Generalized Few-Shot Point Cloud Segmentation Via Geometric Words

Yating Xu\(^1\)  Conghui Hu\(^1\)  Na Zhao\(^2\)  Gim Hee Lee\(^1\)

\(^1\)Department of Computer Science, National University of Singapore
\(^2\)Singapore University of Technology and Design

xu.yating@u.nus.edu  conghui@nus.edu.sg  na.zhao@sutd.edu.sg  gimhee.lee@nus.edu.sg

Abstract

Existing fully-supervised point cloud segmentation methods suffer in the dynamic testing environment with emerging new classes. Few-shot point cloud segmentation algorithms address this problem by learning to adapt to new classes at the sacrifice of segmentation accuracy for the base classes, which severely impedes its practicality. This largely motivates us to present the first attempt at a more practical paradigm of generalized few-shot point cloud segmentation, which requires the model to generalize to new categories with only a few support point clouds and simultaneously retain the capability to segment base classes. We propose the geometric words to represent geometric components shared between the base and novel classes, and incorporate them into a novel geometric-aware semantic representation to facilitate better generalization to the new classes without forgetting the old ones. Moreover, we introduce geometric prototypes to guide the segmentation with geometric prior knowledge. Extensive experiments on S3DIS and ScanNet consistently illustrate the superior performance of our method over baseline methods. Our code is available at: https://github.com/Pixie8888/GFS-3DSeg_GWs.

1. Introduction

Point cloud segmentation aims at predicting the category of each point in the 3D scenes represented by the point cloud, and has wide applications in autonomous driving, robotics, etc. Although fully-supervised point cloud segmentation methods (Full-3DSeg) \cite{19, 20, 28, 11, 9, 29} have achieved impressive performance, they heavily require large-scale annotated training data and rely on the closed set assumption that the class distribution of the testing point cloud remains the same as the training dataset. However, the closed set assumption is unrealistic in the open world where new classes arise continuously. The Full-3DSeg in this challenging open-world setting thus requires large amounts of annotated data for new classes, which are time-consuming and expensive to collect.

The few-shot point cloud segmentation (FS-3DSeg) \cite{35, 14, 8} algorithms are designed to ameliorate the lack of data for novel class adaptation. Generally, FS-3DSeg first trains a model with abundant training samples of the base classes, and then targets at segmenting the new classes by learning from only a small number of samples from the corresponding new classes. By adopting episodic training \cite{26} to mimic the testing environment and specific designs for feature extraction \cite{35}, FS-3DSeg achieves promising novel class segmentation results for the query point clouds. However, in the task of point cloud segmentation, base and novel classes often appear together in one scene (see Figure 5 as examples). An ideal segmentor in practice is expected to give each point in the scene a semantic label. As a result, the FS-3DSeg setting that only segments points of novel classes while ignoring the base classes suffers limited practicality.

In view of the impracticality of FS-3DSeg, we introduce the generalized few-shot point cloud segmentation (GFS-3DSeg) task. As shown in Tab. 1, given the model originally trained on base classes, the objective of GFS-3DSeg is to segment both base and novel classes using merely a limited
number of labeled samples for the new classes during testing. Furthermore, there is no access to the base training data during testing in GFS-3DSeg considering the issues of data privacy and storage memory limitations. In this demanding but practical context, we expect a good generalized few-shot point cloud segmentor to effectively learn to segment novel classes with few samples and also maintain the knowledge of the base classes. A potential solution to GFS-3DSeg can be using prototype learning [25, 21] to generate class prototypes of the base and novel classes as the classifier weights, which can quickly adapt to new classes and avoid forgetting past knowledge caused by fine-tuning. However, effective learning of the representation and classifier for the novel classes is not trivial and remains as a major challenge.

In this paper, we propose geometric words (GWs) as the transferable knowledge obtained from the base classes to enhance the learning of the new classes without forgetting the old ones. Although different (old and new) classes contain distinct semantic representations, they usually share similar local geometric structures as shown in Fig. 1. Based on this observation, we first mine the representation for the local geometric structures from the pretrained features of the base classes and store them as the geometric words to facilitate learning of the new classes with few examples. We then learn a geometric-aware semantic representation based on the geometric words. Specifically, the geometric-aware semantic representation is a fusion of two features: 1) Class-agnostic geometric feature obtained by the assignment of low-level features to their most similar geometric words, 2) Class-specific semantic feature which is the output of a feature extractor. Intuitively, our geometric-aware semantic representation allows the encoding of the transferable geometric information across classes while preserving the semantic information for effective segmentation.

