Few-shot 3D Point Cloud Segmentation via Relation Consistency-guided Heterogeneous Prototypes

Lili Wei, Congyan Lang, *Member, IEEE*, Zheming Xu, Liqian Liang, Jun Liu, *Senior Member, IEEE*

***Abstract*—Few-shot 3D point cloud semantic segmentation is a challenging task due to the lack of labeled point clouds (support set). To segment unlabeled query point clouds, existing prototype-based methods learn 3D prototypes from point features of the support set and then measure their distances to the query points. However, such homogeneous 3D prototypes are often of low quality because they overlook the valuable heterogeneous information buried in the support set, such as semantic labels and projected 2D depth maps. To address this issue, in this paper, we propose a novel Relation Consistency-guided Heterogeneous Prototype learning framework (RCHP), which improves prototype quality by integrating heterogeneous information using large multi-modal models (*e.g.* CLIP). RCHP achieves this through two core components: Heterogeneous Prototype Generation module which collaborates with 3D networks and CLIP to generate heterogeneous prototypes, and Heterogeneous Prototype Fusion module which effectively fuses heterogeneous prototypes to obtain high-quality prototypes. Furthermore, to bridge the gap between heterogeneous prototypes, we introduce a Heterogeneous Relation Consistency loss, which transfers more reliable inter-class relations (*i.e.*, inter-prototype relations) from refined prototypes to heterogeneous ones. Extensive experiments conducted on five point cloud segmentation datasets, including four indoor datasets (S3DIS, ScanNet, SceneNN, NYU Depth V2) and one outdoor dataset (Semantic3D), demonstrate the superiority and generalization capability of our method, outperforming state-of-the-art approaches across all datasets.**

***Index Terms*—Few-shot, point cloud semantic segmentation, heterogeneous prototype, relation consistency**

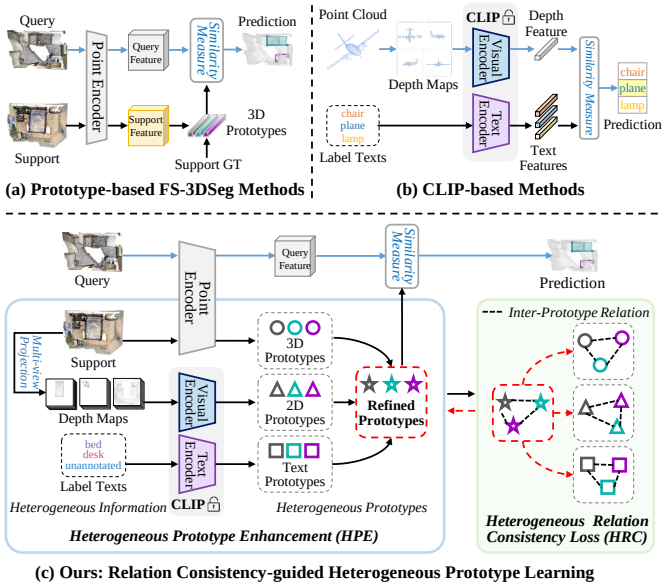


Fig. 1. Illustration of the FS-3DSeg methods. (a) Previous prototype-based FS-3DSeg methods only exploit 3D support information. (b) PointCLIP [16] bridges the gap between point clouds and texts by projecting depth maps. (c) Our method leverages inherent heterogeneous information of the support set to obtain the heterogeneous-enhanced prototypes guided by heterogeneous relation consistency loss.

I. Introduction

SEMANTIC segmentation is a fundamental task in computer vision, encompassing diverse areas such as image segmentation [1], video segmentation [2], [3]), 3D point cloud segmentation [4]–[7], *etc.* Among these, point cloud semantic segmentation (3DSeg) aims to assign semantic labels to each point in a 3D point cloud. Over the past decade, fully supervised 3DSeg methods [4]–[7] have achieved remarkable progress. However, these methods rely heavily on extensive labeled data and struggle to segment novel categories in open-set scenarios. To tackle these issues, recent years have witnessed the development of **f**ew-**s**hot **3D** point cloud semantic **seg**mentation (FS-3DSeg) [8]–[15] which aims to segment unlabeled point clouds (*i.e.*, query set) of new categories by leveraging knowledge learned from a few labeled point clouds (*i.e.*, support set).

