from the limited labeled set, existing methods have constructed prototypes with Farthest Point Sampling (FPS). However, we found that this convention results in performance instability due to its sensitivity to FPS-induced variations, while the prototype generation process remains underexplored in the field. This motivates us to investigate deterministic prototype generation method based on attention mechanism. Despite its potential, we found that vanilla attention module suffers from the distributional gap between prototypical tokens and support features. To overcome this, we provide a simple approach, White Aggregation and Restoration Module (WARM), which resolves the misalignment by wrapping cross-attention with whitening and coloring transformations. Specifically, whitening aligns the features to tokens before the attention process, and coloring subsequently restores the original distribution to the attended tokens. This design enables robust attention, thereby generating prototypes that capture the semantic relationships in support features. WARM achieves state-of-the-art performance with a significant margin on the S3DIS dataset, and competitive performance on the ScanNet dataset. Further experiments demonstrate its effectiveness in deterministic prototype generation. Code is publicly available at: https://github.com/JiyunIm00/WARM.git

Keywords: Few-shot Learning · 3D Semantic Segmentation · Attention

1 Introduction

Understanding the semantics of 3D point clouds has become crucial as its applications have advanced [6, 32]. Although recent methods have achieved remarkable performance, they rely on large amounts of labeled data, which requires expensive labor [3, 9]. To alleviate this data reliance, Few-shot 3D Point Cloud Semantic Segmentation (FS-PCS) was introduced [37]. It aims to segment unseen novel classes in unlabeled point clouds, referred to as the query set, using a small number of labeled examples, referred to as the support set.

* Corresponding author

Table 1: mIoU (%) distribution resulting from random seeds. Performance is evaluated under the 1-way 1-shot setting on the first split of S3DIS [3]. 'FPS + *min-dist.*' denotes a simple baseline that assigns labels based on the minimum distance to FPS-constructed class prototypes. The distribution of mIoUs show the sensitivity of FS-PCS models to FPS results, that even the complex segmentation structure [2] falls short to generalize. On the other hand, attention-based WARM shows much stability.

Method	Max	Min	Mean	Std.
COSeg [2]	52.86	37.99	45.67	2.41
FPS + *min-dist.*	52.14	46.86	49.48	0.86
WARM	60.16	60.16	60.16	-

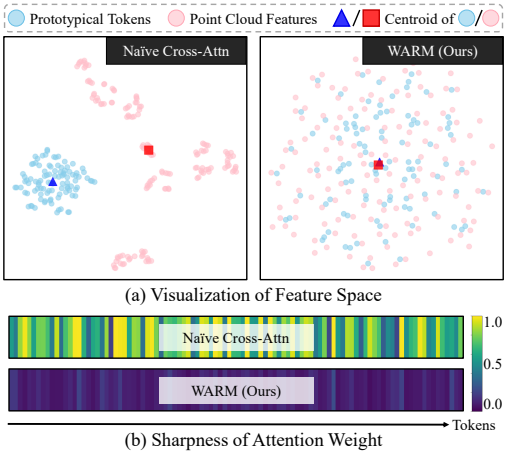

Fig. 1: Comparison of naïve cross-attention and WARM in prototype construction. (a) t-SNE visualization of the alignment between prototypical tokens (queries) and point cloud features (keys). (b) Attention sharpness measured via attention entropy [35], averaged for each token.

To fully utilize the information from the support set, previous studies have constructed class prototypes [1,2,13,25,37]. It is widely adopted in other few-shot downstream tasks, where numerous studies focus on improving representation quality due to its impact on performance [12,19,23,36]. In FS-PCS, most existing methods have relied on algorithmic approach such as Farthest Point Sampling (FPS) to construct multiple prototypes of the target class, following [37].

Surprisingly, these conventional prototypes cause performance instability as shown in Tab. 1. Due to its results being dependent on the initially selected point [34], FPS produces inconsistent results that create fluctuations in performance. Particularly, the subsequent sophisticated module [2] aggravates this sensitivity when compared to a simple distance-based method, highlighting the importance of prototype construction in FS-PCS. This motivates us to devise an advanced, consistent prototype generation module.

One may consider the attention mechanism [29] as a promising candidate, given its deterministic nature and demonstrated adaptability to prototype generation [7,8,19,36]. Nevertheless, we find that attention alone remains insufficient in the case of FS-PCS. Ideally, the cross-attention module should be optimized such that the prototypical tokens adaptively aggregate information from support features to form representative prototypes. However, this process fails due to a large distributional gap between the prototypical tokens and the support point features, as shown in Fig. 1 (a).

This misalignment prevents the prototypical tokens from establishing meaningful interactions across semantically correlated regions, causing attention to

collapse onto a limited subset of features and consequently degrading representation quality as discussed in [11, 15, 18, 35]. Consistently, in Fig. 1 (b), most tokens allocate more than half of their total attention weight to an individual support point, indicating a lack of compositional understanding of the class.

To enable effective attention-based prototype generation in FS-PCS, we introduce a simple approach that integrates cross-attention with established alignment techniques. Our proposed White Aggregation and Restoration Module (WARM) constructs semantically meaningful prototypes via robust attention induced by aligned prototypical tokens and support features. Specifically, in order to align with the prototypical tokens, the support features are transformed to a whitened space [5] where it is standardized and decorrelated. By operating in a shared feature space, the tokens establish interactions with the features that promote semantically coherent representations instead of ones locally confined in the projection space. Moreover, decorrelating features promotes smoother and more stable optimization [10, 14]. Subsequently, coloring—the inverse operation of whitening—restores the separated characteristics to the resulting prototypes, preserving essential feature properties for faithful representation. Ultimately, WARM contributes to effective attention-based prototype generation in FS-PCS by improving compatibility between prototypical tokens and support features, while also fostering stable optimization.