We further introduce geometric prototype (GP) to supplement the original semantic prototype in the prototype learning. Particularly, the geometric prototype refers to the frequency histogram of GWs that can uniquely represent each class from the global geometric perspective despite GWs are class agnostic. We thus leverage GP to propose a geometric-guided classifier re-weighting module for the rectification of biased predictions originating from semantic prototypes. Specifically, we first perform minor frequency pruning on the geometric prototypes to suppress noisy responses of the geometric words for each class. Subsequently, we measure the geometric matching score between each query point and the pruned geometric prototypes, and employ these scores as geometric-guided weights. By re-weighting the semantic logits with geometric-guided weights, our final prediction is enriched with geometric information that is transferable across classes. Such transferable information helps to facilitate the segmentation of new classes that only have a few samples while preserving the knowledge of base classes. Our main contributions can be summarized as:

- We are the first to study the important generalized few-shot point cloud segmentation task, which is more practical than its counterparts of fully-supervised and few-shot setting in the dynamic testing environment.
- We propose geometric words to represent diverse basic geometric structures that are shared across different classes, and a geometric-aware semantic representation that allows for generalizable knowledge encoding.
- We introduce geometric prototype to supplement the semantic prototype. We design a geometric-guided classifier re-weighting module comprising minor frequency pruning to dynamically guide the segmentation of novel classes with geometric similarity.
- We conduct extensive experiments on S3DIS and ScanNet to verify the effectiveness of our method. Specifically, our method improves over the state-of-the-art FS-3DSeg method by 6 and 8 times on the 5-shot and 1-shot settings of ScanNet, respectively.

### 2. Related Work

**Few-shot Learning.** Few-shot learning aims at classification of novel classes by learning with only a few labeled samples during testing stage. There are mainly three types of approaches: meta-learning [5, 15, 16, 10], metric-learning [23, 26, 32, 18] and transfer learning based methods [24, 13]. Meta-learning based methods aim to predict a set of parameters that can be quickly adapted to new tasks. Metric-learning methods learn a mapping from images to an embedding space where images of the same class are clustered together. Transfer learning methods aim to learn a general feature representation during base class training, and then train a classifier for each new task.

Generalized few-shot learning is first proposed in [7], which aims to classify both base and novel classes after learning from the few-shot training samples of novel classes during testing stage. Gidaris et al. [6] and Qi et al. [21] introduce cosine similarity between image feature and class-
classifier weight as the classifier, which unifies the recognition of both novel and base categories. To reduce the biased learning of base classes, Gidaris et al. [31] improve the classifier of the novel classes by utilizing the knowledge of base classes. Gidaris et al. [6] propose few-shot classification weight generator to perform attention over the classification weight vectors of the base classes. Ye et al. [31] synthesize calibrated few-shot classifiers with a shared neural dictionary learned in the base class training stage. In this work, we study generalized few-shot point cloud segmentation, a more challenging task as it targets at dense 3D point-level classification. Instead of fusion classifier weights of novel classes with past knowledge [6, 31], we use geometric prototype as an additional classifier weight to help calibrate the biased prediction. Moreover, we propose geometric-aware semantic representation to facilitate the representation learning for novel classes.

**Few-shot Semantic Segmentation.** Few-shot semantic segmentation performs segmentation of novel classes for images [12, 34, 33, 22] or point clouds [35, 8, 14] by only learning from a few support samples. The methods for few-shot image semantic segmentation can be categorized into metric-based [27, 12] and relation-based [34, 33] methods. Metric-based methods aggregate prototypes from the support set as the classifier, and perform cosine similarity with the query features. Relation-based methods concatenate the support features with query features for dense feature comparison via deep convolutional network. Few-shot point cloud segmentation [35] also adopts metric-based technique by performing label propagation among the query points and the multi-prototypes of each class to infer the query point label.

Generalized few-shot image semantic segmentation (GFS-2DSeg) [25, 2] aims to segment both base and novel classes in the query images during testing stage, which is more practical than the few-shot setting. CAPL [25] leverages the contextual cues from the support set and query images to enhance the classifier weights of the base classes. PIFS [2] finetunes on the support set of novel classes during testing stage and proposes prototype-based distillation loss to combat catastrophic forgetting of base classes. In this paper, we study generalized few-shot point cloud semantic segmentation (GFS-3DSeg), a practical yet unexplored task. Different from GFS-2DSeg which assumes the annotation of base classes is available in the novel training samples, we strictly follow the definition of GFSL that only the annotation of novel classes are provided in the support set. Consequently, both CAPL and PIFS fail to work properly under the GFS-3DSeg since both rely on the co-occurrence of base classes in the support sets of novel classes to calibrate the imbalanced learning between base and novel classes. To solve this challenging problem, we propose to mine the representation for the basic geometric structures as the transferable knowledge to improve the representation and classifier of the novel classes.

**Geometric Primitives** Geometric primitives are the fundamental components for the 3D objects, and has been initially studied in the transfer learning in 3D [3, 36]. Chowdhury et al. [3] use microshapes as the basic geometric components to describe any 3D objects in the 3D object recognition task. However, object-level annotation is not available for the query point cloud in the point cloud semantic segmentation or object detection. Therefore, Zhao et al. [36] only utilize the geometric information at point-level by enhancing the local geometric representation of each query point with a geometric memory bank. Although we also performs point-level enhancement in our geometric-aware semantic representation, different from Zhao et al. [36], we inject geometric information to the high-level semantic representation to help model understand the geometric structures through learning the class semantics. Moreover, we propose to use the geometric prototype as the global geometric enhancement to each class to calibrate the biased prediction during testing stage.

