Zero-Shot Point Cloud Segmentation by Semantic-Visual Aware Synthesis

Yuwei Yang¹ Munawar Hayat² Zhao Jin¹ Hongyuan Zhu³ Yinjie Lei¹
¹Sichuan University ²Monash University ³A*STAR

Consistent (b) Semantic-Visual Alignment (c) Semantic-Visual Consistency

"chair"

Various Visual Representations:

Semantic Embedding of Seen "chair"

(a) Semantic-Visual Correspondence

(b) Semantic-Visual Alignment

(c) Semantic-Visual Consistency

Figure 1. Our zero-shot synthesis approach for point cloud segmentation tackles multiple semantic-visual transfer issues, by enhancing correspondence (Sec. 3.2), alignment (Sec. 3.3) and consistency (Sec. 3.4) between the auxiliary-semantic and 3D-visual spaces.

Abstract

This paper proposes a feature synthesis approach for zero-shot semantic segmentation of 3D point clouds, enabling generalization to previously unseen categories. Given only the class-level semantic information for unseen objects, we strive to enhance the correspondence, alignment and consistency between the visual and semantic spaces, to synthesise diverse, generic and transferable visual features. We develop a masked learning strategy to promote diversity within the same class visual features and enhance the separation between different classes. We further cast the visual features into a prototypical space to model their distribution for alignment with the corresponding semantic space. Finally, we develop a consistency regularizer to preserve the semantic-visual relationships between the real-seen features and synthetic-unseen features. Our approach shows considerable semantic segmentation gains on ScanNet, S3DIS and SemanticKITTI benchmarks. Our code is available at: https://github.com/leolyj/3DPC-GZSL

1. Introduction

Semantic segmentation of 3D point clouds is mostly dominated by fully-supervised methods [37, 39, 49, 45, 21, 57] that require point-wise labelled data for training. While these methods perform well on previously seen objects, they lack scalability to novel and unseen classes for which no samples are available during training. Zero-Shot Learning (ZSL) provides a promising paradigm in such cases since it enables rapid generalization to unseen classes.

While ZSL from RGB images is well explored [15, 1, 17, 56, 19, 48, 24, 53, 7, 16, 41, 47, 6, 52, 18, 9, 20, 27, 58], ZSL for segmentation of point clouds is less investigated, due to unique challenges posed by 3D data (e.g. the lack of large-
scale annotated datasets and pre-trained models [23] which are otherwise ubiquitous in 2D [34]). Most of the existing 3D ZSL methods tackle the relatively simpler classification problem [13, 10, 11, 12], with very few methods developed for segmentation [8, 30]. Chen et al. [8] learn shared geometric primitives to enable seen-to-unseen migration. However, their approach needs non-annotated unseen samples at training [8], which is restrictive and not suitable for practical scenarios where acquiring unseen data is not always feasible. [30] propose a feature synthesis-based approach for 3D segmentation that can simultaneously generalize to both seen and unseen, without requiring any data for unseen categories. Nevertheless, the features synthesized from the generator lack contextual diversity due to mode collapse [50], resulting in limited transfer to unseen classes.

To enable generalization to wider scenarios, we develop a feature synthesis framework, which doesn’t require any samples (annotated or non-annotated) during training. Since semantics are the only common information available for seen and unseen, we need to ensure strong transfer capabilities from the semantic to the visual space. For this purpose, we consider the following semantic-visual transfer issues for ZSL: 1) Semantic-Visual Correspondence Mismatch. The core of ZSL is to exploit and establish a mapping between semantics and vision, such that for a specific object, visual features can be uniquely identified from their corresponding semantics. 2) Heterogeneous Semantic-Visual Embedding. The semantic vectors (embeddings of class-name words) and visual representations (from point cloud data) come from different modalities, and introduce inherent modality-specific heterogeneity that needs to be tackled in order to align the two data modalities. 3) Inconsistent Semantic-Visual Relationship. The relationships between different classes, both seen and unseen, should be consistent in the semantic embedding space and visual feature space, so that the semantics for unseen can faithfully synthesize the unseen visual features.

To address these semantic-visual transfer challenges, as shown in Fig. 1, we design three modules. First, we propose a Mask Correspondence Learning (MCL) module (Sec. 3.2), to learn rich intra-class representations while enhancing inter-class boundary distribution. We believe promoting diversity between the same class features and ensuring separation between classes is critical to synthesize generalized features for unseen classes. Further, for better seen-to-unseen transfer, we align the seen visual prototypes with their corresponding semantics, using our proposed Heterogeneous Prototype Alignment (HPA) module (Sec. 3.3). Finally, while learning to synthesize the unseen visual features, we ensure that the inter-class structural relations of seen+unseen semantics are consistent with their corresponding visual features. For this purpose, we develop a Relational Transfer Consistency (RTC) module (Sec. 3.4) that transfers the seen+unseen semantic relationships with the corresponding real-seen+synthesized-unseen visual ones. Our proposed modules complement each other and constrain the generator to synthesize diverse, discriminative, and semantically relevant unseen visual features that generalize well for zero-shot segmentation.

