Bridging Language and Geometric Primitives for Zero-shot Point Cloud Segmentation

Runnan Chen
The University of Hong Kong

Xinge Zhu
The Chinese University of Hong Kong

Nenglun Chen
The University of Hong Kong

Wei Li
Inceptio

Yuexin Ma
ShanghaiTech University

Ruigang Yang
Inceptio

Wenping Wang
Texas A&M University

ABSTRACT
We investigate transductive zero-shot point cloud semantic segmentation, where the network is trained on seen objects and able to segment unseen objects. The 3D geometric elements are essential cues to imply a novel 3D object type. However, previous methods neglect the fine-grained relationship between the language and the 3D geometric elements. To this end, we propose a novel framework to learn the geometric primitives shared in seen and unseen categories’ objects and employ a fine-grained alignment between language and the learned geometric primitives. Therefore, guided by language, the network recognizes the novel objects represented with geometric primitives. Specifically, we formulate a novel point visual representation, the similarity vector of the point’s feature, and/or a fee. Request permissions from permissions@acm.org.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM Reference Format:
Runnan Chen, Xinge Zhu, Nenglun Chen, Wei Li, Yuexin Ma, Ruigang Yang, and Wenping Wang. 2023. Bridging Language and Geometric Primitives for Zero-shot Point Cloud Segmentation. In Proceedings of the 31st ACM International Conference on Multimedia (MM ’23), October 29–November 3, 2023, Ottawa, ON, Canada. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3581783.3612409

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM Reference Format:
Runnan Chen, Xinge Zhu, Nenglun Chen, Wei Li, Yuexin Ma, Ruigang Yang, and Wenping Wang. 2023. Bridging Language and Geometric Primitives for Zero-shot Point Cloud Segmentation. In Proceedings of the 31st ACM International Conference on Multimedia (MM ’23), October 29–November 3, 2023, Ottawa, ON, Canada. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3581783.3612409

1 INTRODUCTION
Semantic segmentation on the point cloud is a fundamental task in autonomous driving, service robots, digital cities, etc. Although some recent methods [6, 10, 16, 22–25, 27, 38, 41, 42, 45] achieve promising performance, they heavily rely on labour-intensive annotations for supervision. By leveraging word embedding, zero-shot semantic segmentation can recognize the unseen objects whose labels are unavailable. It is beneficial for visual perception in a new scene that contains novel objects. It can also be a pre-annotation tool for automatically labelling the novel objects [5, 7–9, 28, 30].

Zero-shot learning (ZSL) focuses on transferring knowledge from seen to unseen categories. The preliminary ZSL setting predicts only unseen categories, while generalized ZSL (GZSL) predicts both seen and unseen categories. In the aspect of training data, it includes the inductive and transductive settings [30, 43]. Only the seen class samples and labels are available for training the network in the inductive setting. As for the transductive setting, the unlabeled objects of unseen classes are also accessible. In the semantic...
segmentation scenario, transductive GZSL is a more common setting for zero-shot segmentation (ZSS) because the seen and unseen categories often appear together in a scene. The ZSS problem we studied belongs to the transductive GZSL.

Impressive progress of ZSS has been achieved on the 2D images [3, 20, 21, 26, 43]. They typically generate the fake features of unseen categories for training the classifier or enhance the structure consistency between the visual features and semantic representation. ZSS is not fully explored in the 3D point cloud scenario. Only one method [30] investigates this problem to the best of our knowledge. It generates unseen class features with semantic embeddings for training the classifier. However, the fine-grained relationship between the language and the 3D geometric elements in seen and unseen categories, which are important to reason the unseen object types, are not explicitly considered.

In this paper, we investigate transductive zero-shot segmentation (ZSS) on the 3D point cloud, i.e., the visual features of unseen categories are available during training [30, 43]. Our key observation is that the 3D geometric elements are essential cues that imply a novel 3D object type (Figure 1). For example, chairs and sofas have similar geometric elements such as armrest, backrest and cushion, and they are also close in the semantic embedding space (Figure 4). Based on the observation, we propose a novel framework to learn the geometric primitives shared in seen and unseen categories and fine-grained align the language semantics with the learned geometric primitives. Specifically, inspired by the bag-of-words model [19, 39], we formulate a novel point visual representation that encodes geometric primitive information, i.e., the similarity vector of the point’s feature to the geometric primitives, where geometric primitives are a group of learnable prototypes updated by back-propagation. For bridging the language and the geometric primitives, we first conduct the language semantic representation as a mixture-distributed embedding, it is because a 3D object composed of multiple geometric primitives. Besides, the network is naturally biased towards the seen classes, leading to significant misclassifications of the unseen classes (Figure 5). To this end, we propose an Unknown-aware InfoNCE Loss that fine-grained aligns the visual and semantic representation at the same time alleviating the misclassification issue. Essentially, it pushes the unseen visual representations away from the seen categories’ semantic representations (Figure 4), enabling the network to distinguish the seen and unseen objects. In the inferring state, under the guidance of semantic representations, a novel object is represented with learned geometric primitives and can be recognized.