### 3. Problem Formulation

In generalized few-shot point cloud semantic segmentation, a base class training dataset $D_{train}^b$, and a novel class training dataset $D_{train}^n$ with non-overlapping label space $C^b \cap C^n = \emptyset$ are provided. The testing dataset $D_{test}$ has a label space $C_{test} = C^b \cup C^n$. During the training stage, the model learns the base classes $C^b$ from abundance of labeled point cloud data $D_{train}^b = \{(P_k^b, M_k^b)_{k=1}^{D_{train}^b}\}$, where $|D_{train}^b|$ denotes the size of $D_{train}^b$. Each point cloud $P_k^b \in \mathbb{R}^{m \times d_0}$ contains $m$ points with feature dimension $d_0$ and $M_k^b$ denotes the annotation of $C^b$ in $P_k^b$.

During the testing stage, the model first learns $C^n$ from limited labeled data $D_{train}^l = \{(P_k^{n,i}, M_k^{n,i})_{K} \}_{i=1}^{[C^n]}$, with $K$ support point clouds per novel class $C^{n,i} \in C^n$, and $[C^n]$ is the number of novel classes. $P_k^{n,i} \in \mathbb{R}^{m \times d_0}$ and $M_k^{n,i}$ is the binary mask indicating the presence of $C^{n,i}$. Note that during testing stage, the model does not have access to $D_{train}^n$. The testing dataset $D_{test} = \{(P_q^k, M_q^k)_{k=1}^{D_{test}}\}$, with each query point cloud $P_q^k \in \mathbb{R}^{m \times d_0}$. $M_q^k$ represents the ground-truth annotation of $C_{test}$ in $P_q^k$. The goal of GFS-3DSeg is to correctly segment both $C^b$ and $C^n$ in the testing query point cloud.

### 4. Our Method

**Background.** We adopt prototype learning to segment both base and novel classes during testing stage. Class prototypes are learned as the classifier weight for each class.
Base class prototypes are learned in the training stage via gradient descent, and novel class prototypes are learned by aggregating the foreground features of the support set during the testing stage. We name the prototype “semantic prototype” since it captures the semantic information of each class. The prediction of each query point is assigned by the class label of the most similar semantic prototypes. However, naively adopting prototype learning is insufficient to learn well on new classes due to the small support set. We thus propose geometric-aware semantic representation and geometric-guided classifier re-weighting to help segmenting new classes.

**Framework Overview.** Fig. 2 shows the overview of our framework, which consists of four main parts: a) The geometric words (GWs) to enhance the representation and classifier of the new classes. b) The geometric-aware semantic representation (GSR) based on the GWs to learn a transferable representation during the base class training stage. c) We first get the geometric feature of a point as the assignment of the GWs to the point. We then fuse the geometric feature with its corresponding semantic feature to get the final GSR. c) The geometric prototype (GP) to supplement the semantic prototype learned from the insufficient training samples. Specifically, a GP is the frequency histogram of the GWs assigned to the points in each class that can uniquely describe the class from the global geometric perspective. d) The geometric-guided classifier re-weighting (GCR) based on GPs to provide prior knowledge of each query point belonging to the potential target classes based on the geometric similarity.

**4.1. Geometric Words**

Unlike its 2D images counterpart, 3D point clouds contain complete geometric information with shared basic geometric components. Understanding these basic geometric components helps learning across old and new classes due to the shared similar local geometric structures. We thus propose geometric words as the representation of these basic geometric components, and utilize it during training and testing stage to facilitate learning of new classes from few shots training point clouds.

To obtain the geometric words, we pretrain the feature extractor \( E \) of attMPTI [35] on \( D_{\text{train}}^{b} \) and collect the features \( \{f_{\text{low}}\} \) of all the points belonging to \( C^{b} \). We concatenate the output feature of the first three EdgeConv layers and denote it as \( f_{\text{low}} \in \mathbb{R}^{d_{l}} \) since lower level features contain more geometric cues. We then obtain \( H \) geometric words \( G = \{g_{h}\}_{h=1}^{H} \in \mathbb{R}^{H \times d_{l}} \) by applying K-means on \( \{f_{\text{low}}\} \) to calculate the \( H \) centroids. Each \( g_{h} \) is a local aggregation of the points with similar \( f_{\text{low}} \), i.e., similar geometric characteristic.

Fig. 1 visualizes the point-to-GWs assignments by searching points with \( f_{\text{low}} \) that are most similar to a given geometric word \( g_{h} \). As shown in Fig. 1, the horizontal plane of the chair, tables and sofas are all activated to the same geometric word. It suggests that our GWs are able to represent shared geometric components among different classes.