The key challenge in FS-3DSeg lies in effectively utilizing the limited support set, including 3D point clouds and corresponding ground truth (GT) label masks. As shown in Fig. 1 (a), most FS-3DSeg methods [8]–[14] typically adopt prototype-based paradigms, learning 3D prototypes from support point features, and measuring distances between query point features and 3D prototypes to assign labels to the query set. However, their performance is still far from satisfactory since they only leverage partial, incomplete and biased semantic information in limited support point clouds as guidance, overlooking valuable heterogeneous information buried in the support set that could enhance 3D representation. We can observe two phenomenons: 1) due to the scarcity of support point clouds, they lack intra-class diversity, whereas labels

Manuscript received 25 August 2024; revised 4 December 2024 and 20 January 2025; accepted 3 February 2025. Date of publication ***; date of current version ***. The Guest Editor coordinating the review of this manuscript and approving it for publication was Prof. Mengshi Qi. *(Corresponding author: Congyan Lang.)*

Lili Wei, Congyan Lang, Zheming Xu, Liqian Liang are with the Key Laboratory of Big Data & Artificial Intelligence in Transportation, Ministry of Education, China, and also with the School of Computer Science & Technology, Beijing Jiaotong University, Beijing 100044, China (e-mail: 20112014@bjtu.edu.cn; cylang@bjtu.edu.cn; 21112016@bjtu.edu.cn; lqliang@bjtu.edu.cn).

Jun Liu is with Lancaster University, Lancaster, United Kingdom, LA1 4YW (e-mail: j.liu81@lancaster.ac.uk).

Digital Object Identifier ***.

names from GT masks provide richer semantic guidance; 2) due to the sparsity and irregularity of support point clouds, they only capture geometric structures of objects without perceiving visual information such as shapes and boundaries. In contrast, 2D depth maps [16] projected from point clouds provide clearer visual cues, as most points tend to fall along the boundary edges after projection. Driven by these facts, we aim to fully unleash the potential of the support set by simultaneously harvesting inherent heterogeneous information from the support set, including original point clouds, label names buried in GT masks, and visual information from projected 2D depth maps.

To leverage heterogeneous support information, we draw inspiration from Large Multi-modal Models (LMM), such as CLIP [17], which is pre-trained on large-scale visual-language data and has obtained a strong capacity to process data in multi-modalities. As shown in Fig. 1 (b), PointCLIP [16] projects point clouds into depth maps and leverages CLIP's image-text alignment for depth map classification. However, the sparsity and irregularity of point clouds, as well as information loss during projection and feature extraction, make this method unsuitable for directly handling FS-3DSeg task. In contrast, we aim to leverage the diversity and complementarity of heterogeneous support information and use CLIP to generate auxiliary class-specific text and visual features to enhance 3D prototypes, which not only preserves the geometric structural information of point clouds but also enriches with richer semantic and visual information. To achieve this, we propose a simple yet effective framework that integrates heterogeneous prototypes from point clouds, label texts, and projected 2D depth maps, as shown in Fig. 1 (c). To the best of our knowledge, this is the first approach to fully leverage the diversity and complementarity of heterogeneous information hidden in the support set in FS-3DSeg. For this approach to be beneficial, effectively fusing different prototypes into refined prototypes is crucial. However, there exist huge heterogeneous gaps between heterogeneous prototypes, posing challenges to coordinating or unifying multimodal information, potentially impacting segmentation performance.

Ultimately, in this paper, we propose a **R**elation **C**onsistency-guided **H**eterogeneous **P**rototype learning framework (RCHP) for FS-3DSeg. Specifically, we first design a heterogeneous prototype enhancement (HPE) module that generates several heterogeneous class-wise prototypes (including 3D, text, and 2D prototypes), followed by a simple yet effective heterogeneous prototype fusion scheme to derive refined prototypes. Benefiting from the HPE module, the refined prototypes encompass richer semantics and visual cues, and can better associate with the query points. For example, given the support set and query set containing "table legs" and "table surface" respectively, the 3D prototype and refined prototype can represent "table legs" and "table", the latter can better match with the query. Additionally, the refined prototypes preserve more reliable inter-prototype structural relations (IPR), such as distance-wise IPR between a pair of prototypes and angle-wise IPR among a triplet of prototypes. Based on this, we further propose a heterogeneous relation consistency (HRC) loss to transfer more reliable inter-class relations (*i.e.*, IPR) from refined prototypes to heterogeneous prototypes, guiding the learning process of the model. By collaborating the HRC loss with the HPE module, we enable mutual promotion and bidirectional optimization between the refined prototypes and heterogeneous ones, thereby effectively reducing heterogeneous gaps and improving the performance of the model. We conduct extensive experiments on five datasets, spanning both indoor and outdoor scenes. Experimental results show that our method achieves state-of-the-art performance and demonstrates strong generalization.