Extensive experiments conducted on S3DIS, ScanNet, Semantic-KITTI and nuScenes datasets show that our method outperforms other state-of-the-art methods, with the improvement of 17.8%, 30.4, 9.2% and 7.9%, respectively.

The contributions of our work are as follows.

- We propose a novel framework to model the fine-grained relationship between language and geometric primitives that transfer the knowledge from seen to unseen categories.
- We propose an Unknown-aware InfoNCE Loss for the fine-grained visual and semantic representation alignment among seen and unseen categories.
- Our method achieves state-of-the-art performance for zero-shot point cloud segmentation on S3DIS, ScanNet, Semantic-KITTI and nuScenes datasets.

2 RELATED WORK

2.1 Zero-Shot Segmentation on 2D images

Zero-shot semantic segmentation (ZSS) is dominated by generalized zero-shot learning because the objects of seen and unseen categories often appear in a scene together. ZS3Net [3] generates pixel-wise fake features from semantic information of unseen classes and then integrate the real feature of seen classes for training the classifier. Gu et al. [20] further improve ZS3Net by introducing a contextual module that generates context-aware visual features from semantic information. Li et al. [26] propose a Consistent Structural Relation Learning (CSRL) approach to model category-level semantic relations and to learn a better visual feature generator. However, they improperly use each unseen ground truth pixel locations for fake feature generation. Hu et al. [21] promote the performance by alleviating noisy and outlying training samples from seen classes with Bayesian uncertainty estimation. There is an obvious bias between real and fake features that hinder the knowledge transferring from seen classes to unseen classes. Zhang et al. [43] replace the unseen objects with other images to generate training samples and perform segmentation with prototypical matching and open set rejection. Lv et al. [29] mitigate this issue using a transductive setting that uses both labelled seen images and unlabeled unseen images for training. In this paper, we follow the transductive setting that leverages unseen objects’ features for supervision, and the ground truth pixel locations of individual unseen objects are not accessible, which naturally meets the semantic segmentation.

2.2 Zero-shot Learning on 3D Point Cloud

Unlike promising progress of zero-shot learning on 2D images, there are few studies conducted on the 3D point cloud. Some methods [11–14] are to study point cloud classification. Cheraghian et al. [14] adapts pointNet [34] to extract object representation and GloVe [33] or W2V [32] to obtain semantic information for reasoning the unseen object’s types. Cheraghian et al. [11] adapt GZSL setting and propose a loss function composed of a regression term [44] and a skewness term [35, 36] to alleviate the hubness problem, which indicates that the model may predict only a few target classes for most of the test instances. Cheraghian et al. [12] further improves [11] by using a triplet loss. To the best of our knowledge, only one method [30] is proposed for semantic segmentation that generates fake features with class prototypes for training the classifier. However, they do not explicitly consider the 3D geometric elements shared across the seen and unseen categories, which are important cues to align the 3D visual features and semantic embedding. In this paper, our method learns geometric primitives that transfer the knowledge from seen to unseen categories for more accurate reasoning of unseen category objects. Besides, instead of generating fake features for the unseen classes, we naturally leverage unseen visual features extracted from the backbone network, which is more intuitive and natural under the transductive setting.
while the labels are unavailable. The goal is to generate pixel-level \(X, W, Y\) whose labels are unavailable during training. Suppose (hollow circles) is formulated as the similarity vector of its feature to the prototypes \(D\) is no overlap between the seen and unseen classes. 

Our framework is illustrated in Figure 2. The training stage contains two modules, i.e., Visual Representation with Geometric Primitives and Fine-grained Alignment on Visual and Semantic Representation. 

3.2 Approach Overview

Our framework is illustrated in Figure 2. The training stage contains two modules, i.e., Visual Representation with Geometric Primitives and Fine-grained Alignment on Visual and Semantic Representation. Firstly, we extract the point-wise feature of the seen and unseen categories and formulate the point visual representation as the similarity vector between geometric primitives and point features. Next, we perform fine-grained alignment on visual and semantic representation for both seen and unseen categories via an Unknown-aware InfoNCE Loss. In the inferring stage, our method represents unseen category points with the learned geometric primitives and infers the specific unseen class under the guidance of the semantic representations. In what follows, we present these modules in detail.