As a result, our attention-based WARM alone achieves competitive performance on FS-PCS benchmarks [3,9], with state-of-the-art performance on S3DIS dataset. We validate our approach through extensive experiments, demonstrating its effectiveness in addressing the challenges of adapting attention-based prototype generation to FS-PCS. The contribution of our work lies in rethinking a simple architecture in light of task-specific challenge and introducing a complementary perspective for future research in FS-PCS. In summary, our contributions are as follows:

- We identify the challenges inherent in adapting attention mechanisms for prototype generation in FS-PCS, providing detailed analysis of the distributional factors that limit their representational capacity.
- We devise a simple solution, WARM, that utilizes concrete alignment method to address the distributional disparity during cross-attention, and validate its effectiveness through comprehensive ablation analyses.
- Our method achieves competitive performance in FS-PCS benchmarks, without the aid of complex decoders.

2 Related Works

2.1 Few-Shot 3D Point Cloud Segmentation

Few-Shot 3D Point Cloud Segmentation (FS-PCS) aims to predict per-point labels for a query point cloud given a small set of labeled support samples. Following the prototypical paradigm [28], prior FS-PCS methods construct prototypes

to represent the support set information. Given the limited supervision, the expressiveness of these prototypes is crucial to performance. Most existing works adopt the multi-prototype pipeline introduced by [37], where seeds are selected via Farthest Point Sampling (FPS) in coordinate [1, 2] or feature space [37], followed by clustering the surrounding point cloud or multi-modal [1] features to form prototypes. Alternatively, some methods use a single prototype derived from masked average pooling [13].

While these approaches are intuitive, they rely on hand-crafted rules, and most studies [21, 25] focus on adapting such prototypes to the query rather than improving the prototype construction itself. In contrast, we explore a learnable alternative based on attention mechanisms to enhance the flexibility and expressiveness of prototype generation.

2.2 Attention-based Prototype Generation

In the image domain, it is common to construct prototypes using learnable prototypical tokens through cross-attention mechanisms, such as DETR [7] and Mask2Former [8]. In contrast, 3D instance segmentation and object detection typically adopt non-learnable, non-parametric tokens sampled directly from the input point cloud to summarize features via attention [22, 24, 27]. Although some work in 3D domain leverage parametric tokens [31, 33], they rely on auxiliary information such as camera coordinates or 2D image features. This tendency primarily arises from the inherent difficulty of aligning the point cloud distribution with parameter initializations [38]. Through our work, we inspect the underlying matter that induces the misalignment between point cloud features and prototypical tokens, and propose a simple approach that enables attention-based prototype generation in FS-PCS.

3 Preliminary

3.1 Problem Formulation

FS-PCS aims to segment unlabeled point clouds based on a small set of labeled points. Specifically, in each N-way K-shot episode following the well-known meta-learning paradigm [30], it consists of K labeled point clouds for N classes, called the support set $S = \{\{(X^S_{n,k}, Y^S_{n,k})\}^K_{k=1}\}^N_{n=1}$ and U unlabeled ones, named the query set $Q = \{(X^Q_u, Y^Q_u)\}^U_{u=1}$. Here, X and Y represent the point cloud and its corresponding mask, respectively. As a result, the goal of FS-PCS is to predict per-point semantic labels for each query point cloud in Q, leveraging the information in the support set S. Note that training and testing utilize mutually exclusive class sets, $\mathcal{C}_{\text{base}}$ and $\mathcal{C}_{\text{novel}}$, individually, *i.e.*, $\mathcal{C}_{\text{base}} \cap \mathcal{C}_{\text{novel}} = \varnothing$. In the below sections, we assume a 1-way 1-shot setting for better clarity.

3.2 Farthest Point Sampling (FPS)

Given a point cloud $X \in \mathbb{R}^{L\times 3}$, we extract the point-wise features $F \in \mathbb{R}^{L\times D}$, where L and D denote the number of points and the feature dimension, respectively. For simplicity, we assume that both the support and query sets contain L points. We divide the support set into two semantic classes, foreground (FG) and background (BG), based on the ground-truth mask Y^S. Therefore, we obtain the class-specific features $F^C \in \mathbb{R}^{L^C\times D}$ for each class $C \in \{\mathrm{FG}, \mathrm{BG}\}$ within the support features F^S, where L^C denotes the number of points of the class C.

To effectively leverage the information within a few labeled samples, FS-PCS methods typically construct prototypes using the Farthest Point Sampling (FPS) algorithm [1, 2, 21, 37]. The FPS algorithm starts from a randomly selected l-th point feature F_l^C for each class, which serves as the initial element r_1^C of the subset. Then, it iteratively expands the subset $R_t^C = \{r_1^C, \cdots, r_t^C\}$ at each step $t = 1, \cdots, T$, as follows:

$$r_{t+1}^C = \underset{f^C \in F^C \setminus R_t^C}{\arg\max} \big(\min_{r^C \in R_t^C} \left\| f^C - r^C \right\|_2 \big). \tag{1}$$

The resulting subset R_T^C serves as a compact set of representative point features for each class.

4 Motivation

4.1 Attention-based Prototype Generation

Despite the wide adoption of FPS-based prototypes in FS-PCS, it suffers from intrinsic randomness. As detailed in Sec. 3.2, it initiates by randomly selecting a point from the input, then iteratively chooses the farthest point from the previously selected point set. Therefore the resulting set is highly dependent on the random choice, leading to instability in performance, as shown in Tab. 1.

In contrast to algorithmic approaches in FS-PCS, other few-shot downstream tasks [19, 23, 36] have developed prototype generation techniques that leverage the cross-attention mechanism. These attention-based strategies offer notable advantages over the FPS-based ones. First, they generate deterministic outputs, eliminating stochastic effects and ensuring consistent performance. Secondly, the attention mechanisms are fully parallelizable via matrix multiplications, unlike the inherently sequential nature of FPS. Furthermore, they are task-optimizable, as they constitute learnable components rather than fixed sampling heuristics like FPS. These strengths motivate us to adopt attention-based prototype generation in FS-PCS.