**Geometric-aware Semantic Representation.** Based on the GWs, we propose the geometric-aware semantic representation (GSR) to enhance the feature representation of the new classes. The GSR is a fusion of a class-agnostic
Figure 3. Motivation for geometric-guided classifier re-weighting. For each histogram, the horizontal axis represents the index of the GWs and the vertical axis represents the normalized frequency ratio within the class or point. (a) Visualize the geometric feature $\hat{f}_{\text{geo}}$ of the yellow query point on the window frame. (b), (c) and (d) shows the geometric prototypes of window, door and beam, respectively. The red bar denotes the GW with same index.

Geometric feature $f_{\text{geo}} \in \mathbb{R}^H$ and a class-specific semantic feature $f_{\text{sem}} \in \mathbb{R}^{d_2}$. Specifically, the geometric feature $f_{\text{geo}}$ that represents the geometric information of each point is computed by the soft-assignment of its feature $f_{\text{low}}$ to its most similar GW as follow:

$$f_{\text{geo}} = \text{Softmax} \left( \left[ f_{\text{low}} \cdot g_1, \ldots, f_{\text{low}} \cdot g_H; \tau \right] \right),$$  \hspace{1cm} (1)

where $\cdot$ denotes the cosine similarity between the feature and a geometric word. $\tau$ is the temperature to sharpen the probability vector, and we empirically set it to 10. We adopt soft assignment to make it differentiable with respect to the feature extractor. We use the final output of $E$ as the semantic feature $f_{\text{sem}}$. Subsequently, $f_{\text{geo}}$ and $f_{\text{sem}}$ are concatenated and sent into a small convolution block $E_{\text{fuse}}$ to obtain the final representation $f_{\text{fin}} \in \mathbb{R}^{d_3}$ for each point as follow:

$$f_{\text{fin}} = E_{\text{fuse}} (f_{\text{geo}} \parallel f_{\text{sem}}),$$  \hspace{1cm} (2)

where $\parallel$ represents the concatenation of two vectors.

During base class training, we simulate query and fake novel class support set in each batch following [25] to enhance the model’s adaptability to unseen environments. The optimization objective is to minimize cross-entropy loss computed by the prototypes generated through the assembling of $\{p^c_{\text{sem}} | c \in C^{b}\}$ and the fake novel prototypes from the simulated support set. We refer readers to [25] for a more comprehensive understanding of the training strategy.

4.2. Geometric prototype

Although GWs are class-agnostic, their combinations are able to represent different classes in a geometric way. We visualize the frequency of GWs assigned to the points in different classes in Fig. 3(b), (c) and (d). The horizontal axis represents the index of the GWs and the vertical axis represents the normalized frequency ratio. The histogram conveys the global structure of each class via the frequency ratios of the GWs, and different classes have different histograms. The histogram thus uniquely represents its corresponding class and we refer to it as geometric prototype $p^c_{\text{geo}} \in \mathbb{R}^H$:

$$p^c_{\text{geo}} = \frac{\sum_{i=1}^{N^c} \left[ \hat{f}_{\text{geo}} \right]_{c,i}}{N^c},$$  \hspace{1cm} (3)

where $N^c$ denotes the number of points belonging to class $c$ in the training dataset $D_{\text{train}}^b$ or $D_{\text{train}}^a$. $\hat{f}_{\text{geo}} \in \mathbb{R}^H$ is the hard assignment in the form of one-hot vector. We augment the semantic prototype $p^c_{\text{sem}}$ with the geometric prototype $p^c_{\text{geo}}$, as the semantic prototype primarily encodes semantic information and, as a result, becomes insufficient in representing the new classes due to limited training samples.

**Geometric-guided Classifier Re-weighting.** Based on the GP, we propose the geometric-guided classifier re-weighting module to help the prediction of the novel classes. As shown in Fig. 3, the corresponding geometric word for a point on the window frame is activated in the GP of window and the geometrically similar class door, but suppressed in beam which does not have the frame structure. This implies that comparing the geometric feature of the query point with GP can be employed as a hint for segmentation. Therefore, we compute a geometric matching score $s^c$ based on the cosine similarity between a query point’s geometric feature and GP as:

$$s^c = 1 \left[ p^c_{\text{geo}} \cdot \hat{f}_{\text{geo}} \right] = \begin{cases} 1 & p^c_{\text{geo}} \cdot \hat{f}_{\text{geo}} > 0 \\ 0 & \text{otherwise} \end{cases},$$  \hspace{1cm} (4)

where $1[\cdot]$ is an indicator function, and $s^c = 1$ indicates the query point has the same geometric structure as class $c$. $c$ is the class name.