3.3 Visual Representation with Geometric Primitives

Unlike 2D images in the projected view, the 3D object contains complete geometric information. We observe that 3D shapes are composed of common 3D structure elements, such as the cube, cuboid, sphere, cone, cylinder, pyramid, torus, etc. Thus, our basic idea is to disentangle the 3D objects into a set of learnable geometric primitives (Figure 3). The geometric primitives could constitute a novel 3D object. For example, a simple chair consists of a cushion and four legs (cylinders). Therefore, our method first learns the shared 3D geometric elements in seen and unseen categories and then takes them to represent novel objects.

Inspired by the bag-of-words model [19, 39] that a sentence can be a dictionary, and its semantic depends on each word’s frequency. Thus, our basic idea is to disentangle the 3D objects into a set of learnable geometric primitives (Figure 3). The geometric primitives could constitute a novel 3D object. For example, a simple chair consists of a cushion (cuboid) and four legs (cylinders). Therefore, our method first learns the shared 3D geometric elements in seen and unseen categories and then takes them to represent novel objects.

In the paper, we focus on the transductive zero-shot segmentation, which is more common in the semantic segmentation task because the seen and unseen categories often appear together. Notably, "unseen" here indicates that visual features are accessible while the labels are unavailable. The goal is to generates pixel-level segmentation masks for both seen and unseen categories.
where \( \{g_m\}_{m=1}^M \) are \( M \) learned geometric primitives. \( \{\alpha^i_{t,m}\}_{m=1}^M \) are the weights of the corresponding \( g_m \), defined:

\[
\alpha^i_{t,m} = \frac{\exp(\lambda \times d(\theta(x^i_t), \varphi(g_m)))}{\sum_{m=1}^M \exp(\lambda \times d(\theta(x^i_t), \varphi(g_m)))},
\]

where \( x^i_t \in X_t \) is the \( t \)-th point visual feature. \( d(\cdot) \) measures the similarity between the point feature \( x^i_t \) and the \( m \)-th geometric primitives \( g_m \). We utilize dot product operation in this work. \( \theta(\cdot) \) and \( \varphi(\cdot) \) denote the key and the query function, respectively. \( \lambda \) is the inversed temperature term. Note that \( \alpha^i_{t,m} \) is positive and the sum of possibilities is restricted to \( 1 \) to prevent network deterioration, i.e., \( \sum_{m=1}^M \alpha^i_{t,m} = 1 \).

As the geometric primitives are shared for all categories, we simplify the point representation to be a \( M \) dimensional vector:

\[
\hat{x}^i_t = [\alpha^i_{t,1}, \alpha^i_{t,2}, \alpha^i_{t,3}, ..., \alpha^i_{t,M}].
\]

Essentially, \( \hat{x}^i_t \) is a possibility distribution that indicates the proportion of the geometric primitives. In this way, a similar distribution implies the same category, and distinct primitives distribution indicates different categories.

### 3.4 Fine-grained Alignment on Visual and Semantic Representation

In this section, we perform fine-grained alignment on the visual and semantic representation in the aspect of the semantic representation formulation and the Unknown-aware InfoNCE Loss design. In the end, we present how the network infers a scene for both seen and unseen objects.

**Semantic Representation of Mixture Distribution.** 3D object’s visual representation should be the mixture distribution as it consists of multiple geometric primitives. Therefore, we generate the semantic representation \( \hat{w}_c \) with multi-kernels from the word embeddings for the fine-grained alignment of the visual-semantic representation.

\[
\hat{w}_c = \sum_{k=1}^K \hat{w}_{c,k}, \quad \hat{w}_{c,k} = G_k(w_c),
\]

where \( \{w_c\}_{C_s^u+C_u^u} \) are the word embeddings. They are the output embeddings of the word2vec [31] or glove [33] model with the input of categories’ names. \( C_s^u \) and \( C_u^u \) are the number of seen and unseen categories, respectively. \( K \) is the number of kernels. \( \{G_k(\cdot)\}_{k=1}^K \) are the generation networks.

**Unknown-aware InfoNCE Loss.** We propose an unknown-aware InfoNCE Loss to model the fine-grained relationship between the visual (geometric primitive distribution) and the semantic representation. The loss function is designed for two purposes: 1) distinguishing the specific seen category, 2) enabling the network to identify whether the object is the seen or unseen category. In what follows, we present the details and insights.