Given M prototypical tokens $P = [P_1, P_2, \cdots, P_M]$ where each token $P_m \in \mathbb{R}^D$, we can construct support prototypes $\hat{P}^C \in \mathbb{R}^{M\times D}$ for each class, as follows:

$$\hat{P}_m^C = P_m + \sum_{l \in L^C} \frac{\exp\left(W_q(P_m) W_k(F_l^C)\right)}{\sum_{l' \in L^C} \exp\left(W_q(P_m) W_k(F_{l'}^C)\right)} W_v(F_l^C), \tag{2}$$

Table 2: Distributional discrepancy analysis. P and F^{FG} denote prototypical tokens and support foreground (FG) features, respectively, while $W_q(P)$ and $W_k(F^{\mathrm{FG}})$ are projected versions of them using the attention layer. (a) $\mathrm{Dist}(\cdot,\cdot)$ measures the degree of misalignment between two matrices by computing their average distance, whereas (b) corresponds to the distance between their means. (c) $\mathcal{D}^{\mathrm{instance}}$ represents the feature scale within each instance, while (d) $\kappa^{\mathrm{instance}}$ quantifies anisotropy.

	P	F^{FG}	$W_q(P)$	$W_k(F^{\mathrm{FG}})$
(a) Distance of two matrices $\mathrm{Dist}(\cdot,\cdot)$ (Eq. (3))	282.61		271.18	
(b) Distance of mean $\mathrm{Dist}_\mu(\cdot,\cdot)$ (Eq. (4))	264.17		251.56	
(c) Dipersion $\mathcal{D}^{\mathrm{instance}}$ (Eq. (5))	1.14	94.68	1.12	95.40
(d) Anisotropy $\kappa^{\mathrm{instance}}$ (Eq. (6))	5.36	1122.81	10.41	12392.23

where $W_q(\cdot)$, $W_k(\cdot)$, and $W_v(\cdot)$ are projection layers for the *query*, *key*, and *value*, respectively.

4.2 Attention Misalignment in FS-PCS

To construct representative prototypes using the cross-attention mechanism, it is crucial that the attention in Eq. (2) effectively captures the underlying semantic relationships among the support features. Otherwise, the whole process is prone to yield sub-optimal outputs, often relying heavily on skip connections [11,18,35]. For this to hold, the support features and prototypical tokens must reside in a well-aligned embedding space where semantic similarity is reflected by feature proximity.

In order to assess the robustness of attention-mechanism in FS-PCS, we analyze the vanilla cross-attention module in terms of the distribution in the embedding space of P and F^{FG}. Note that our analysis is derived by averaging across test episodes from the 1-way 1-shot setting on the first split of the S3DIS [3] dataset with a focus on the foreground (FG) class, as the background (BG) often contains multiple semantic categories that could introduce confounding factors.

First, the simplest metric to quantify the discrepancy between the distributions of F^{FG} and P is the average distance, denoted as:

$$\mathrm{Dist}(P, F^{\mathrm{FG}}) = \frac{1}{L^{\mathrm{FG}}} \sum_{l=1}^{L^{\mathrm{FG}}} \min_m \parallel P_m - F_l^{\mathrm{FG}} \parallel_2 . \tag{3}$$

To better understand the contributing factors in $\mathrm{Dist}(\cdot,\cdot)$ presented in Tab. 2 (a), we introduce a related metric that measures the distance between the mean vectors:

$$\mathrm{Dist}_\mu(P, F^{\mathrm{FG}}) = \parallel \mu^P - \mu^{\mathrm{FG}} \parallel_2, \tag{4}$$

where $\mu^{\mathrm{FG}} = \frac{1}{L^{\mathrm{FG}}} \sum_{l=1}^{L^{\mathrm{FG}}} F_l^{\mathrm{FG}} \in \mathbb{R}^{1\times D}$ is the mean of support FG features, and similarly μ^P the mean of P. Tab. 2 (a) and (b) imply that a substantial portion of $\mathrm{Dist}(\cdot,\cdot)$ arises from the discrepancy between the means of F^{FG} and P. Therefore, by aligning the centers, the misalignment is expected to be largely mitigated.

Nevertheless, even with the centers aligned, additional components of misalignment still exist. For instance, if the features F^{FG} exhibit substantially larger dispersion around the center than the tokens P, this scale discrepancy can hinder effective semantic aggregation. Conversely, pronounced anisotropy in the feature directions may still restrict the tokens from fully capturing the semantic context, despite comparable scale. To inspect these factors, we first define the within-instance dispersion $\mathcal{D}^{\mathrm{instance}}$, to describe the scale of the feature space:

$$\mathcal{D}^{\mathrm{instance}} = \frac{1}{L^{\mathrm{FG}}} \sum_{l=1}^{L^{\mathrm{FG}}} \parallel F_l^{\mathrm{FG}} - \mu^{\mathrm{FG}} \parallel_2 . \tag{5}$$

Next, to compare the geometric differences, we measure instance-wise anisotropy $\kappa^{\mathrm{instance}}$, defined as follows:

$$\begin{aligned} \kappa^{\mathrm{instance}} &= \Sigma_{11}^{\mathrm{FG}} / \min\{\Sigma_{ii}^{\mathrm{FG}} \mid 1 \le i \le r, 0 < \Sigma_{ii}^{\mathrm{FG}}\}, \\ \mathrm{diag}\left(\Sigma^{\mathrm{FG}}\right) &= \left[\Sigma_{11}^{\mathrm{FG}}, \Sigma_{22}^{\mathrm{FG}}, \cdots, \Sigma_{rr}^{\mathrm{FG}}\right], \quad r = \min\left(L^{\mathrm{FG}}, D\right), \end{aligned} \tag{6}$$

where $\mathrm{diag}(\cdot)$ extracts the diagonal elements, and $\Sigma^{\mathrm{FG}} \in \mathbb{R}^{L^{\mathrm{FG}}\times D}$ is the diagonal matrix of singular values resulting from the SVD of the centered features $F^{\mathrm{FG}} - \mu^{\mathrm{FG}}$. Intuitively, a large $\kappa^{\mathrm{instance}} > 1$ indicates that the features are dominated by a few principal directions, whereas $\kappa^{\mathrm{instance}} \approx 1$ suggests an approximately spherical distribution. The same procedure is applied to P, with detailed derivations provided in the supplementary material.

As shown in Tab. 2 (c) and (d), the difference between $\mathcal{D}^{\mathrm{instance}}$ and $\kappa^{\mathrm{instance}}$ from P and F^{FG} is evident, indicating mismatches in feature scale and direction. This discrepancy is not resolved with the projection layers preceding the cross-attention mechanisms; hence, the cross-attention struggles to capture the semantically meaningful relationships due to extreme misalignment. Such large distributional gaps between prototypical tokens and support features not only hinder semantic correspondence during cross-attention, but also impede optimization [26]. This observation motivates us to project support features into a more coherent and structured representation space for effective attention-based prototype generation.