We evaluate our model under the challenging Generalized Zero-Shot Learning (GZSL) in inductive setting, where training data contains no labelled or unlabelled unseen class samples, while the model is required to predict both seen and unseen classes at inference. We show significant gains over the current state-of-the-art on three public datasets ScanNet [14], S3DIS [2] and SemanticKITTI [3], by 7.7%, 3.8% and 3.0% respectively, according to the HmIoU metric. Our contributions can be summarized as follows:

- We propose an effective masked learning strategy, where visual features of the masked semantics are recovered via contrastive learning to enhance intra-class diversity and inter-class separation of the learned visual features, enhancing transfer to unseen classes.
- We propose cross-modality prototypical learning that aligns semantics with the visual space, thus promoting generalization to novel concepts.
- We develop consistency regularization that maintains relationships between the real+synthesized visual features with their corresponding semantics.

2. Related Works

By only using auxiliary class attributes or semantics, Zero-Shot Learning (ZSL) enables transfer of prior knowledge from seen to novel unseen classes. Here, we first review ZSL methods developed for RGB images. We then discuss existing 3D semantic segmentation techniques, followed by ZSL methods on 3D point clouds.

**ZSL on RGB Images.** The existing methods for zero-shot learning can be categorized as attribute-based, projection-based, knowledge-based and generative-based methods. The attribute-based methods [15, 26, 22, 1] recognize new objects using the semantic attributes of different classes. The projection-based approaches [17, 43, 51, 56] learn a mapping between visual representations and the auxiliary semantic prototypes (such as Word2Vec embeddings [31] or GloVe [36]). Knowledge-based models [19, 48, 24] employ graph networks to migrate structured knowledge from seen classes to unseen. Recently popular generative approaches [53, 7, 16, 41, 47] train generative models (e.g. conditional generative adversarial models [32] or variational autoencoder [46]), and then synthesize unseen latent features conditioned upon the corresponding class prototypes. The synthesized features are applied to update the classifier to include unseen classes, which helps reduce the bias towards
seen classes. The above mentioned ZSL methods are primarily developed for image classification. Some recent techniques [6, 52, 18, 9, 20, 27] extend them to ZSL for semantic segmentation in RGB images. Amongst these, generative approaches have shown most promise for RGB semantic segmentation in zero-shot setting [28, 18, 9].

3D Point Cloud Semantic Segmentation. Most of the existing methods on 3D segmentation are fully-supervised [44, 29, 38, 37, 39, 49, 45, 4, 21, 57], and project point cloud into multi-view 2D images [44, 38] or process them using voxel grids [29]. Since the seminal work PointNet [37], point clouds are encoded by using deep networks with MLPs [39], point-wise convolution [49, 45, 4], graph networks [21] or transformer [57]. While these deep models show impressive results in fully-supervised setting [35, 54], they require expensive point-wise annotations, and lack generalization to unseen classes in zero-shot setting.

ZSL on 3D Point Clouds. Compared with ZSL from RGB images, 3D ZSL is relatively less investigated. [13] adapts 2D ZSL to 3D, by learning a projection between the PointNet [37] features and the auxiliary semantics. Their work is further extended in [10, 12] to tackle the hubness problem [40], and in [11] assuming non-annotated unseen samples are available. For zero-shot segmentation, [8] learns shared geometric primitives between the seen and unseen classes by assuming that the samples of the unseen classes are available at training. Since their approach requires access to unlabelled unseen class data, i.e., transductive setting, it limits their applicability to real-life scenarios where acquiring training samples for rare categories is not feasible. The closest to our approach is [30], where no training samples for unseen classes are used. While [30] synthesizes the features for unseen, they do not fully exploit the semantic-visual relationships, resulting in coarse visual features that lack effective transfer.

We can therefore conclude that while some progress has been made towards 3D zero-shot classification, 3D zero-shot semantic segmentation with no unseen training samples remains an open research problem. This work makes a progress towards this direction by learning diverse and discriminative visual features, that are well-aligned with the corresponding semantic space, thus enabling the synthesized features to generalize well to unseen classes.