To efficiently learn the geometric primitives shared in seen and unseen classes, we cluster these points of seen classes to ensure they are inner-class compact and inter-class distinguishable. i.e., pull in the corresponding visual and semantic representation while pushing away if they are not. The objective function is as follows:

\[
L_s = -\log \sum_{i=1}^{N_s} \sum_{t=1}^{T_i} \frac{\exp(D(\hat{x}^i_t, \hat{w}_{y^i_t})/\tau)}{\sum_{c=1}^{C_s^c+C_u^u} \exp(D(\hat{x}^i_t, w_c)/\tau)},
\]

where \( \hat{w}_{y^i_t} \) is the ground truth semantic representation of the \( t \)-th point in the \( i \)-th sample. \( \tau \) is the inversed temperature term. \( C_s^c \) and \( C_u^u \) are the number of seen and unseen categories, respectively. \( D(\cdot) \) indicates the similarity function between visual and semantic representation, defined as follows:

\[
D(\hat{x}^i_t, \hat{w}_c) = \sum_{k=1}^K d(\hat{x}^i_t, \hat{w}_{c,k}),
\]

where
Table 1: Comparison with state-of-the-art methods on S3DIS, ScanNet, semanticKITTI and nuScenes dataset by evaluation metrics of mIoU and hIoU. † indicates the adaptation of 2D methods to 3D point clouds, reported in [30]. “Supervised” indicates the fully supervised version reported in [30], while “Supervised” is with our backbone network.

| Model         | S3DIS (seen) | S3DIS (unseen) | ScanNet (seen) | ScanNet (unseen) | SemanticKITTI (seen) | SemanticKITTI (unseen) | NuScenes (seen) | NuScenes (unseen) |
|---------------|--------------|----------------|----------------|-----------------|---------------------|----------------------|----------------|------------------|
| Supervised†   | 74.0         | 50.0           | 66.6           | 59.6            | 43.3                | 51.9                 | 45.1           | 47.2             |
| Supervised    | 71.8         | 25.9           | 57.7           | 38.1            | 56.4                | 69.8                 | 59.1           | 62.4             |
| ZSLFPC [14]   | 5.2          | 1.3            | 4.0            | 2.1             | 16.4                | 4.2                  | 13.9           | 6.7              |
| Devis e† [18] | 3.6          | 1.4            | 3.0            | 2.0             | 12.8                | 3.0                  | 10.9           | 4.8              |
| ZSSNet [3]    | 56.6         | 4.8            | 40.6           | 8.8             | 56.8                | 3.9                  | 46.2           | 7.3              |
| PMOSR [43]    | 57.4         | 6.3            | 41.6           | 11.4            | 55.3                | 5.4                  | 45.3           | 9.8              |
| 3DGenZ [30]   | 53.1         | 7.3            | 39.0           | 12.9            | 32.8                | 7.7                  | 27.8           | 12.5             |
| Ours          | **60.4**     | **20.6**       | **48.2**       | **30.7**        | **57.9**            | **34.1**             | **53.1**       | **42.9**         |
|               |              |                |                |                 | **54.6**            | **17.3**             | **46.7**       | **26.3**         |
|               |              |                |                |                 |                     |                      |                | **65.7**         |

$\text{mIoU}_{\text{seen}} = \frac{\sum_{i=1}^{N_s} \sum_{c=1}^{C_s} \exp(D(x_i^c, \hat{w}_c))}{\sum_{c=1}^{C_s} \exp(D(x_i^c, \hat{w}_c))}$

Note that the ground truth label $\hat{w}_c$ of unseen points are not accessed. Essentially, $L_u$ contains a softmax operation on the possibility distribution among seen and unseen classes. It enforces the possible sum of seen classes to be 0 and the possible sum of unseen classes to be 1. The final loss function is the combination of two items:

$L_f = L_s + L_u$.  

**Inference**. When inferring an new scene scan with the point features $\{x_i^j\}_{i=1}^{T_j}$ obtained by the backbone network $F(\cdot)$, where $T_j$ is the number of scene points. The $t$-th point feature $x_i^j$ is transferred to the visual representation $\hat{x}_i^j$ with formula 1, 2, 3, and its class $C_i^j$ is determined by the following formula.

$C_i^j = \arg \max_c \frac{\exp(D(\hat{x}_i^j, \hat{w}_c))}{\sum_{c=1}^{C_s+c_w} \exp(D(\hat{x}_i^j, \hat{w}_c))}$.

where $\hat{w}_c$ is the semantic representation for seen and unseen classes.

**S3DIS**. S3DIS includes 271 scanned rooms with points labelled among 13 classes. We utilize Area 5 as the validation set and use the other five areas as the training set. To comprehensively evaluate the performance, we also conduct zero-shot setups with the different number of unseen classes, constructing the 2-, 4-, 6-class unseen sets. The detailed splits are: 2-sofa/beam, 4-column/window, 6-bookshelf/board. Note that the classes in the unseen set increase incrementally for different setups, e.g., the 4-unseen set contains the 2-unseen set, and the 6-unseen set contains the 4-unseen set.

**ScanNet**. ScanNet is an indoor scene dataset that contains 1513 scans with annotations for 20 classes, where 1201 scans for training and 312 scans for validation. We conduct zero-shot setups with the different number of unseen classes, including the 2-sofa/desk, 4-bookshelf/toilet, 6-bath/tub/bed, 8-curtain/window, 10-door/counter. The classes in the unseen set increase incrementally for different setups.