5 Method

5.1 Overview

In this section, we present WARM, a cross-attention-based prototype generation framework for FS-PCS that incorporates a simple alignment strategy for improved semantic aggregation. As shown in Fig. 2, WARM comprises three stages:

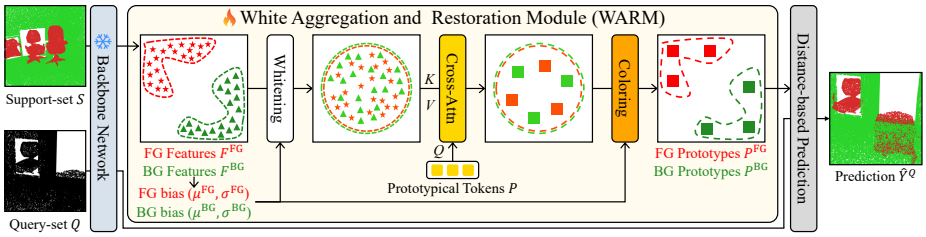

Fig. 2: Overall pipeline of WARM in the 1-way 1-shot setting. Support and query features are extracted using a pretrained network. Within WARM, support features are divided into foreground and background components, which are whitened to align with the prototypical tokens and aggregated via cross-attention to generate prototypes. The prototypes are then recolored to restore instance-specific statistics. Finally, query points are classified based on their similarity to the prototypes.

whitening, cross-attention, and coloring. Support features are first whitened to decouple distributional biases and align them with prototypical tokens. Cross-attention then aggregates the aligned features to form prototypes, which are subsequently colored with the instance-specific statistics of the features. Finally, segmentation is performed by assigning each query point to its nearest prototype. Overall, WARM achieves effective feature–prototype alignment while preserving essential instance characteristics.

5.2 White Aggregation and Restoration Module

As detailed in Sec. 4.2, the vanilla cross-attention is insufficient to aggregate point features based on their semantic relationships. To address this issue, we suggest the incorporation of alignment and restoration techniques with the cross-attention mechanism. Specifically, for each class $C \in \{\mathrm{FG}, \mathrm{BG}\}$ in the support set, the covariance matrix $\mathcal{S}^C \in \mathbb{R}^{D\times D}$ is computed as:

$$\mathcal{S}^C = \frac{1}{L^C - 1}(F^C - \mu^C)^T(F^C - \mu^C). \tag{7}$$

Then, we apply ZCA whitening [5, 14] to F^C as follows:

$$Z^C = (F^C - \mu^C) \cdot (\mathcal{S}^C)^{-\frac{1}{2}}, \tag{8}$$

where $Z^C \in \mathbb{R}^{L^C\times D}$ denotes the whitened features with zero mean and decorrelated channels, *i.e.*, ${Z^C}^\top \cdot Z^C = I$. By normalizing instance-specific statistics, the whitened features reside in a standardized space, where prototypical tokens are aligned with the point features. With the whitened support features Z^C, we generate prototypes $\tilde{P}^C \in \mathbb{R}^{M\times D}$, which replace those in Eq. (2), as follows:

$$\tilde{P}^C_m = P_m + \mathrm{CA}(P_m, Z^C). \tag{9}$$

This facilitates stable aggregation based on semantics, as the attention mechanism no longer needs to account for instance-specific distributional variations that lead to isolated pairwise matching.

Although whitening discards statistics that would otherwise hinder alignment during attention, prototypes representing support features should retain these statistics to preserve the original expressiveness. To this end, we introduce a coloring step after cross-attention. This can be simply implemented as the inverse of the ZCA whitening in Eq. (8), as follows:

$$P^C = \tilde{P}^C \cdot (\mathcal{S}^C)^{\frac{1}{2}} + \mu^C, \tag{10}$$

where $P^C \in \mathbb{R}^{M \times D}$ denotes the final prototypes that incorporate complete information. As such, we obtain representative prototypes that are derived by considering semantic relationships within point clouds while restoring instance-specific distributional statistics.

5.3 Regularized Whitening for Stability

In order to apply whitening as in Sec. 5.2, inverse square root of $\mathcal{S}^C$ is required, which is obtained by inverting the square root of its eigendecomposition:

$$(\mathcal{S}^C)^{-\frac{1}{2}} = (V \Lambda V^T)^{-\frac{1}{2}} = V \Lambda^{-\frac{1}{2}} V^T. \tag{11}$$

The diagonal entries $\{\lambda_0, \lambda_1, ...\lambda_D\}$ of the diagonal matrix Λ are the eigenvalues of $\mathcal{S}^C$ and the columns of V are the corresponding eigenvectors.

However in FS-PCS, where the number of input points is dynamic, the covariance matrix $\mathcal{S}^C$ is possibly ill-conditioned in the case of sparse points due to linear dependencies. In such cases, even a small variance in a certain direction that is most likely a noise, can be falsely amplified as a signal in the whitened space. Therefore, to ensure stability and to prevent such noise-amplification, we regularize the eigenvalues:

$$\Lambda \leftarrow (\mathbb{1}_{\lambda > \nu} \lambda + \mathbb{1}_{\lambda \leq \nu} \nu) \cdot I, \; \nu = \lambda_{\text{median}} \left(1 + \sqrt{D / L^C}\right)^2, \tag{12}$$

where the threshold ν is set to the upper bound of estimated noise [4], with λ_{median} representing the median eigenvalue. By thresholding the eigenvalues adaptively to the point cloud size, the inversion process is stabilized even in sparse scenes. Further details are provided in supplementary.