3. Methodology

3.1. Problem Definition

Let’s define a set of object categories as $C$, with the seen $C^S$ and the unseen $C^U$ classes. Let $\mathcal{D}$ denote the dataset with the point cloud set $\mathcal{P}$, the corresponding label set $\mathcal{Y}$ and class prototypes set $\mathcal{T}$, where $\mathcal{T}$ contains the auxiliary $D$-dimensional semantic embedding vectors (e.g. given by Word2Vec [31] or GloVe [36]). Since we follow the challenging inductive Generalized ZSL setting instead of vanilla ZSL, we train the model using samples containing only $C^S$ categories, and test on the scenes containing point cloud with classes both in $C^S$ and $C^U$. Thus, the training set $\mathcal{D}_{\text{train}}$ and test set $\mathcal{D}_{\text{test}}$ can be denoted as $\mathcal{D}_{\text{train}} = \{(p, y, t) \mid \forall i, y_i \in C^S\}$ and $\mathcal{D}_{\text{test}} = \{(p, y, t) \mid \forall i, y_i \in C^S \cup C^U\}$, where $p \in \mathcal{P}$ has $N$ points, $y \in \mathcal{Y}, t \in \mathcal{T}$, and $y_i$ is the ground-truth label for point $i$.

For our approach, we define the generator as $G(\cdot)$, the feature embedding network as $\theta(\cdot)$, and the segmentor as $f(\cdot)$. As illustrated in Fig. 2, the overall training pipeline can be summarized as follow: a) Train a feature embedding network $\theta$ and a seen-class segmentor $f_{\text{seen}}$ using only the seen class data; b) Train a generator $G(\cdot)$ on seen data using the auxiliary semantic vectors, so that the synthetic visual features generated by $G$ are as similar as possible to the real point features extracted by frozen $\theta$; c) Combine the synthetic unseen features of classes $C^U$ generated by $G$ together with the real extracted features on seen classes $C^S$ to train the final segmentor $f_{\text{final}}$. The ultimate goal is that the resulting composite network $\theta$ with $f_{\text{final}}$ can effectively segment point clouds for both seen and unseen classes. The challenge however is weak transfer from semantic to visual space. To promote generalization to unseen categories, we improve semantic-visual transfer by enhancing correspondence (Sec. 3.2), alignment (Sec. 3.3) and consistency (Sec. 3.4) between semantic and visual spaces, aiming to assist generator training and improve the synthesized features quality.

3.2. Mask Correspondence Learning

Given the auxiliary semantic embeddings $t^s_c$ and random noise $z^s_c$ of seen class $c$ as input into $G(\cdot)$, we synthesize features $F^s_c$, such that they closely follow the distribution of the real features $F^s_c = \theta(p^s_c)$ extracted from model $\theta$. Unlike 2D counterparts, we lack large-scale 3D pre-trained backbones to train a generator that can synthesize diverse point-wise features. To promote transfer to unseen classes, we propose to establish a strong correspondence between the input semantic and output visual spaces. Such a correspondence should enhance intra-class richness and ensure features belonging to different classes are well separated. Therefore, while training generator $G(\cdot)$ on the seen classes, we ensure that the generated features have within-class diversity, and clear decision boundaries exist between different classes. Further, the class-wise features should be unique and follow the corresponding semantics. With these objectives in mind, we develop a masking strategy that learns by recovering the masked context.

As shown in Fig. 2, for the point cloud $p^s_c$ containing seen class $c \in C^S$, we randomly mask out part of the corresponding input auxiliary semantic embeddings $t^s_c$, then the
generated features by $G(\cdot)$ can be represented as:

$$\hat{F}^c_s = G(H(q)t^c_s \oplus z^c_s), \ c \in C^S$$  \hspace{1cm} (1)

where $\oplus$ indicates the concatenation operation, $H(\cdot)$ is initialized to 1 and masked with 0 with probability $q$. The size of $t^c_s$ and $z^c_s$ is set to match the number of points in class $c$ of the current scene. During training, the generator $G(\cdot)$ recovers the visual features conditioned on the randomly masked semantics, which helps it learn the intra-class diversity. Semantic-conditioned visual synthesis is essentially one-to-many mapping, and masking the semantics introduces diversity in the semantic space, and thus promotes diversity and richness in the corresponding visual space. Besides, to promote discrimination between visual features of different classes, we consider the real feature $F_s$ extracted by frozen $\theta$ of seen class $c$ as the positive samples $F^c_s$, and the features of other seen classes $h$ in current $p_s$ as the negative samples $F^h_s$, and apply InfoNCE [33] loss:

$$\mathcal{L}_{con} = -\log \frac{\exp(\hat{F}^c_s \cdot F^c_s/\tau)}{\sum_{k \in C^S, k \neq c} \exp(\hat{F}^c_s \cdot F^k_s/\tau) + \exp(\hat{F}^c_s \cdot F^c_s/\tau)},$$  \hspace{1cm} (2)

where $F^c_s = \theta(p^c_s), F^k_s = \theta(p^k_s), p_s = p^c_s \cup p^k_s, p_s$ represents the seen class point clouds and $\tau$ is the temperature parameter. Contrastive learning enhances the discrimination between different categories. Our proposed semantics masking and visual contrast learning strategies therefore ensure that the learned visual space is rich and discriminative.

### 3.3. Heterogeneous Prototype Alignment

The semantic embeddings and visual features are from different modalities, and directly using the semantics for visual synthesis, without any alignment, is sub-optimal. We therefore propose to align the cross-modality heterogeneous features before synthesis. Inspired by the prototypical learning [42], we cast the original features into the prototypical space to model their distribution for alignment. Since the semantic embedding vectors $t^c_s$ corresponding to a seen class $c$ can naturally be regarded as a prototype, we only need visual prototypes on the seen features $F^c_s$.

To generate visual prototypes, instead of the simple average for point cloud visual features, we develop a neighbor-aware approach that reflects the intra-class fine-grained local structure. Specifically, we adopt the Farthest Point Sampling (FPS) algorithm to sample $r$-proportion $(0 < r < 1)$ point features $\{F^{c,a}_{s,a}\}_{a=1}^{[n*r]}$ as anchors on the real seen features $\{F^{c,i}_{s,i}\}_{i=1}^{n}$ embedded by $\theta$, $n \leq N$ is the number of points for class $c$, $\lfloor \cdot \rfloor$ denotes the rounding operation. We calculate the $\ell_2$ distance between $n$ point features and $\lfloor n*r \rfloor$ anchors and assign the nearest anchor index to each point. We average the point features of the same anchor index to form $\lfloor n*r \rfloor$ $(\geq 1)$ visual prototypes $\{H^{c,b}_{s,b}\}_{b=1}^{[n*r]}$ as:

$$H^{c,b}_{s,b} = \frac{1}{\xi^{c,b}_s} \sum_{F^{c,i}_{s,i} \in \xi^{c,b}_s} F^{c,i}_{s,i},$$  \hspace{1cm} (3)

where $\xi^{c,b}_s$ is the partition region composed of the point features assigned to anchor $b$. After getting the semantic and
visual prototypes, we align them to enhance visual synthesis quality. We apply the linear $\sigma(\cdot)$ function to map the semantic embedding $t^c_v$ to the same dimension as the visual prototypes $H^{c,b}_s$, and minimize cosine distance $d(\cdot, \cdot)$:

$$L_{align} = \frac{1}{n \times r} \sum_{b=1}^{|n \times r|} d(H^{c,b}_s, \sigma(t^c_v)),$$  

(4)

where $\sigma: \mathbb{R}^{D_1} \rightarrow \mathbb{R}^{D_2}$, $D_1$ and $D_2$ are the feature dimensions of semantic embeddings and visual prototypes respectively. We further add the aligned semantic vector $\sigma(t^c_u)$ with the synthesized feature $\hat{F}^{c}_t$ to enhance representations $\hat{F}^{c}_u + \sigma(t^c_u)$ for generator $G(\cdot)$ training. Besides, alignment on seen data helps to obtain a well-learned semantic-visual mapping which helps better synthesizes of unseen features $\hat{F}^{c}_u + \sigma(t^c_u), c' \in C_U$ for $f_{final}$ segmentor.

### 3.4. Relational Transfer Consistency

Since the model is only optimized on the seen class data, and never encounters unseen data (as it is not available), the model becomes biased and confuses unseen classes as seen. To counter this, inspired by [27], we propose semantic-visual consistency regularization. We argue that even though the seen and unseen might have different semantic and visual structures, the inter-class relationships in their respective spaces should be preserved. Specifically, we employ the generator $G$ to synthesize visual features for a specific unseen class $c'$, denoted as $\hat{F}^{c'}_u$ and its corresponding semantic prototype is $t^c_u$. Similarly, for a seen class $c$, its synthetic visual features and semantic prototype can be represented as $\hat{F}^{c}_u$ and $t^c_u$ respectively. We construct unseen $c'$ visual synthetic prototype $\hat{H}^{c'}_u$ and seen $c$ visual synthetic prototype $\hat{H}^c_u$ as:

$$\hat{H}^{c'}_u, \hat{H}^c_u = \frac{1}{|n'|} \sum_{i=1}^{n'} \hat{F}^{c',i}_u, \frac{1}{|n|} \sum_{i=1}^{n} \hat{F}^{c,i}_u,$$  