However, the geometric matching may be negatively influenced by the noisy GWs in $p^c_{\text{geo}}$ due to the scene context. As shown in Fig. 4, although points on the wall (in red) are on the vertical plane, they are still activated to the same GW as the points on the horizontal table plane due to adjacency. To suppress these noisy GWs in $p^c_{\text{geo}}$ and improve the accuracy of geometric matching, we propose minor frequency pruning as shown in Algorithm 1. Our motivation is that the GWs representing the typical geometric structure of the class usually contributes a large frequency on the
histogram, while the GWs that are introduced by the scene context have relatively low frequencies. Consequently, we remove GWs corresponding to lower frequencies to only preserve the representative geometric structures. We denote the pruned geometric prototype as $p^{c}_{\text{geo}}$. The frequency limit $\alpha$ in Algorithm 1 denotes the amount of the frequencies to keep in the original $p^c_{\text{geo}}$, and the $p^{c,j}_{\text{geo}}$ denotes the $j$-th entry of $p^{c}_{\text{geo}}$. The computation of the matching score is then updated as:

$$ s^c = 1 \left[ p^{c}_{\text{geo}} \cdot \hat{f}_{\text{geo}} \right], $$

To highlight the geometrically matched class in the prediction, we set a weight $w^c$ to the prediction of each class according to the matching score as follow:

$$ w^c := \begin{cases} \beta & s^c = 1 \\ 1 & s^c = 0 \end{cases}. $$

We set $\beta > 1$ to highlight the potential target classes of the query point. We then re-weight the semantic classification logit $l^c_{\text{sem}}$ with $w^c$ to compute the final prediction logit $l^c_{\text{fin}}$ as follows:

$$ l^c_{\text{fin}} = w^c \times l^c_{\text{sem}}, \quad l^c_{\text{sem}} = p^{c}_{\text{geo}} \cdot f_{\text{fin}}. $$

The $l^c_{\text{fin}}$ considers both semantic and geometric similarity when segmenting new classes, which is more reliable than using semantic prediction logit $l^c_{\text{sem}}$ alone as shown in Figure 4. Illustration of minor frequency pruning. The top left figure shows the green points on the table are activated to the GW representing the horizontal plane, while the red points on the vertical plane of the wall are wrongly activated to the same GW due to scene context. The right figure shows the proposed minor frequency pruning to suppress the activation to the wrong GWs introduced by scene context.

**Algorithm 1 Minor Frequency Pruning**

**Input:** Geometric prototype $p^c_{\text{geo}}$, frequency limit $\alpha \in [0, 1]$

**Output:** Pruned geometric prototype $p^{c}_{\text{geo}}$

1: $\gamma \leftarrow 0$ \> records the accumulated frequencies and is initialized to 0
2: $p^{c}_{\text{geo}} \leftarrow [0, \ldots, 0]^H$ \> Initialize all entries of $p^c_{\text{geo}}$ to 0
3: $\{\text{idx}_1, \ldots, \text{idx}_H\} \leftarrow \text{Sort.Descending} \left\{ p^{c,j}_{\text{geo}} \right\}_{j=1}^{H}$ \> Get indices of $\left\{ p^{c,j}_{\text{geo}} \right\}_{j=1}^{H}$ in descending order
4: $\text{idx}_t \leftarrow \text{idx}_t$
5: while $\gamma < \alpha$ do
6: $\gamma \leftarrow \gamma + p^{c,\text{idx}_t}_{\text{geo}}$
7: $\text{idx}_t \leftarrow \text{idx}_t + 1$
9: end while
10: $p^{c}_{\text{geo}} \leftarrow p^{c}_{\text{geo}}/\text{Sum} \left(p^{c}_{\text{geo}}\right)$ \> Normalization

We evaluate the performance of our model using mean intersection-over-union (mIoU). We use

$$ y = \arg\max \left( \text{Softmax} \left( \left[ l^{l_1}_{\text{fin}}, \ldots, l^{l_{C_{\text{fin}}}}_{\text{fin}} \right]; \tau \right) \right). \quad (8) $$

**5. Experiments**

**5.1. Datasets and Setup**

**Datasets.** We evaluate on two datasets: 1) S3DIS [1] consists 272 point clouds from six areas with annotation corresponding to 13 semantic classes. We use area 6 as the testing dataset $D_{\text{test}}$, and leverage the other five areas to construct the training dataset for base and novel classes. 2) ScanNet [4] consists of 1,513 point clouds with annotation corresponding to 20 semantic classes. We use 1,201 point clouds to construct training dataset of $D_{\text{train}}^b$ and $D_{\text{train}}^n$, and the rest 312 point clouds to construct $D_{\text{test}}$.

For both datasets, we choose the last 6 classes with least labeled points in the corresponding dataset as the novel classes $C^n$ and the rest classes as base classes $C^b$. The motivation is to simulate the scenario in real world, where the frequency of novel class occurring is low and it is hard to collect sufficient training data. Consequently, the novel classes for S3DIS are table, window, column, beam, board and sofa. The novel classes for ScanNet are sink, toilet, bathtub, shower curtain, picture and counter. We follow the data pre-processing of [35] to divide each point cloud into blocks with size of 1 meter $\times$ 1 meter on the $xy$ plane. From each block, we sample $m = 2,048$ points as input. The dimension $d_0$ for the input feature is 9 with XYZ, RGB and normalized XYZ to the block.