**SemanticKITTI**. SemanticKITTI is a large-scale driving-scene dataset for point cloud segmentation, SemanticKITTI contains 22 sequences, where ten sequences for training and sequence 08 for validation. We conduct zero-shot setups with the different number of unseen classes, including the 2-Motorcycle/truck, 4-bicyclist/traffic-sign, 6-car/terrain, 8-vegetation/sidewalk. The classes in the unseen set increase incrementally for different setups.

**nuScenes**. nuScenes contains 16 classes for LiDAR semantic segmentation, where 850 scans for training and validation, and the remaining 150 scans are for testing. We conduct several zero-shot settings with different numbers of unseen classes, including the 2-Motorcycle/trailer, 4-terrain/traffic-cone, 6-bicycle/car, 8-vegetation/sidewalk.

**4.2 Evaluation Metrics**

We report the mean-intersection-over-union (mIoU) in experiments. Besides, following [3, 40, 43], we adopt the harmonic mean of seen mIoU and unseen mIoU ($hIoU$) to demonstrate the overall performance of ZSS.

$hIoU = \frac{2 \times \text{mIoU}_{\text{seen}} \times \text{mIoU}_{\text{unseen}}}{\text{mIoU}_{\text{seen}} + \text{mIoU}_{\text{unseen}}}$.

where $\text{mIoU}_{\text{seen}}$ and $\text{mIoU}_{\text{unseen}}$ represents the mean IoU of seen classes and unseen classes respectively.
Table 2: Results on S3DIS dataset with the evaluation metrics of mIoU and hIoU. We show the performance of the setups with the different numbers (2-sofa/beam, 4-column/window, 6-bookshelf/board) of unseen classes. 'Supervised' is trained on labelled seen and unseen categories with our backbone network.

| setting | Model       | Seen mIoU | Unseen mIoU | Overall mIoU | hIoU |
|---------|-------------|-----------|-------------|--------------|------|
| 0       | Supervised  | -         | -           | 59.1         | -    |
| 2       | Devise[18]  | 52.1      | 4.0         | 47.3         | 7.4  |
|         | 3DGenZ [30] | 33.4      | 12.8        | 31.3         | 18.5 |
|         | Ours        | 58.6      | 51.6        | 57.9         | 54.9 |
| 4       | Devise[18]  | 50.9      | 1.9         | 41.1         | 3.6  |
|         | 3DGenZ [30] | 32.8      | 7.7         | 27.8         | 12.5 |
|         | Ours        | 57.9      | 34.1        | 53.1         | 42.9 |
| 6       | Devise[18]  | 48.2      | 2.2         | 34.4         | 4.2  |
|         | 3DGenZ [30] | 31.2      | 4.8         | 23.3         | 8.3  |
|         | Ours        | 55.2      | 15.4        | 43.2         | 24.1 |
| 8       | Devise[18]  | 54.9      | 2.0         | 33.7         | 3.9  |
|         | 3DGenZ [30] | 29.7      | 2.1         | 18.7         | 3.9  |
|         | Ours        | 51.6      | 12.1        | 35.8         | 19.6 |
| 10      | Devise[18]  | 51.5      | 0.1         | 25.8         | 0.2  |
|         | 3DGenZ [30] | 30.1      | 1.4         | 15.8         | 2.7  |
|         | Ours        | 52.5      | 9.5         | 31.0         | 16.1 |

Table 3: Results on ScanNet dataset with the evaluation metrics of mIoU and hIoU. We show the performance of the setups with the different numbers (2-sofa/desk, 4-bookshelf/toilet, 6-bathtub/bed, 8-curtain/window, 10-door/counter) of unseen classes. 'Supervised' is trained on labelled seen and unseen categories with our backbone network.

| setting | Model       | Seen mIoU | Unseen mIoU | Overall mIoU | hIoU |
|---------|-------------|-----------|-------------|--------------|------|
| 0       | Supervised  | -         | -           | 59.1         | -    |
| 2       | Devise[18]  | 55.1      | 10.5        | 48.2         | 17.6 |
|         | 3DGenZ [30] | 50.4      | 9.1         | 44.0         | 15.4 |
|         | Ours        | 60.2      | 26.3        | 55.0         | 36.6 |
| 4       | Devise[18]  | 59.0      | 8.7         | 43.5         | 15.2 |
|         | 3DGenZ [30] | 53.1      | 7.3         | 39.0         | 12.9 |
|         | Ours        | 61.1      | 22.1        | 49.1         | 32.5 |
| 6       | Devise[18]  | 68.3      | 3.8         | 38.5         | 7.2  |
|         | 3DGenZ [30] | 56.4      | 3.5         | 32.0         | 6.6  |
|         | Ours        | 69.5      | 8.8         | 41.5         | 15.6 |