Our main contributions can be summarized as follows:

- To fully exploit the limited support set, we propose a simple yet effective FS-3DSeg framework, named RCHP, that simultaneously learns and fuses heterogeneous prototypes, including 3D, 2D, and text. To the best of our knowledge, we are the first to unify prototype-based and CLIP-based methods into a unified framework to address the FS-3DSeg task.
- We propose a heterogeneous prototype enhancement (HPE) module, which generates and fuses heterogeneous class-wise prototypes to enhance prototype representation ability.
- To bridge the heterogeneous gap, we introduce a heterogeneous relation consistency (HRC) loss to facilitate the mutual enhancement of refined and heterogeneous prototypes.

II. Related Work

A. 3D Point Cloud Semantic Segmentation

3D point cloud semantic segmentation (3DSeg) aims to assign semantic labels to 3D point clouds. Supervised 3DSeg approaches can be broadly categorized into two groups: voxel-based [18], [19] and point-based methods [4]–[7], with the latter gaining more attention for their simplicity and effectiveness. DGCNN [5] introduced the EdgeConv module to capture local structures. Recently, [6], [7] design self-attention-like networks to model long-range contexts from distant neighbors. Several recent approaches [20]–[24] have significantly advanced 3D feature learning and scene understanding. Some methods focus on efficient module designs [20] or effective feature learning strategies [22]. However, these methods require large amounts of labeled points. To reduce reliance on large-scale labeled points, other methods explore data-efficient 3D learning strategies, such as self-supervised learning [21], unsupervised learning [23] and weakly supervised learning [24]. However, these methods are inconvenient to segment novel categories. In this paper, following mainstream FS-3DSeg methods [8], [10], [14], we utilize DGCNN as the point encoder and extend its capability to segment novel classes.

B. Few-shot learning

Few-shot learning methods, aiming to generalize a classifier to new classes with very few labeled samples, comprise three groups: augmentation-based methods enhance data diversity using augmentation techniques [25], [26] or extra data [27], [28]; optimization-based methods [29]–[31] learn transferable

knowledge through meta-learning; and metric-based methods [32]–[35] measure distances between query and support samples to predict the class. Specifically, the prototype learning frameworks [32]–[35] which learn semantic prototypes from the support set have achieved effective results. We follow this line to address a more complicated FS-3DSeg problem.

C. Few-shot 3D Point Cloud Segmentation

Few-shot 3D point cloud semantic segmentation (FS-3DSeg) aims to train a model on base classes and effectively segment novel 3D classes using only a few annotated point cloud samples. Current FS-3DSeg methods [8]–[14] mostly follow the metric-based prototype learning paradigms, which represent each class by prototypes from support set and segment query points based on their similarity to prototypes. Specifically, AttMPTI [8] proposed the first FS-3DSeg method, which adopts an attention-aware multi-prototype transductive inference framework. BFG [9] embedded global perception into local point features and their prototypes in a mutually enhancing fashion. Furthermore, [10]–[12], [14] enhanced the performance by reducing contextual gaps between support prototypes and query features via cross attention [36]. In addition to prototype learning, SCAT [15] applied transformer blocks [36] to explore class-specific relations between all query and support features without using pooling operations. However, these methods focus solely on analyzing 3D support data and overlook the inherent heterogeneous support information, such as 2D depth maps and label names, resulting in low-quality prototypes. Our approach simultaneously leverages the diversity and complementarity of heterogeneous support information to enhance FS-3DSeg.

D. Large Multi-modal Models

Large multi-modal models (LMMs) have gained significant attention for their ability to process and integrate information across multiple modalities, such as text, images, videos, and audio. These models are repositories of extensive knowledge for pre-training on large-scale multi-modal datasets. Notable examples include CLIP [17] and BLIP [37], which can associate visual and text information and perform well in tasks such as image-text matching and visual question answering. LLaVA [38], GPT-4 Vision [39] and MiniGPT [40] further combined language models with visual understanding capabilities. More recently, there also emerge several 3D-LMMs (*e.g.*, Point-LLM [41], MiniGPT-3D [42], Uni3D-LLM [43], PointCLIP [16] and PointCLIP V2 [44]) which associate point clouds with texts and other modalities. Nevertheless, current 3D-LMMs are not tailored for segmenting novel classes in meta-learning-based FS-3DSeg. Our approach aims to leverage the capabilities of CLIP to solve the FS-3DSeg problem by leveraging its rich multimodal features.