5.4 Training Objective and Inference

We adopt a basic segmentation approach, in which each query point is assigned the class of its nearest prototype. The prediction $\hat{Y}_l^Q$ for the l-th query point features F_l^Q is determined by the distance to the nearest prototype within each class $C \in \{\text{FG}, \text{BG}\}$:

$$\hat{Y}_l^Q = \arg\min_C \left(\min_m \| F_l^Q - P_m^C \|_2 \right). \tag{13}$$

For supervision, we use a margin loss with the margin set to zero, to penalize only the wrong predictions:

$$\mathcal{L}_{\text{margin}} = \frac{1}{L}\sum_{l=1}^{L}\Big(\mathbb{1}_{Y_l^Q=\text{FG}}\max\left(d_l^{\text{FG}} - d_l^{\text{BG}}, 0\right) + \mathbb{1}_{Y_l^Q=\text{BG}}\max\left(d_l^{\text{BG}} - d_l^{\text{FG}}, 0\right)\Big). \tag{14}$$

where $\mathbb{1}_{\text{condition}}$ is the indicator function, which equals 1 if the condition is true and 0 otherwise. To further prevent the multi-prototypes from collapsing into trivial representations, we additionally apply a simplification loss [17], as:

$$\mathcal{L}_{\text{sim}} = \frac{1}{N+1}\sum_{C}\Big(\frac{1}{L^C}\sum_{l=1}^{L^C}\min_m d_{lm}^C + \frac{1}{M}\sum_{m=1}^{M}\min_l d_{lm}^C + \max_m \min_l d_{lm}^C\Big), \tag{15}$$

where $d_{lm}^C = \| F_l^C - P_m^C \|_2$. The loss is computed separately for FG and BG.

As a result, the overall training objective is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{margin}} + \alpha \cdot \mathcal{L}_{\text{sim}}, \tag{16}$$

where α is a coefficient hyperparameter, set to 0.5.

6 Experiments

6.1 Experimental Results

Baselines. FS-PCS studies are conducted under two distinct strategies of preprocessing raw point clouds, 1) *foreground oversampling* [13, 25, 37] and 2) *uniform sampling* [1, 2]. As noted in [2], the former relies on ground-truth masks to oversample foreground regions, limiting its use in real-world settings where annotations are unavailable. Therefore, we adopt and compare with the baselines under the uniform sampling protocol [1,2]. In addition, we report results for baselines originally developed under the foreground oversampling protocol [13,25,37], re-implemented within the uniform sampling setting, provided in [2].

Datasets. We perform experiments on two benchmark datasets for FS-PCS: S3DIS [3] and ScanNet [9]. S3DIS comprises 271 indoor scenes with 12 semantic classes, and ScanNet contains 1,513 indoor scenes with annotations for 20 categories. For both datasets, the classes are split into two folds, S_0 and S_1, for cross-validation, where they do not overlap with each other. Following prior work [37], each scene is divided into 1m×1m blocks, with each block valid only if it contains $\geq 5\%$ foreground points, with a 100 point minimum. We adopt the same data preprocessing as in [2], where each block is voxelized into 0.02m grids, and uniformly sampled up to 20,480 points.

Table 3: Quantitative result on S3DIS in mIoU (%). Combinations of 1/2-way, 1/5-shot setting are reported, for splits S_0 and S_1, including their mean. The best and second-best are **bolded** and underlined, respectively. † denotes reproduced performance without foreground leakage with Stratified Transformer [16] as backbone, provided in [2]. $*$ denotes methods that utilize additional modality.

Method	1-way 1-shot			1-way 5-shot			2-way 1-shot			2-way 5-shot			Overall
	S_0	S_1	Mean	S_0	S_1	Mean	S_0	S_1	Mean	S_0	S_1	Mean	
AttMPTI† [37]	36.32	38.36	37.34	46.71	42.70	44.71	31.09	29.62	30.36	39.53	32.62	36.08	37.12
QGE† [25]	41.69	39.09	40.39	50.59	46.41	48.50	33.45	30.95	32.20	40.53	36.13	38.33	39.86
QGPA† [13]	35.50	35.83	35.67	38.07	39.70	38.89	25.52	26.26	25.89	30.22	32.41	31.32	32.94
COSeg [2]	46.31	48.10	47.21	51.40	48.68	50.04	37.44	36.45	36.95	42.27	38.45	40.36	43.64
MM-FSS* [1]	49.84	**54.33**	52.09	51.95	56.46	54.21	41.98	**46.61**	44.30	46.02	**54.29**	50.16	50.19
FPS + *min-dist.*	51.35	45.68	48.52	68.43	61.51	64.97	44.48	33.00	38.74	58.22	45.52	51.87	51.03
WARM (ours)	**60.16**	50.50	**55.33**	**72.23**	**62.66**	**67.45**	**50.91**	38.34	**44.63**	**61.46**	48.95	**55.21**	**55.66**

Table 4: Quantitative result on ScanNet in mIoU (%). Experimental setting is the same as Tab. 3, performed on ScanNet [9].

Method	1-way 1-shot			1-way 5-shot			2-way 1-shot			2-way 5-shot			Overall
	S_0	S_1	Mean	S_0	S_1	Mean	S_0	S_1	Mean	S_0	S_1	Mean	
AttMPTI† [37]	34.03	30.97	32.50	39.09	37.15	38.12	25.99	23.88	24.94	30.41	27.35	28.88	31.11
QGE† [25]	37.38	33.02	35.20	45.08	41.89	43.49	26.85	25.17	26.01	28.35	31.49	29.92	33.66
QGPA† [13]	34.57	33.37	33.97	41.22	38.65	39.94	21.86	21.47	21.67	30.67	27.69	29.18	31.19
COSeg [2]	41.73	41.82	41.78	48.31	44.11	46.21	28.72	28.83	28.78	35.97	33.39	34.68	37.86
MM-FSS* [1]	**46.08**	**43.37**	**44.73**	**54.66**	45.48	50.07	**43.99**	**34.43**	**39.21**	**48.86**	39.32	**44.09**	**44.53**
FPS + *min-dist.*	37.95	33.66	35.81	52.77	47.72	50.25	30.13	27.06	28.60	43.41	39.35	41.38	39.01
WARM (ours)	41.16	39.10	40.13	53.40	**49.04**	**51.22**	30.27	30.02	30.15	45.02	**41.61**	43.32	41.21

Implementation Details. We use the encoder layers of Stratified Transformer [16] as the feature extractor, with pretrained weights from [2]. For prototype generation, we use a single WARM layer with 100 prototypical tokens for each foreground and background, totaling 200 tokens. In a multi-shot setting $K > 1$, we generate prototypes for each sample and average them across shots as in [28]. Evaluation is based on mean Intersection-over-Union (mIoU), and is performed on 1,000 episodes per class in the 1-way setting and 100 episodes per class combination for the 2-way setting to ensure stable results.