**Evaluation Metrics.** We evaluate the performance of model using mean intersection-over-union (mIoU). We use
<table>
<thead>
<tr>
<th>Method</th>
<th>5-shot</th>
<th>1-shot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>mIoU-B</td>
<td>mIoU-N</td>
</tr>
<tr>
<td>Fully Supervised</td>
<td>76.51</td>
<td>58.69</td>
</tr>
<tr>
<td>CAPL [25]</td>
<td>73.56</td>
<td>35.18</td>
</tr>
<tr>
<td>Ours</td>
<td>73.61</td>
<td>43.26</td>
</tr>
</tbody>
</table>

Table 2. Results on S3DIS under 5-shot and 1-shot settings.

<table>
<thead>
<tr>
<th>Method</th>
<th>5-shot</th>
<th>1-shot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>mIoU-B</td>
<td>mIoU-N</td>
</tr>
<tr>
<td>Fully Supervised</td>
<td>43.12</td>
<td>37.04</td>
</tr>
<tr>
<td>PIFS [2]</td>
<td>35.14</td>
<td>2.92</td>
</tr>
<tr>
<td>CAPL [25]</td>
<td>73.56</td>
<td>35.18</td>
</tr>
<tr>
<td>Ours</td>
<td>73.61</td>
<td>43.26</td>
</tr>
</tbody>
</table>

Table 3. Results on ScanNet under 5-shot and 1-shot settings.

mIoU-B, mIoU-N and mIoU-A to denote the mIoU on the base classes, novel classes, and all the classes, respectively. In addition, we use harmonic mean of mIoU-B and mIoU-N to better describe the overall performance on base and novel classes, i.e., $\text{HM} = \frac{2 \times \text{mIoU-B} \times \text{mIoU-N}}{\text{mIoU-B} + \text{mIoU-N}}$. In comparison to mIoU-A, HM is not biased towards the base classes [31].

### 5.2. Implementation details

We adopt the feature extractor of [35] as $E$ and pre-train it on the $D_{\text{train}}$ for 100 epochs. We then perform K-means on the collection of the base class features $\{f_{\text{low}}\}$. We set $H$ as 200 for S3DIS and 180 for ScanNet, respectively. During base class training, we perform geometric-aware semantic representation learning and learn semantic prototypes for base classes $\{p_{\text{sem}}^c | c \in C^b\}$. $d_1$ and $d_2$ are both 192 following attMPTI [35], and $d_2$ is set to 128. We set batch size to 32 and train for 150 epochs. We use Adam optimizer with initial learning rate of 0.01 and decayed by 0.5 every 50 epochs. We load the pre-trained weight of the first three EdgeConv layers and set their learning rate to be 0.001. We compute the geometric prototypes for base classes $\{p_{\text{geo}}^c | c \in C^b\}$ after training completes. During testing stage, we first obtain the semantic prototypes $\{p_{\text{sem}}^c | c \in C^n\}$ and geometric prototypes $\{p_{\text{geo}}^c | c \in C^n\}$ for novel classes by averaging $f_{\text{low}}$ and $f_{\text{geo}}$ (followed by minor frequency pruning) of the foreground points in the support set, respectively. We then predict the class labels for each query point via proposed GCR. $\alpha$ is set to 0.9 and 0.95 for S3DIS and ScanNet, respectively. $\beta$ is set to 1.2.

### 5.3. Baselines

We design three baselines for comparison with our method. 1) **attMPTI** [35] is the state-of-the-art FS-3DSeg method. We follow the original implementation in [35] and episodically train attMPTI on base class dataset. Upon finishing training, we collect multi-prototypes for base classes. During testing stage, we first generate multi-prototypes for novel classes from $D_{\text{train}}^n$. We then estimate the query label by performing label propagation among query points and prototypes of base and novel classes. 2) **PIFS** [2] is the state-of-the-art method for GFS-2DSeg that fine-tunes on $D_{\text{train}}^n$ to learn novel classes. We apply their proposed prototype-based distillation loss to only the scores of novel classes since we do not provide the annotation of base classes in $D_{\text{train}}^n$. 3) **CAPL** [25] is the state-of-the-art method in GFS-2DSeg that performs prototype learning to learn novel classes. We remove the SCE module of CAPL since the annotations of base classes in the $D_{\text{train}}^n$ are not available. All the baselines use the same feature extractor with us for fair comparison.

In addition, we design an oracle setting, **Fully Supervised**, where the model is trained on the fully annotated dataset of base and novel classes using the same feature extractor with us and a small segmentation head.