(5)

where $n'$ and $n$ denote the number of points belonging to unseen $c'$ and seen classes $c$, respectively, we apply simple averaging to obtain the visual prototype, since the generated features lack fine-grained structure relative to the real features. We build sets $\{t^{c'}_u\}_{c' \in C_U}$ and $\{\hat{H}^{c'}_u\}_{c' \in C_U}$ for semantic and visual prototypes in unseen classes. We further get the distance distribution relation matrices for semantic $\mathcal{W} \in \mathbb{R}^{m \times m}$ and visual $\mathcal{V} \in \mathbb{R}^{m \times m}$ between the prototypes of seen and unseen sets respectively,

$$\mathcal{W}_{ej} = ||t^e - t^j||^2, \mathcal{V}_{ej} = ||\hat{H}^e - \hat{H}^j||^2,$$  

(6)

where $m$ is the total number of elements in set $t^m = \{t^e\}_{c' \in C_U}$ or $\hat{H}^m = \hat{H}^e \cup \{\hat{H}^e\}_{c' \in C_U}$. $e \leq m$ and $j \leq m$ denote the index of an element in the set $t^m$ and $\hat{H}^m$.

We strive to keep the distance distribution in the two spaces consistent by minimizing the cosine distance $d(\cdot, \cdot)$ as:

$$L_{cst} = \sum_{e=1}^{m} d(\mathcal{W}_{ej}, \mathcal{V}_{ej}).$$  

(7)

Thus, we establish a consistency bridge between the visual and semantic space of seen and unseen classes, so the model can effectively tackle the bias towards the seen.

### 3.5. Network Training and Inference

For the backbone $\theta(\cdot)$ and $f_{seen}(\cdot)$ training, we apply the cross-entropy loss between network output and labels $y_c$ on only seen point clouds $p^c, c \in C^S$,

$$L_{f_{seen}} = - \sum_{c} y_c \log(f_{seen}(\theta(p^c))).$$  

(8)

To train the generator $G$, we apply the Maximum Mean Discrepancy (MMD) loss [28] to narrow the distribution mismatch between the synthesized $\hat{F}^{c}_u + \sigma(t^c_u)$ as $\hat{F}^{c}_s,t$ and the real features $F^{c}_s$ on seen $c$, and combine $L_{cons}, L_{align}, L_{cst}$ losses to form the joint loss:

$$L_{MMD} = \sum_{x,x' \in F^{c'}_s} \mu(x, x') + \sum_{\hat{x}, \hat{x}' \in \hat{F}^{c'}_{s,t}} \mu(\hat{x}, \hat{x}') - 2 \sum_{x \in F^{c'}_s} \sum_{\hat{x} \in \hat{F}^{c'}_{s,t}} \mu(x, \hat{x}),$$  

(9)

$$L_{G} = \sum_{c} (L_{MMD} + L_{cons} + L_{align} + \alpha L_{cst}),$$  

(10)

where $\mu(\cdot, \cdot)$ is the Gaussian kernel function, $\mu(x, x') = \exp(-\frac{1}{2}||x - x'||^2), \alpha$ is a hyper-parameter for loss balance. It should be noted that we do not use discriminator to make the features more realistic, which is demonstrated in [30] that it may be harmful for 3D point clouds. The well-trained generator $G$ synthesizes unseen features $\hat{F}^{c}_u + \sigma(t^c_u)$ as $\hat{F}^{c}_s,t$ $c' \in C^U$ which will combine with the real seen features $F^{c}_s$ on $c$ to train the final segmentor $f_{final}$ using,

$$L_{f_{final}} = - \sum_{c} y_{c'} \log(f_{final}((\hat{F}^{c'}_u))) - \sum_{c'} \hat{y}_{c'} \log(f_{final}((\hat{F}^{c'}_u))),$$  

(11)

where $\hat{y}_{c'}$ denotes the synthetic unseen labels. At inference time, we combine the $\theta$ and $f_{final}$ to jointly predict both seen $C^S$ and unseen $C^U$ categories.