III. Method

A. Problem Formulation and Overview

1) Problem Formulation: According to the few-shot learning paradigm [8], [45], we adopt the episode paradigm to train and test our model. Each episode instantiates an *'N-way K-shot'* segmentation task. The data used by each task contains a support set $S = \{(\mathbf{P}_s^{n,k}, \mathbf{M}_s^{n,k})_{k=1}^{K}\}_{n=1}^{N}$ and a query set $Q = \{(\mathbf{P}_q^i, \mathbf{M}_q^i)\}_{i=1}^{T}$, where N, K and T denote the number of classes, the number of support point clouds for each class, and the number of query point clouds. $\mathbf{P}_s^{n,k}$ and $\mathbf{P}_q^i$ denote the support and query point cloud, each contains M points. $\mathbf{M}_s^{n,k} \in \{0,1\}^{M\times 1}$ denotes support ground-truth (GT) binary mask, and $\mathbf{M}_q^i \in \{0,...,N\}^{M\times 1}$ denote the query GT mask. Beyond the original S, we reformulate a heterogeneous support set $\tilde{S}$ by integrating inherent heterogeneous information of S:

$$\tilde{S} = \left\{ \left(\mathbf{P}_s^{n,k}, \mathbf{M}_s^{n,k}, (\mathbf{D}_s^{n,k,v}, \hat{\mathbf{D}}_s^{n,k,v})_{v=1}^{V}\right)_{k=1}^{K}, \mathbf{W}_s^n \right\}_{n=1}^{N}, \quad (1)$$

where $\mathbf{W}_s^n$ denotes semantic text (label names) from support GT mask, $\mathbf{D}_s^{n,k,v} \in \mathbb{R}^{H\times W}$ and $\hat{\mathbf{D}}_s^{n,k,v} \in \mathbb{R}^{H\times W}$ denote the depth map projected from $\mathbf{P}_s^{n,k}$ for class n and background, V denotes the number of projection views.

In this paper, our goal is to learn a model F_θ using $\tilde{S}$ to predict the mask $\hat{\mathbf{M}}_q^i$ for $\mathbf{P}_q^i$. The optimization target is to minimize label prediction errors through a segmentation loss $\mathcal{L}_{SEG}$, *i.e.*, a standard cross-entropy loss, expressed as:

$$\mathcal{L}_{SEG} = \sum_{i=1}^{T} \mathcal{L}_{CE}(\hat{\mathbf{M}}_q^i, \mathbf{M}_q^i). \quad (2)$$

2) Overview: Fig. 2 illustrates the architecture of the RCHP framework, comprising two key components: an HPE module and an HRC loss. Specifically, HPE contains support and query flows. In the support flow, HPE adopts a shared point encoder to extract query point features $\mathbf{F}_q^i$ and support point features $\mathbf{F}_s^{n,k}$ for Q and S, and utilize CLIP to extract visual and text features for support set. Then HPE generates several heterogeneous prototypes (3D, text, and 2D) from $\tilde{S}$, followed by a fusion scheme to obtain refined prototypes $\ddot{\mathbb{P}}$. Additionally, during training, an HRC loss is employed to mitigate the heterogeneous gap. In the query flow, similarity maps between $\mathbf{F}_q^i$ and $\ddot{\mathbb{P}}$ are calculated using cosine distance. Each point cloud in the query set is then assigned the label of the most similar prototype. During testing, we use the HPG module to extract point features from both the support and query sets, and generate heterogeneous prototypes from the support set. Then we fuse these prototypes by the HPF module to obtain refined prototypes. The final segmentation results are predicted by measuring the distance between query features and the refined prototypes.

Subsequently, we provide a detailed description of the HPE module and HRC loss as below.

B. Heterogeneous Prototype Enhancement

1) Heterogeneous Prototype Generation: To fully exploit the potential of S, inspired by [16], [46], [47] which incorporates multi-modal heterogeneous features to enhance features, we reformulate S as $\tilde{S}$ by incorporating the inherent heterogeneous information of S. Then several class-wise heterogeneous prototypes can be generated: 3D prototypes, text prototypes, and 2D prototypes.