For training, AdamW optimizer with a learning rate of 1×10^{-4} and a weight decay of 0.01 is applied. Training is conducted for 10 and 20 epochs for the N-way 1-shot and 5-shot settings, respectively, where each epoch consists of 400 episodes. The learning rate is decayed by a factor of 0.1 at 60% and 80% of the total number of epochs. All experiments are performed on an RTX A6000 GPU.

Quantitative Result. We compare WARM with existing baselines under 1/2-way and 1/5-shot settings. As shown in Tab. 3, WARM achieves state-of-the-art performance for uni-modal methods on the S3DIS dataset. This superiority is mostly maintained on the ScanNet dataset in Tab. 4 for uni-modal models, although it has lower performance in most scenarios compared to multi-modal method such as MM-FSS [1]. MM-FSS exploits text and image features in addition to point clouds for more comprehensive pretraining [20], whereas the main

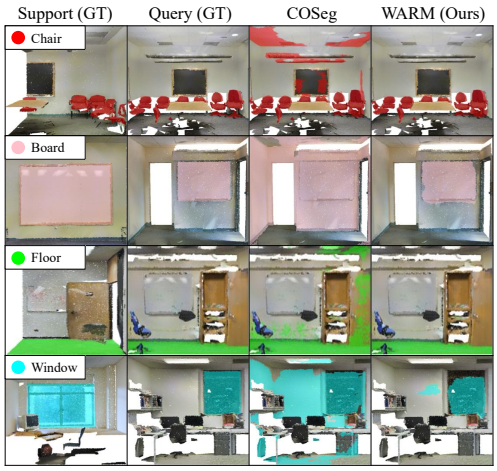

Fig. 3: Qualitative results of COSeg [2] and WARM on 1-way 1-shot setting of the S3DIS [3] dataset. Each row denotes a single test episode, where ground-truths (GT) and predictions are highlighted with their corresponding class color.

Table 5: Component ablation study. **N**, **W**, and **R** denote normalization, whitening, and coloring. Normalization means scaling the features to zero mean and unit variance along each channel dimension. Dist(Q, K) indicates the distance between queries and keys as in Eq. (3).

	N	**W**	**R**	Dist(Q, K) ↓	mIoU (%)
(a)				288.81	47.29
(b)	✓			13.30	14.18
(c)		✓		13.14	10.83
(d)	✓		✓	13.72	57.59
(e)		✓	✓	**13.13**	**60.16**

Table 6: Loss ablation study.

	$\mathcal{L}_{\text{margin}}$	$\mathcal{L}_{\text{sim}}$	mIoU (%)
Naïve CA	✓	✓	47.29
WARM	✓		59.34
		✓	58.74
	✓	✓	**60.16**

contribution of WARM is focused on the prototype construction, rather than feature quality.

Given the simplicity of WARM compared to heavy decoder-based models [1,2], its competitive performance highlights the importance of careful prototype design. Moreover, the strong results achieved by 'FPS + *min-dist.*' further question the necessity of complex decoders. These findings suggest that future work should place greater emphasis on developing more expressive and robust prototype representations, as explored in other few-shot downstream tasks [19, 36].

Qualitative Result. In addition to the quantitative results, we display qualitative results in Fig. 3. Compared to COSeg [2], WARM consistently reduces mispredictions and increases correct classifications across episodes, while maintaining a simpler segmentation architecture. Along with quantitative results, this demonstrates the importance of prototype integrity, rather than focusing on complex decoders.

6.2 Ablation Study

Experiments are conducted under the 1-way 1-shot setting with S_0 of S3DIS [3].

Component Ablation. Tab. 5 presents a component-wise ablation study verifying the effectiveness of each alignment step. In (a), the naïve cross-attention (CA) suffers from severe misalignment between query and key features, as reflected in the large Dist(Q, K). We address this via whitening-based alignment,

Table 7: Quantitative comparison of attention maps. Entropy (normalized to the range $[0, 1]$) measures the uniformity of each attention distribution. Diversity is computed as the inverse of average similarity among the attention maps of the prototypical tokens, reflecting how distinctly they focus on different regions.

Method	Entropy	Diversity
Cross-Attention	0.2079	0.8824
WARM	0.8387	0.6442

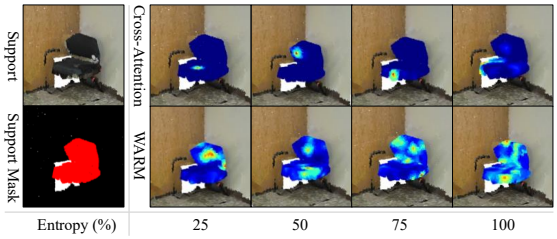

Fig. 4: Qualitative comparison of attention maps. To ensure a fair assessment, each attention map is sampled based on its respective ranking within the entropy score distribution.

with comparisons against normalization, an alternative alignment method. As shown in (b), normalization standardizes features to unit variance per channel, substantially reducing misalignment, but remains sub-optimal as it preserves the covariance structure. In contrast, whitening (c) enforces an isotropic distribution by removing the covariance, further decreasing the misalignment. However, while both methods enable cross-attention to capture semantic relationships, they also strip essential distributional characteristics, causing prototypes to lose the original instance's inherent semantics, as reflected in their low mIoU. Therefore, coloring (d–e) should be accompanied to restore these properties after alignment. The full model in (e) with whitening and coloring achieves the best performance, confirming that both feature alignment and semantic restoration are necessary for faithful prototype representation.

Loss Ablation. As shown in Tab. 6, WARM substantially outperforms CA, even with only the task objective $\mathcal{L}_{\text{margin}}$, indicating that WARM's architecture itself mitigates attention degradation. Combining $\mathcal{L}_{\text{margin}}$ and $\mathcal{L}_{\text{sim}}$ achieves the best result, suggesting complementary effects.