Table 4: Results on semanticKITTI dataset. We show the performance of the setups with the different numbers (2-Motorcycle/truck, 4-bicyclist/traffic-sign, 6-car/terrain, 8-vegetation/sidewalk) of unseen classes. 'Supervised' is trained on labelled seen and unseen categories with our backbone network.

| setting | Model       | Seen mIoU | Unseen mIoU | Overall mIoU | hIoU |
|---------|-------------|-----------|-------------|--------------|------|
| 0       | Supervised  | -         | -           | 59.1         | -    |
| 2       | Devise[18]  | 53.7      | 7.6         | 48.8         | 13.3 |
|         | 3DGenZ [30] | 40.9      | 12.4        | 37.9         | 19.0 |
|         | Ours        | 58.3      | 28.8        | 55.2         | 38.6 |
| 4       | Devise[18]  | 51.4      | 4.2         | 41.5         | 7.8  |
|         | 3DGenZ [30] | 41.4      | 10.8        | 35.0         | 17.1 |
|         | Ours        | 55.9      | 22.8        | 48.9         | 32.4 |
| 6       | Devise[18]  | 48.8      | 2.4         | 34.1         | 4.6  |
|         | 3DGenZ [30] | 40.3      | 6.5         | 29.6         | 11.2 |
|         | Ours        | 53.6      | 13.3        | 40.9         | 21.3 |
| 8       | Devise[18]  | 49.2      | 0.5         | 28.7         | 1.0  |
|         | 3DGenZ [30] | 38.3      | 1.3         | 22.7         | 2.5  |
|         | Ours        | 53.2      | 8.6         | 34.4         | 14.8 |

Table 5: Results on nuScenes dataset. We show the performance of the setups with the different numbers (2-Motorcycle/trailer, 4-terrain/traffic-cone, 6-bicycle/car, 8-vegetation/sidewalk) of unseen classes. 'Supervised' is trained on labelled seen and unseen categories with our backbone network.

| setting | Model       | Seen mIoU | Unseen mIoU | Overall mIoU | hIoU |
|---------|-------------|-----------|-------------|--------------|------|
| 0       | Supervised  | -         | -           | 69.1         | -    |
| 2       | Devise[18]  | 53.7      | 7.6         | 48.8         | 13.3 |
|         | 3DGenZ [30] | 67.8      | 4.2         | 59.9         | 7.9  |
|         | Ours        | 58.9      | 26.9        | 54.9         | 36.9 |
| 4       | Devise[18]  | 51.4      | 4.2         | 41.5         | 7.8  |
|         | 3DGenZ [30] | 67.2      | 3.1         | 51.2         | 5.9  |
|         | Ours        | 65.7      | 14.8        | 53.0         | 24.2 |
| 6       | Devise[18]  | 48.8      | 2.4         | 34.1         | 4.6  |
|         | 3DGenZ [30] | 53.8      | 3.2         | 34.8         | 6.0  |
|         | Ours        | 68.8      | 14.1        | 48.3         | 23.4 |
| 8       | Devise[18]  | 49.2      | 0.5         | 28.7         | 1.0  |
|         | 3DGenZ [30] | 36.5      | 2.1         | 19.3         | 4.0  |
|         | Ours        | 68.4      | 13.7        | 41.1         | 22.8 |

4.3 Implementation Details

We adopt MinkowskiNet14 [15] as backbone for ScanNet and S3DIS dataset. The point visual feature dimension is set to be 96. The input and output of the key $\theta(\cdot)$ and the query $\phi(\cdot)$ function are 96 and 16-dimensional vectors, respectively. The generation network $G(\cdot)$ is a two-layer MLP network with the dimension of 96 and $M \times K$, where $M$ and $K$ are the numbers of prototypes and semantic representation kernel, and set to be 128 and 16, respectively. The voxel size of all experiments is set to be 5 cm for efficient training. And the temperature term $\lambda$ is set to be 4. We adopt the codebase Cylinder3D [45] and apply cartesian coordinates for the semanticKITTI dataset. Following [30], we use the W2V+GloVe word embeddings (600-dimensional vector) as the auxiliary semantic information. Our method is built on the Pytorch platform, optimized by Adam with the default configuration. The batch size for the S3DIS, ScanNet, semanticKITTI and nuScenes are 8, 8, 4 and 8, respectively. Training 300 epochs cost 2 hours on four RTX 2080 TI GPUs for the S3DIS dataset, 8 hours for ScanNet and training 20 epochs cost 40 hours for the semanticKITTI dataset, 15 hours for the nuScenes dataset.
Figure 5: Qualitative results on ScanNet. From the first row to the last row are input scene (point cloud), ground truth, prediction without ZSS, and our prediction, respectively. From the left column to the right are the unseen classes sofa (brown colour on the ground truth), desk (pink) and toilet (green) and bookshelf (purple). Note that black is the unlabeled region. The model without zero-shot segmentation misclassifies the unseen classes to some seen classes. For example, in the second column, the sofa (the brown region) is misclassified into the table and chair. In comparison, our method achieves decent performance.