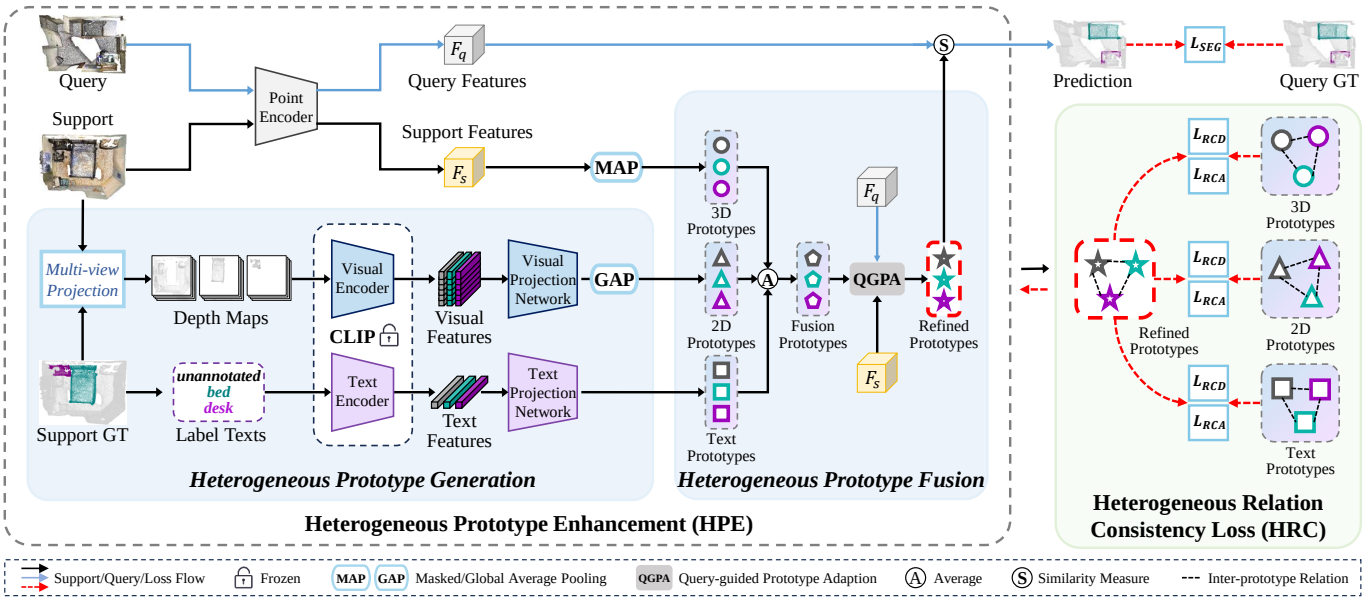


Fig. 2. Architecture overview of our proposed method. The support flow is responsible for Heterogeneous Prototype Enhancement (HPE), including heterogeneous prototype generation and fusion processes, where 3D, 2D and text prototypes are integrated together to obtain refined prototypes. Heterogeneous relation consistency loss further enhances the prototype enhancement process by distilling relations between prototypes. The query flow calculates the distance between query point features and refined prototypes to predict segmentation results. The figure illustrates a 2-way 1-shot setting.

(1) 3D Prototypes. Given support point cloud $\mathbf{P}_s^{n,k}$ and query point cloud $\mathbf{P}_q^i$, following common practice [8], [10], we utilize a shared point encoder to extract their per-point features, represented as $\mathbf{F}_s^{n,k} \in \mathbb{R}^{M\times d}$ and $\mathbf{F}_q^i \in \mathbb{R}^{M\times d}$ respectively, where d denotes the feature dimension. Then we apply masked average pooling (MAP) [10] to generate class-wise 3D prototypes $\mathbb{P}_{3D} = \{\mathcal{P}_{3D}^n\}_{n=0}^N \in \mathbb{R}^{(N+1)\times d}$ from support features, including a background prototype and N foreground prototypes.