Semantic Attention by Whitening. We validate that whitening enables prototypical tokens to aggregate features by semantic relationship, rather than spatial proximity in the projected space, through quantitative and qualitative analysis. We define two metrics: entropy, quantifying attention distribution uniformity (normalized to $[0, 1]$), and diversity, quantifying inter-prototype distinctiveness as the inverse of average attention-map similarity. As shown in Tab. 7, naïve cross-attention exhibits low entropy and high diversity, indicating that prototypes attend to features close in projection space but lacking semantic relevance, causing attention to collapse into point-level aggregation without semantic grouping. WARM, by contrast, achieves substantially higher entropy alongside a moderate but sufficient diversity score, indicating that each prototype attends broadly and consistently to semantically related regions, yielding richer and more coherent representations. Qualitative results in Fig. 4 corroborate this:

Table 8: Parametric capacity ablation. Results (a-c) indicate that increasing the parameter has minimal effect on performance, whereas the substantial gain in (d) suggests that structural design is the primary determinant of performance.

	Method	mIoU	Backbone param. #	Extra param. #
(a)	FPS + *min-dist.*	51.35	3.91M	-
		52.34	5.16M	-
(b)	COSeg [2]	46.31	3.91M	2.20M
	MM-FSS [1]	49.84	7.55M	7.22M
(c)	CA	47.29	3.91M	1.13M
(d)	WARM	**60.16**	3.91M	1.13M

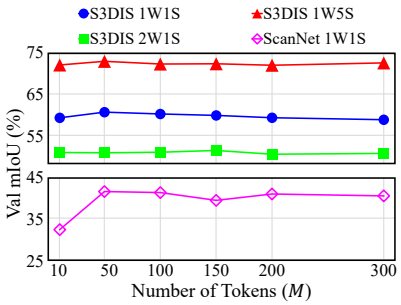

Fig. 5: Number of prototypical tokens (M) ablation for different settings.

naïve attention is spatially narrow, whereas WARM captures semantically coherent point groupings. Visualizations are sampled by entropy percentile rather than cherry-picked results.

Compatibility with FS-PCS Methods. To evaluate the versatility of WARM as a general-purpose prototype generation module, we integrate it into existing FS-PCS frameworks in the uniform sampling protocols, such as COSeg [2] and MM-FSS [1]. Specifically, we replace their standard FPS-based prototype generation with WARM. Notably, the mIoU of COSeg increases from 46.31% to 59.26%, representing a substantial 12.95% improvement. Similarly, MM-FSS shows a steady gain, rising from 49.84% to 52.62%. These results demonstrate that WARM is a robust, plug-and-play solution that effectively substitutes conventional heuristic-based sampling. Additional experiments on the foreground oversampling baselines are provided in the supplementary.

Parametric Capacity Ablation. To verify that WARM's gain comes from its structured design rather than additional parametric capacity, we analyze the performance of existing methods with respect to their parameter counts. In Tab. 8, (a) increasing the backbone capacity of 'FPS+*min-dist.*' from 3.91M to 5.16M brings only marginal gains in performance, while (b) existing FPS-based methods underperform WARM despite using comparable or larger backbones with much larger additional modules. These results imply that additional module capacity does not directly improve performance. Conclusively, (c) shows that even the same structure as WARM performs poorly without whitening/coloring, while (d) confirms superior performance. To sum up, WARM's gain mainly comes from resolving the misalignment between learnable tokens and support features through whitening and coloring, rather than from simply adding parameters.

Number of Tokens. We provide ablation on M in Fig. 5 across different datasets and settings. For S3DIS, the performance remains stable when varying

M, indicating that increasing the number of tokens does not consistently improve performance. However in ScanNet, the performance largely drops at small value of $M = 10$, likely due to its larger number of FG classes requiring higher representational capacity, and stabilizing once $M \geq 50$. In conclusion, the results suggest that more complex datasets may require a sufficient number of tokens for adequate representational capacity, although performance becomes stable once M exceeds a moderate threshold.

Memory and Runtime Overhead. We compare peak memory and runtime of WARM against FPS-based prototype generation. For constructing 100 prototypes, we vary the point count (L =1K∼25K; D=100) and the feature dimension (D =100∼2K; L=1K), one factor at a time. As shown in Tab. 9, WARM uses slightly more peak memory due to the attention parameters, but is significantly faster than FPS. This is because FPS is inherently sequential, whereas WARM is dominated by GPU-parallel batched matrix multiplications. Overall, the computational efficiency of the attention module outweighs the additional cost incurred by whitening, thereby maintaining the practical runtime feasibility of WARM.

Table 9: Runtime and memory overhead of WARM compared to FPS.

Overhead	Method	Point Count			Feature Dimension		
		1K	5K	25K	100	500	2K
Memory ↓ (MB)	FPS	39.4	196.5	983.6	39.4	193.8	772.9
	WARM	42.6	204.3	1016.1	42.6	216.5	974.2
Runtime ↓ (ms)	FPS	9.9	20.8	78.3	10.1	20.5	87.8
	WARM	2.8	2.8	6.0	2.8	16.2	85.6

7 Conclusion

In this paper, we investigated prototype generation for FS-PCS, addressing the limitations of conventional approaches in FS-PCS settings. We observed that even a widely adopted mechanism struggles to bridge the distributional gap when trained with limited samples, motivating our design of an enhanced cross-attention module that incorporates whitening and coloring transformations. Through this design, we obtained representative prototypes by decoupling detrimental factors while preserving originality, thus achieving competitive performance across FS-PCS benchmarks even without the aid large decoders. Pointing out the underexplored impact of prototype generation, we hope our work offers a complementary direction for FS-PCS.

Acknowledgements

This work was supported in part by MSIT/IITP (No. RS-2022-II220680, RS-2020-II201821, RS-2019-II190421, RS-2024-00459618, RS-2024-00360227, RS-2024-00437633, RS-2024-00437102, RS-2025-25442569), MSIT/NRF (No. RS-2024-00357729), KNPA/KIPoT (No. RS-2025-25393280), and SEMES-SKKU collaboration funded by SEMES.