### 4. Experiments

#### 4.1. Datasets and Settings

**Datasets.** We follow [30] to conduct experiments based on three public 3D semantic segmentation datasets ScanNet
Table 1. Generalized 3D zero-shot semantic segmentation results on three benchmarks. All methods use GloVe+Word2Vec embeddings. The evaluation metric are mIoU and HmIoU (%). $C^U$ stands for pseudo generated unseen data. The results of all comparison methods are derived from [30]. Our approach shows impressive gains of 7.7%, 3.8%, 3.0% based on HmIoU in ScanNet, S3DIS and SemanticKITTI datasets respectively.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Full supervision</th>
<th>ZSL backbone</th>
<th>ZSL-trivial</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C$^S$ $U$</td>
<td>C$^S$ $U$</td>
<td>C$^S$ $U$</td>
</tr>
<tr>
<td>ScanNet</td>
<td>43.3 51.9 45.1</td>
<td>41.5 39.2 40.3</td>
<td>39.2 0.0 31.3</td>
</tr>
<tr>
<td>S3DIS</td>
<td>47.2 74.0 50.0 66.6</td>
<td>40.3 60.9 21.5 48.7</td>
<td>0.0 70.2 0.0 48.6</td>
</tr>
<tr>
<td>SemanticKITTI</td>
<td>59.6 59.4 57.5</td>
<td>31.8 25.9 42.3</td>
<td>0.0 55.8 0.0 44.0</td>
</tr>
</tbody>
</table>

**Generalized zero-shot-learning methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>segmentor</th>
<th>mIoU</th>
<th>HmIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeviSe-3DSeg* [17]</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>20.0 0.0 16.0</td>
<td>0.0 70.2 0.0 48.6</td>
</tr>
<tr>
<td>ZSLPC-Seg* [13]</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>16.4 4.2 13.9</td>
<td>6.7 5.2 1.3 4.0</td>
</tr>
<tr>
<td>DeviSe-3DSeg [17]</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>12.8 3.0 10.9</td>
<td>4.8 3.6 1.4 3.0</td>
</tr>
<tr>
<td>3DGenZ [30]</td>
<td>C$^S$ $U$</td>
<td>32.8 7.7 27.8</td>
<td>12.5 53.1 7.3 39.0</td>
<td></td>
</tr>
</tbody>
</table>

**Ours** | C$^S$ $U$ | 34.5 | 44.3 | 30.4 | 20.2 |

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>segmentor</th>
<th>mIoU</th>
<th>HmIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeViSe-3DSeg*</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>28.2 0.0 22.6</td>
<td>0.0 65.5 0.0 45.3</td>
</tr>
<tr>
<td>ZSLPC-Seg*</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>16.4 4.2 13.9</td>
<td>6.7 5.2 1.3 4.0</td>
</tr>
<tr>
<td>DeviSe-3DSeg</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>12.8 3.0 10.9</td>
<td>4.8 3.6 1.4 3.0</td>
</tr>
<tr>
<td>3DGenZ [30]</td>
<td>C$^S$ $U$</td>
<td>32.8 7.7 27.8</td>
<td>12.5 53.1 7.3 39.0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>segmentor</th>
<th>mIoU</th>
<th>HmIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>C$^S$ $U$</td>
<td>34.5 44.3 30.4</td>
<td>20.2 58.9 9.7 43.8</td>
<td></td>
</tr>
<tr>
<td>DeViSe-3DSeg*</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>28.2 0.0 22.6</td>
<td>0.0 65.5 0.0 45.3</td>
</tr>
<tr>
<td>ZSLPC-Seg*</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>16.4 4.2 13.9</td>
<td>6.7 5.2 1.3 4.0</td>
</tr>
<tr>
<td>DeviSe-3DSeg</td>
<td>C$^S$</td>
<td>C$^S$</td>
<td>12.8 3.0 10.9</td>
<td>4.8 3.6 1.4 3.0</td>
</tr>
<tr>
<td>3DGenZ [30]</td>
<td>C$^S$ $U$</td>
<td>32.8 7.7 27.8</td>
<td>12.5 53.1 7.3 39.0</td>
<td></td>
</tr>
</tbody>
</table>

Table 1 compares different approaches in terms of mIoU. We observe that our method achieves consistently superior performance on the three datasets. We outperform the current state-of-the-art method 3DGenZ [30] by a large margin of 7.7%, 3.8% and 3.0% of HmIoU metric on ScanNet, S3DIS and SemanticKITTI, respectively. The results suggest that our modules can effectively migrate to unseen knowledge and generalize to novel categories. Moreover, it should be noted that our approach also retains performance on seen classes. We believe that our generator synthesizes realistic features that are distinguishable between different classes, while the seen and unseen...
Figure 3. Qualitative comparison with 3DGenZ [30] under inductive generalized zero-shot setting. The results in black on the ScanNet and SemanticKITTI datasets represent unlabeled data. The regions in red boxes highlight the effectiveness of our method.

visual features are well aligned with the corresponding semantic space, thus benefiting the training of the final segmentor. Compared with the performance on SemanticKITTI, our method shows a higher improvement than other methods on the ScanNet and S3DIS datasets. The reason for this might be that the categories in the large outdoor scenes involved in SemanticKITTI are more complex, making it challenging to generalize to novel unseen classes. We further report the IoU of the individual seen and unseen categories for various datasets in the supplementary material.