4.4 Results and Discussion

In this section, we report the comparisons with other state-of-the-art methods on four datasets, i.e., S3DIS, ScanNet, SemanticKITTI and NuScenes. Following [30], we benchmark against previous methods in Table 1. Besides, follow the zero-shot segmentation on 2D images [3, 43], we conduct zero-shot setups with the different number of unseen classes to comprehensively evaluate the performance (Table 2, 3, 4, 5). We also qualitatively evaluate our method and the counterpart trained only on seen classes (Figure 5). In the end, we discuss the limitations and potential directions for future works.

**Baselines.** To our knowledge, only one method (3DGenZ [30]) investigates the zero-shot point cloud segmentation. We directly compare the method on all settings. Besides, we also adapt three 2D methods [3, 18, 43] to 3D point cloud in the transductive setting for a fair comparison, i.e., the backbone is the same as ours, and we pseudo-labelling the unseen points via the Argmax operation on unseen categories and retrain the network with the pseudo labels. Specifically, we directly show the performance of Devise† in [30]. We adopt the ZSSNet version in [3], i.e., the model’s own predictions on unseen data is additional pseudo-labelled training data. And the network is trained with labelled seen data and pseudo-labelled unseen data. As for [43], we use the unseen visual features to train the "unknown visual prototype", named PMOSR in Table 3.

**Comparison with state-of-the-art methods.** Following [30], we take four classes as unseen categories for three datasets. Specifically, Beam, column, window and sofa are unseen categories for S3DIS datasets; Desk, bookshelf, sofa, toilet for ScanNet datasets; Motorcycle, truck, bicyclist and traffic-sign for SemanticKITTI dataset; and motorcycle, trailer, terrain, traffic-cone for NuScenes dataset. As shown in Table 1, our method outperforms other state-of-the-art methods by a large margin, with the hIoU improvement of 17.8%, 30.4%, 9.2% and 7.9% in four datasets, respectively. Note that our method also works well on the outdoor lidar point cloud dataset (semanticKITTI and nuScenes), showing the generalization ability when the point cloud is sparse.

**Performance under different number of unseen classes.** To evaluate the performance, we conduct zero-shot setups with the different numbers of unseen classes in the four datasets. As shown in Table 2, 4, 3, 5, our method significantly outperforms other state-of-the-art methods, showing the method could handle the different number of unseen classes. As shown in Table 2, our method achieves the harmonic mean of seen mIoU and unseen mIoU (hIoU) of 36.6%,
Table 6: Ablation experiments on ScanNet dataset with 4-unseen class. Base+兰_u+GP_{128}+MK_{16} is our full method.

| Model                | Seen mIoU | Unseen mIoU | Overall mIoU | hIoU |
|----------------------|-----------|-------------|--------------|------|
| Base                 | 54.8      | 43.8        | 48.6         | 31.6 |
| Base+兰_pseudo       | 53.3      | 3.6         | 45.0         | 6.8  |
| Base+兰_self         | 54.2      | 16.1        | 46.6         | 24.8 |
| Base+兰_u            | 55.2      | 22.1        | 48.6         | 31.6 |
| Base+兰_u+MK_{16}    | 57.3      | 29.4        | 51.7         | 38.9 |
| Base+兰_u+GP_{128}+MK_{2} | 57.1 | 31.6        | 52.0         | 40.7 |
| Base+兰_u+GP_{128}+MK_{4} | 57.9 | 33.1        | 52.9         | 42.1 |
| Base+兰_u+GP_{128}+MK_{8} | 57.7 | 33.3        | 52.8         | 42.2 |
| Base+兰_u+GP_{128}+MK_{16} | 57.9 | 34.1        | 53.1         | 42.9 |
| Base+兰_u+GP_{128}+MK_{32} | 57.1 | 33.1        | 52.3         | 41.9 |
| Base+兰_u+GP_{48}+MK_{16} | 53.4 | 33.1        | 49.3         | 40.9 |
| Base+兰_u+GP_{96}+MK_{16} | 54.9 | 32.9        | 50.5         | 41.1 |
| Base+兰_u+GP_{128}+MK_{16} | 57.9 | 34.1        | 53.1         | 42.9 |
| Base+兰_u+GP_{256}+MK_{16} | 57.4 | 26.2        | 51.2         | 36.0 |