### 5.4. Comparison with Baselines

Tab. 2 and Tab. 3 show the results of GFS-3DSeg on S3DIS and ScanNet, respectively. We conduct experiments in two settings with the number of support point clouds $K = \{1, 5\}$ on each dataset. We randomly generate 5 sets of $D_{\text{train}}^n$ using different seeds for each setting and calculate the averaged results over all 5 sets to obtain a more reliable results. It is clear to see that the segmentation accuracy of novel classes increases with more number of shots. Compared with all the baselines, our method is able to utilize the limited number of training samples from $D_{\text{train}}^n$ in a more effective way and achieves much better performance in terms
attMPTI [35] fails to perform well on the GFS-3DSeg since it only focuses on establishing the decision boundary for those classes appearing in each episode. When including all the classes in the evaluation, the original decision boundary collapses. PIFS [2] also performs poorly on the GFS-3DSeg. The large intra-class variances of the 3D objects make novel class adaptation difficult for the 2D fine-tuning method. Moreover, their fine-tuning method leads to severe catastrophic forgetting of the base classes due to the absence of base class training data. Our method is based on CAPL [25]. Compared to CAPL which mainly utilizes the context information to enhance the semantic prototypes of the base classes, we focus on enhancing the learning of the new classes by leveraging transferable knowledge GWs. As a result, our method shows great superiority over CAPL in the performance of the novel classes. This can also be verified in Fig. 5, where our model is able to segment target new classes (board and table in the first row, and beam in the second row) more precisely.

**Figure 5. Qualitative comparison on 5-shot setting of S3DIS dataset. Novel classes are marked with red rectangle. The target novel classes in the first row are board, table, column and window. The target novel classes in the second row is beam.**

### Table 4. Effectiveness of geometric-aware semantic representation (GSR) and geometric-guided classifier re-weighting (GCR) on S3DIS.

<table>
<thead>
<tr>
<th>GSR</th>
<th>GCR</th>
<th>mIoU-B</th>
<th>mIoU-N</th>
<th>mIoU-A</th>
<th>HM</th>
</tr>
</thead>
<tbody>
<tr>
<td>✗</td>
<td>✗</td>
<td>73.56</td>
<td>35.18</td>
<td>55.85</td>
<td>47.51</td>
</tr>
<tr>
<td>✓</td>
<td>✗</td>
<td>73.48</td>
<td>40.10</td>
<td>58.07</td>
<td>51.81</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>73.61</td>
<td>43.26</td>
<td>59.60</td>
<td>54.42</td>
</tr>
</tbody>
</table>

### Table 5. Ablation study of number of geometric words in S3DIS.

<table>
<thead>
<tr>
<th>Number of GWs</th>
<th>mIoU-B</th>
<th>mIoU-N</th>
<th>mIoU-A</th>
<th>HM</th>
</tr>
</thead>
<tbody>
<tr>
<td>H = 100</td>
<td>72.45</td>
<td>43.61</td>
<td>59.14</td>
<td>54.38</td>
</tr>
<tr>
<td>H = 150</td>
<td>73.88</td>
<td>42.27</td>
<td>59.29</td>
<td>53.71</td>
</tr>
<tr>
<td>H = 200</td>
<td>73.61</td>
<td>43.26</td>
<td>59.60</td>
<td>54.42</td>
</tr>
<tr>
<td>H = 250</td>
<td>72.96</td>
<td>41.29</td>
<td>58.32</td>
<td>52.63</td>
</tr>
<tr>
<td>H = 400</td>
<td>73.19</td>
<td>40.84</td>
<td>58.26</td>
<td>52.36</td>
</tr>
</tbody>
</table>

### Table 6. Ablation Study of frequency limit $\alpha$.

<table>
<thead>
<tr>
<th>Frequency Limit</th>
<th>mIoU-B</th>
<th>mIoU-N</th>
<th>mIoU-A</th>
<th>HM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha = 1$</td>
<td>73.50</td>
<td>41.05</td>
<td>58.50</td>
<td>52.59</td>
</tr>
<tr>
<td>$\alpha = 0.9$</td>
<td>73.48</td>
<td>43.19</td>
<td>59.50</td>
<td>54.34</td>
</tr>
<tr>
<td>$\alpha = 0.85$</td>
<td>73.59</td>
<td>43.01</td>
<td>59.48</td>
<td>54.22</td>
</tr>
</tbody>
</table>

### Table 7. Ablation Study of logits weight $\beta$.

<table>
<thead>
<tr>
<th>logits weight</th>
<th>mIoU-B</th>
<th>mIoU-N</th>
<th>mIoU-A</th>
<th>HM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta = 1$</td>
<td>73.48</td>
<td>40.10</td>
<td>58.07</td>
<td>51.81</td>
</tr>
<tr>
<td>$\beta = 1.2$</td>
<td>73.61</td>
<td>43.26</td>
<td>59.60</td>
<td>54.42</td>
</tr>
<tr>
<td>$\beta = 1.5$</td>
<td>73.20</td>
<td>42.25</td>
<td>58.90</td>
<td>53.51</td>
</tr>
</tbody>
</table>

### Table 8. Comparison using state-of-the-art feature extractor Point Transformer v2 (PTv2) [30].

<table>
<thead>
<tr>
<th>Method</th>
<th>mIoU-B</th>
<th>mIoU-N</th>
<th>mIoU-A</th>
<th>HM</th>
</tr>
</thead>
<tbody>
<tr>
<td>PTv2+CAPL</td>
<td>86.35</td>
<td>27.58</td>
<td>59.22</td>
<td>41.68</td>
</tr>
<tr>
<td>PTv2+Ours</td>
<td>86.38</td>
<td>37.26</td>
<td>63.71</td>
<td>52.01</td>
</tr>
</tbody>
</table>