**More comparisons with adapted 2D methods:** In Tab. 2, we adapt five 2D generalized zero-shot semantic segmentation methods [52, 6, 27, 18, 55] to 3D point cloud. We evaluate these methods in our inductive setting using the same 3D backbone (FKACConv [5]) on ScanNet dataset. Results suggest that existing classical 2D methods are not directly suitable for 3D point cloud data.

**Qualitative results.** We visualize the results of our method compared with 3DGenZ [30] on three different datasets in Fig. 3. Our method performs better than 3DGenZ on all classes, especially on unseen classes e.g. On the ScanNet dataset our method is more successful in segmenting unseen bookshelf, whereas 3DGenZ is confused on unseen sofa. The same phenomenon occurs in window on the S3DIS dataset and bicyclist on the SemanticKITTI dataset. Our method can more effectively help the network to transfer knowledge from the seen to the unseen situation by synthesizing the unseen features being semantic-visual aware.

### 4.3. Ablation study

**Ablation study of different modules.** We progressively integrate different modules to study their contribution

| Methods      | Publication | miIoU | miIoU | All
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>C^v</td>
<td>C^u</td>
<td>All</td>
</tr>
<tr>
<td>SPNet [52]*</td>
<td>CVPR 2019</td>
<td>16.2</td>
<td>1.6</td>
<td>13.3</td>
</tr>
<tr>
<td>ZS3Net [6]*</td>
<td>NeurIPS 2019</td>
<td>33.6</td>
<td>4.1</td>
<td>27.7</td>
</tr>
<tr>
<td>CSRL [27]*</td>
<td>NeurIPS 2020</td>
<td>34.2</td>
<td>4.6</td>
<td>28.3</td>
</tr>
<tr>
<td>GaGNet [18]*</td>
<td>ACM MM 2020</td>
<td>33.8</td>
<td>5.2</td>
<td>28.1</td>
</tr>
<tr>
<td>PMOSR [55]*</td>
<td>ICCV 2021</td>
<td>32.5</td>
<td>5.4</td>
<td>27.1</td>
</tr>
<tr>
<td>Ours</td>
<td>ICCV 2023</td>
<td>34.5</td>
<td>14.3</td>
<td>30.4</td>
</tr>
</tbody>
</table>
Table 3. Ablation study of MCL (Sec. 3.2), HPA (Sec. 3.3) and RTC (Sec. 3.4) modules on ScanNet dataset. We observe that all three proposed modules contribute to the performance.

<table>
<thead>
<tr>
<th>MCL</th>
<th>HPA</th>
<th>RTC</th>
<th>mIoU</th>
<th>HmIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>×</td>
<td>×</td>
<td>×</td>
<td>34.2</td>
<td>7.7</td>
</tr>
<tr>
<td>×</td>
<td>✓</td>
<td>×</td>
<td>33.5</td>
<td>9.7</td>
</tr>
<tr>
<td>×</td>
<td>×</td>
<td>✓</td>
<td>33.5</td>
<td>10.0</td>
</tr>
<tr>
<td>✓</td>
<td>×</td>
<td>×</td>
<td>34.5</td>
<td>10.6</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>×</td>
<td>34.0</td>
<td>13.0</td>
</tr>
<tr>
<td>✓</td>
<td>×</td>
<td>✓</td>
<td>33.9</td>
<td>13.9</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>33.7</td>
<td>14.0</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>34.5</td>
<td>14.3</td>
</tr>
</tbody>
</table>

Table 4. Contributions of Contrastive Learning (CL) and Masking Strategy (MS) in MCL module (Sec. 3.2).

<table>
<thead>
<tr>
<th>Methods</th>
<th>mIoU</th>
<th>HmIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>34.2</td>
<td>12.5</td>
</tr>
<tr>
<td>+ CL</td>
<td>31.7</td>
<td>16.8</td>
</tr>
<tr>
<td>+ CL + MS</td>
<td>34.0</td>
<td>18.8</td>
</tr>
</tbody>
</table>

Table 5. Effects of different prototype generation strategies. For HPA, the neighbor-aware prototype generation works best, while for simple averaging performs better for RTC.