32.5% and 15.6% on 2-, 4- and 6- unseen class, respectively. In the ScanNet 2-, 4-, 6-, 8-, and 10- unseen classes setting, the improved hIoU range from 14% ~ 36% (Table 3). For the semanticKITTI dataset, our method outperforms other state-of-the-art methods by a large margin, with the mIoU improvement of 19%, 15%, 10% and 12% in the 2-, 4-, 6-, and 8- unseen classes setting (Table 4). As for NuScenes dataset, the improved hIoU is 19%, 10%, 12% and 13% in the 2-, 4-, 6- and 8- unseen classes setting (Table 5). Moreover, we show the upper bound performance with full supervision (refer to setting 0).

Surprisingly, the method achieves high unseen mIoU in the sofa and desk categories (51.6% mIoU), close to the supervision with ground truth (Table 3). The main reason is that sofa and desk have similar semantic representations to the seen classes chair and table, respectively (Figure 4). Constrained by the Unknown-aware InfoNCE Loss, the network concludes that those unknown objects semantically closed to the chair are probably a sofa, and those close to the table are probably a desk. We also show the performance by training on the annotated seen and unseen categories (Supervised), which is 57.7% mIoU. However, as the number of unseen classes increases, the mIoU and hIoU gradually reduce. It is because the search space for reasoning the unseen categories is increased accordingly.

**Qualitative Evaluation.** As shown in Figure 5, we qualitatively evaluate our method and the counterpart trained only on seen classes (w/o ZSS) on the ScanNet dataset, where sofa, desk, toilet and bookshelf are unseen classes, and the rest of classes are seen classes. If only training on the seen categories, the network misclassifies the unseen object into seen objects. For example, the sofa is recognized to be the chair (the first column) or table (the second column); the desk is to be the table; the toilet is to be other furniture; the bookshelf to be the wall (the fifth column) or cabinet (the last column). It shows that the network is naturally biased towards the training classes, leading to significant misclassifications for those unseen categories whose semantic representations are similar to those seen. Our method alleviates such issues and achieves decent performance in recognizing unseen objects.

### 4.5 Ablation Study

As shown in Table 6, we conduct experiments on ScanNet 4-unseen class setting to verify the effectiveness of different components in our method, including geometric primitives-based visual representation (GP), the multi-kernels semantic representations (MK) and Unknown-aware InfoNCE Loss (兰_u). We first conduct the baseline method by following operation: 1) we remove the prototypes and directly use the point feature extracted from the backbone network as its visual representation; 2) we generate the one-kernel semantic representation from word embedding; 3) we perform visual and semantic alignment only on seen class points by the loss L_4. In what follows, we present more details and insights.

**Effect of Geometric Primitives.** To verify the effectiveness of the Geometric Primitives, we take the point feature extracted from the backbone network as its visual representation (Base+兰_u+MK_{16}). Compared with our full method (Base+兰_u+GP_{128}+MK_{16}), the unseen mIoU is dropped by about 5%, showing the visual representation with geometric primitives beneficial for knowledge transferring from seen to unseen categories. Besides, when the geometric primitives number M is 48, 96, 128 and 256 (Base+兰_u+GP_{48,96,128,256}+MK_{16}), we find that too big or too small M harms the performance.

**Effect of Semantic Representations.** By applying the multi-kernel semantic representations (Base+兰_u+MK_{16}), the mIoU is 7% larger than that without it (Base+兰_u), indicating the multi-kernel semantic representations well match the mixture-distributed visual representation. We also conduct the experiments with the different numbers of kernels, where K is 2, 4, 8, 16 and 32. (Base+兰_u+GP_{128}+MK_{2,4,8,16,32}). We choose K to be 16 in our full method, achieving the best performance.

**Effect of Unknown-aware InfoNCE Loss.** We implement multiple loss functions to constrain the unseen class points. L_pseudo indicates pseudo-labeling the unseen points via the Argmax operation on unseen categories and retrains the network with the pseudo labels. L_self is borrowed from Fixmatch [37] that imposes prediction consistency of unseen points on two augmented scenes. The last one is L_4 described in formula 7. The comparison results show our Unknown-aware InfoNCE loss’s superiority (Base+兰_pseudo, Base+兰_self and Base+兰_u).

### 5 CONCLUSIONS

We investigate transductive zero-shot semantic segmentation in this paper. Our method models the fine-grained relationship between language and geometric primitives that transfers knowledge from seen to unseen. To this end, we propose a novel visual feature representation and a novel Unknown-aware InfoNCE Loss. Therefore, under the guidance of semantic representation, the network can segment the novel objects represented by the learned geometric primitives. Extensive experiments conducted on four datasets show that our method significantly outperforms other state-of-the-art methods.