References

1. An, Z., Sun, G., Liu, Y., Li, R., Wu, M., Cheng, M.M., Konukoglu, E., Belongie, S.: Multimodality helps few-shot 3d point cloud semantic segmentation. In: ICLR (2025)
2. An, Z., Sun, G., Liu, Y., Liu, F., Wu, Z., Wang, D., Van Gool, L., Belongie, S.: Rethinking few-shot 3d point cloud semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3996–4006 (2024)
3. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3d semantic parsing of large-scale indoor spaces. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1534–1543 (2016)
4. Bai, Z.D., Yin, Y.Q., et al.: Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. Ann. Probab **21**(3), 1275–1294 (1993)
5. Bell, A.J., Sejnowski, T.J.: The "independent components" of natural scenes are edge filters. Vision research **37**(23), 3327–3338 (1997)
6. Betsas, T., Georgopoulos, A., Doulamis, A., Grussenmeyer, P.: Deep learning on 3d semantic segmentation: A detailed review. Remote Sensing **17**(2), 298 (2025)
7. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
8. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1290–1299 (2022)
9. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017)
10. Daneshmand, H., Kohler, J., Bach, F., Hofmann, T., Lucchi, A.: Batch normalization provably avoids ranks collapse for randomly initialised deep networks. Advances in Neural Information Processing Systems **33**, 18387–18398 (2020)
11. Dong, Y., Cordonnier, J.B., Loukas, A.: Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In: International conference on machine learning. pp. 2793–2803. PMLR (2021)
12. Fan, Q., Pei, W., Tai, Y.W., Tang, C.K.: Self-support few-shot semantic segmentation. In: European conference on computer vision. pp. 701–719. Springer (2022)
13. He, S., Jiang, X., Jiang, W., Ding, H.: Prototype adaption and projection for few- and zero-shot 3d point cloud semantic segmentation. IEEE Transactions on Image Processing (2023)
14. Huang, L., Yang, D., Lang, B., Deng, J.: Decorrelated batch normalization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 791–800 (2018)
15. Kovaleva, O., Romanov, A., Rogers, A., Rumshisky, A.: Revealing the dark secrets of bert. arXiv preprint arXiv:1908.08593 (2019)
16. Lai, X., Liu, J., Jiang, L., Wang, L., Zhao, H., Liu, S., Qi, X., Jia, J.: Stratified transformer for 3d point cloud segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8500–8509 (2022)
17. Lang, I., Manor, A., Avidan, S.: Samplenet: Differentiable point cloud sampling. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7578–7588 (2020)

18. Lee, M., Kim, J., Heo, J.P.: Activating self-attention for multi-scene absolute pose regression. Advances in Neural Information Processing Systems **37**, 38508–38529 (2024)
19. Lee, S., Moon, W., Seong, H.S., Heo, J.P.: Temporal alignment-free video matching for few-shot action recognition. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5412–5421 (2025)
20. Li, B., Weinberger, K.Q., Belongie, S., Koltun, V., Ranftl, R.: Language-driven semantic segmentation. In: International Conference on Learning Representations (2022)
21. Li, Z., Wang, Y., Li, W., Sun, R., Zhang, T.: Localization and expansion: A decoupled framework for point cloud few-shot semantic segmentation. In: European Conference on Computer Vision. pp. 18–34. Springer (2024)
22. Liu, H., Cai, M., Lee, Y.J.: Masked discrimination for self-supervised learning on point clouds. In: European Conference on Computer Vision. pp. 657–675. Springer (2022)
23. Liu, Y., Zhang, X., Zhang, S., He, X.: Part-aware prototype network for few-shot semantic segmentation. In: European conference on computer vision. pp. 142–158. Springer (2020)
24. Misra, I., Girdhar, R., Joulin, A.: An end-to-end transformer model for 3d object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2906–2917 (2021)
25. Ning, Z., Tian, Z., Lu, G., Pei, W.: Boosting few-shot 3d point cloud segmentation via query-guided enhancement. In: Proceedings of the 31st ACM international conference on multimedia. pp. 1895–1904 (2023)
26. Santurkar, S., Tsipras, D., Ilyas, A., Madry, A.: How does batch normalization help optimization? Advances in neural information processing systems **31** (2018)
27. Schult, J., Engelmann, F., Hermans, A., Litany, O., Tang, S., Leibe, B.: Mask3d: Mask transformer for 3d semantic instance segmentation. arXiv preprint arXiv:2210.03105 (2022)
28. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. Advances in neural information processing systems **30** (2017)
29. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
30. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. Advances in neural information processing systems **29** (2016)
31. Wang, Y., Guizilini, V.C., Zhang, T., Wang, Y., Zhao, H., Solomon, J.: Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In: Conference on robot learning. pp. 180–191. PMLR (2022)
32. Xiao, A., Huang, J., Guan, D., Zhang, X., Lu, S., Shao, L.: Unsupervised point cloud representation learning with deep neural networks: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(9), 11321–11339 (2023)
33. Xie, Y., Jiang, H., Gkioxari, G., Straub, J.: Pixel-aligned recurrent queries for multi-view 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18370–18380 (2023)
34. Yang, J., Zhang, Q., Ni, B., Li, L., Liu, J., Zhou, M., Tian, Q.: Modeling point clouds with self-attention and gumbel subset sampling. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3323–3332 (2019)

35. Zhai, S., Likhomanenko, T., Littwin, E., Busbridge, D., Ramapuram, J., Zhang, Y., Gu, J., Susskind, J.M.: Stabilizing transformer training by preventing attention entropy collapse. In: International Conference on Machine Learning. pp. 40770–40803. PMLR (2023)
36. Zhang, J.W., Sun, Y., Yang, Y., Chen, W.: Feature-proxy transformer for few-shot segmentation. Advances in neural information processing systems **35**, 6575–6588 (2022)
37. Zhao, N., Chua, T.S., Lee, G.H.: Few-shot 3d point cloud semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8873–8882 (2021)
38. Zhou, X., Liang, D., Xu, W., Zhu, X., Xu, Y., Zou, Z., Bai, X.: Dynamic adapter meets prompt tuning: Parameter-efficient transfer learning for point cloud analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14707–14717 (2024)