<table>
<thead>
<tr>
<th>Section</th>
<th>Prototype Construction Methods</th>
<th>mIoU</th>
<th>HmIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sec. 3.3 HPA</td>
<td>Simple Averaging</td>
<td>34.1</td>
<td>13.8</td>
</tr>
<tr>
<td></td>
<td>K-Means Clustering</td>
<td>32.7</td>
<td>13.8</td>
</tr>
<tr>
<td></td>
<td>Neighbor-Aware</td>
<td>34.5</td>
<td>14.3</td>
</tr>
<tr>
<td>Sec. 3.4 RTC</td>
<td>Simple Averaging</td>
<td>34.5</td>
<td>14.3</td>
</tr>
<tr>
<td></td>
<td>K-Means Clustering</td>
<td>32.3</td>
<td>13.3</td>
</tr>
<tr>
<td></td>
<td>Neighbor-Aware</td>
<td>33.4</td>
<td>12.1</td>
</tr>
</tbody>
</table>

Hyper-parameters. Fig. 4 shows the impact of two critical hyper-parameters (i.e., mask probability $q$ and visual prototypes ratio $\tau$). We observe that with a gradual increase in $q$, the models performance improves, indicating that the model is able to complete the visual features of the missing semantic embeddings according to the contextual information, so as to obtain better representations. However, the higher mask probability will result in a lack of sufficient semantics to assist generator training, resulting in performance degradation. The highest performance is achieved when the mask probability is 0.2 for all three datasets. In addition, we observe a low performance when the ratio is small for the visual prototypes. It is due to the insufficient prototypical representations in the visual space, which leads to the deviation in alignment with semantic vectors. Beyond $\tau > 0.04$, the performance starts to to decline, probably caused by over-fitting that leads to adverse impact.

Figure 4. Effect of hyper-parameters: mask probability $q$ and visual prototypes ratio $\tau$ on three datasets, $q = 0.2$ and $\tau = 0.04$ show the best results across all the datasets.

Effects of different auxiliary semantic embeddings. Tab. 6 compares different choices of auxiliary semantic embeddings (Word2Vec, GloVe and GloVe+Word2Vec for 300, 300, 600-dimensional semantic embeddings respectively). In general, a higher dimensional semantic embeddings produce richer feature representations, but we observe that more dimensional embeddings on different datasets may not always lead to the best results. Using the GloVe embeddings produces better performance in unseen mIoU and HM for ScanNet datasets, but not the other two, and similar phenomenon appears in the Word2Vec embeddings for SemanticKITTI. Only for S3DIS dataset, we achieve the best gain using a combination of Word2Vec and GloVe. We argue that the higher dimensional embedding space may bring more complexity and information redundancy for our
model, especially for ScanNet and SemanticKITTI, which contains relatively more classes than S3DIS.

<table>
<thead>
<tr>
<th></th>
<th>mIoU</th>
<th>mIoU</th>
<th>mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C^S</td>
<td>C^U</td>
<td>All</td>
</tr>
<tr>
<td>GloVe</td>
<td>33.3</td>
<td>11.7</td>
<td>29.0</td>
</tr>
<tr>
<td>Word2Vec</td>
<td>36.6</td>
<td>10.4</td>
<td>29.4</td>
</tr>
<tr>
<td>GloVe + Word2Vec</td>
<td>34.8</td>
<td>12.1</td>
<td>30.1</td>
</tr>
<tr>
<td>SN</td>
<td>58.9</td>
<td>14.4</td>
<td>39.2</td>
</tr>
<tr>
<td>S3</td>
<td>58.9</td>
<td>9.1</td>
<td>43.6</td>
</tr>
<tr>
<td>SK</td>
<td>45.8</td>
<td>14.4</td>
<td>39.2</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper, we propose a feature synthesis-based approach for Generalized Zero-Shot Semantic Segmentation of 3D point clouds. Our goal is to enhance semantic-visual correspondence, alignment and consistency, to learn generic representations that can transfer across novel unseen classes. Through our developed strategies, we promote the intra-class diversity in the visual features, while enhancing separation between classes. We further align the visual features and their semantics in the prototypical space, and preserve semantic-visual relationships through consistency regularization. Our empirical evaluations suggest that the proposed method can effectively segment point clouds for both seen and unseen classes at inference time, and achieve significant gains over the current state-of-the-art. For future work, we plan to extend our current approach for open-vocabulary zero-shot point cloud semantic segmentation.

Acknowledgement: This work was supported by the National Natural Science Foundation of China (No.62276176).

References