### 5.5. Ablation Study

#### Effectiveness of GSR and GCR.

We verify the effectiveness of the geometric-aware semantic representation (GSR) and geometric-guided classifier re-weighting (GCR) on 5-shot setting using S3DIS (Tab. 4). The model without GSR uses $f_{sem}$ as the feature representation for each point. The model without GCR adopts $l_{sem}$ as the final prediction logit of class $c$. Both GSR and GCR are beneficial to new class segmentation, which suggests the successful enhancement of the representation and classifier of the novel classes by using transferable geometric information.

#### Analysis of the number of GWs.

Tab. 5 studies the influence of GW numbers on the performance of 5-shot setting using S3DIS. Although the performance varies with the different numbers of geometric words, they all largely outperform the baselines without GWs. We choose $H = 200$ in our model regarding its best overall performance.
Effectiveness of the minor frequency pruning. In Tab. 6, we analyze the effect of the minor frequency pruning in the geometric prototype. GP without pruning ($\alpha = 1$) shows worst performance compared to the pruned GPs. $\alpha = 0.9$ gives the best performance, thus we adopt $\alpha = 0.9$ in our final model.

Analysis of the weight $\beta$. We present the ablation study of the weight $\beta$ in Tab. 7. $\beta = 1$ does not highlight any potential target classes predicted by geometric matching, which is equivalent to the model without GCR. Setting a moderate threshold $> 1$ is helpful to improve the segmentation performance, so we choose $\beta = 1.2$ for model testing.

Using SOTA feature extractor. Tab. 8 shows the result of replacing DGCNN-based feature extractor [35] with Point Transformer v2 (PTv2) under the 5-shot setting of S3DIS. Our method outperforms the strongest baseline CAPL by a large margin on novel classes and overall performance. It verifies that our method can work successfully with a state-of-the-art point cloud feature extractor. We notice that the mIoU-N in Tab. 8 is lower than that of Tab. 2. One possible reason is that the feature extractor in [35] is specially designed to be able to quickly learn new classes from a small support set.

Standard deviation and per-class IoU. Tab. 10 shows the standard deviation results on the 5 testing sets of the 5-shot setting of S3DIS. Our model shows similar variation with CAPL. Tab. 9 shows per-class IoU of the 5-shot setting of S3DIS. Our method largely outperforms CAPL for all new classes while maintaining on-par performance on base classes.

6. Conclusion

In this paper, we present the unexplored yet important generalized few-shot point cloud segmentation. We address the challenge of facilitating new class segmentation with limited training samples by utilizing transferable knowledge geometric words (GWs) mined from the base classes. We propose geometric-aware semantic representation to learn generalizable representation where geometric features described through GWs are fused with semantic representation. We further propose the geometric prototype (GP) to supplement the semantic prototype in the testing stage. Extensive experiments on two benchmark datasets demonstrate the superiority of our method.

Acknowledgement. This research work is fully done at the National University of Singapore and is supported by the Agency for Science, Technology and Research (A*STAR) under its MTC Programmatic Funds (Grant No. M23L7b0021).

References


<table>
<thead>
<tr>
<th>Method</th>
<th>Ceiling</th>
<th>Floor</th>
<th>Wall</th>
<th>Beam</th>
<th>Column</th>
<th>Window</th>
<th>Door</th>
<th>Table</th>
<th>Chair</th>
<th>Sofa</th>
<th>Bookcase</th>
<th>Board</th>
<th>Clutter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPL</td>
<td>93.17</td>
<td>97.36</td>
<td>72.92</td>
<td>52.44</td>
<td>28.82</td>
<td>35.12</td>
<td>77.57</td>
<td>60.60</td>
<td>61.29</td>
<td>10.38</td>
<td>54.42</td>
<td>23.75</td>
<td>58.19</td>
</tr>
<tr>
<td>Ours</td>
<td>92.34</td>
<td>97.16</td>
<td>70.73</td>
<td>60.37</td>
<td>32.16</td>
<td>46.30</td>
<td>76.21</td>
<td>64.41</td>
<td>63.42</td>
<td>16.99</td>
<td>54.77</td>
<td>39.30</td>
<td>60.66</td>
</tr>
</tbody>
</table>

Table 10. Per-class IoU. Red denotes new classes.

<table>
<thead>
<tr>
<th>Method</th>
<th>mIoU-B</th>
<th>mIoU-N</th>
<th>mIoU-A</th>
<th>HM</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPL</td>
<td>0.17</td>
<td>3.25</td>
<td>1.46</td>
<td>2.94</td>
</tr>
<tr>
<td>Ours</td>
<td>0.35</td>
<td>3.34</td>
<td>1.63</td>
<td>2.75</td>
</tr>
</tbody>
</table>

Table 10. Standard deviation.