(2) Text Prototypes. Due to S containing limited semantic information, vanilla 3D prototypes $\mathbb{P}$ struggle to associate with Q. To cope with the lack of semantics, inspired by CLIP's [17] extensive training on diverse visual-text data and capturing rich semantic information, we aim to leverage a frozen CLIP text encoder to generate text prototypes. Specifically, given label names $W_s = \{\mathbf{W}_s^n\}_{n=0}^N$ for both background and N sampled classes, we place each of them to the class token position of a predefined 3D-specific template: "*point cloud of a [CLASS].*", utilize a frozen CLIP text encoder E_T to extract text features, and employ a trainable semantic projection network F_T following [10] to generate text prototypes $\mathbb{P}_{text} = \{\mathcal{P}_{text}^n\}_{n=0}^N \in \mathbb{R}^{(N+1)\times d}$, formulated as:

$$\mathcal{P}_{text}^n = F_T(E_T(\mathbf{W}_s^n)), \quad n \in \{0, ..., N\}. \tag{3}$$

(3) 2D Prototypes. Compared to sparse and unordered point clouds that provide incomplete geometric structures, projecting them into 2D space provides clearer boundary details, more defined shapes, and richer visual context, highlighting edges and contours less noticeable in 3D. Inspired by this, we project the original $\mathbf{P}_s^{n,k}$ from V different views to generate class-specific multi-view 2D depth maps,formulated as:

$$\begin{cases} \mathbf{D}_s^{n,k,v} = Proj(\mathbf{P}_s^{n,k} \odot \mathbf{M}_s^{n,k}, v), & v \in \{1, ..., V\} \\ \hat{\mathbf{D}}_s^{n,k,v} = Proj(\mathbf{P}_s^{n,k} \odot \neg\mathbf{M}_s^{n,k}, v), & v \in \{1, ..., V\} \end{cases}, \tag{4}$$

where $\mathbf{D}_s^{n,k,v}, \hat{\mathbf{D}}_s^{n,k,v} \in \mathbb{R}^{H\times W}$ represent depth maps with the size of $H \times W$ for class n and background. Next, we use a frozen CLIP visual encoder E_V to extract a visual feature from each depth map. After globally averaging these visual features per class, a visual projection network F_V generates 2D prototypes $\mathbb{P}_{2D} = \{\mathcal{P}_{2D}^n\}_{n=0}^N \in \mathbb{R}^{(N+1)\times d}$, formulated as:

$$\mathcal{P}_{2D}^n = \begin{cases} F_V\big(\frac{1}{KV}\sum_{k=1}^{K}\sum_{v=1}^{V} E_V(\mathbf{D}_s^{n,k,v})\big), n \in \{1, ..., N\} \\ F_V\big(\frac{1}{NKV}\sum_{c=1}^{N}\sum_{k=1}^{K}\sum_{v=1}^{V} E_V(\hat{\mathbf{D}}_s^{c,k,v})\big), \quad n = 0 \end{cases}. \tag{5}$$

2) Heterogeneous Prototype Fusion: To incorporate text knowledge and visual cues into vanilla prototypes $\mathbb{P}$, we introduce a simple but well-performed heterogeneous prototype fusion scheme. For simplicity and scalability, we conduct a flexible feature-level averaging operation to produce the fused prototypes $\dot{\mathbb{P}} = \{\dot{\mathcal{P}}^n\}_{n=0}^N \in \mathbb{R}^{(N+1)\times d}$, where each is calculated by:

$$\dot{\mathcal{P}}^n = \frac{1}{3}(\mathcal{P}_{3D}^n + \mathcal{P}_{text}^n + \mathcal{P}_{2D}^n), \quad n \in \{0, ..., N\}. \tag{6}$$

To mitigate the feature channel distribution gap [10], [48] between prototypes and query features, QGPA [10] proposed a Query-Guided Prototype Adaption (QGPA) module, which utilizes cross-attention [36] to enhance the prototype by query-support feature interaction. Following this, we adopt QGPA to generate a set of refined prototypes $\ddot{\mathbb{P}}^i = \{\ddot{\mathcal{P}}^{n,i}\}_{n=0}^N \in \mathbb{R}^{(N+1)\times d}$ for each query point cloud $\mathbf{P}_q^i$, where each formulated by:

$$\ddot{\mathcal{P}}^{n,i} = \begin{cases} QGPA\big(\mathbf{F}_q^i, (\mathbf{F}_s^{n,k})_{k=1}^K, \dot{\mathcal{P}}^n\big), & n \in \{1, ..., N\} \\ QGPA\big(\mathbf{F}_q^i, \{(\mathbf{F}_s^{c,k})_{k=1}^K\}_{c=1}^N, \dot{\mathcal{P}}^n\big), & n = 0 \end{cases}, \tag{7}$$

where $i \in \{1, ..., T\}$. After that, we can obtain T sets of refined prototypes $\ddot{\mathbb{P}} = \{\ddot{\mathbb{P}}^0, ..., \ddot{\mathbb{P}}^T\}$ for T query point clouds.

Finally, we calculate similarity scores using cosine distance between $\mathbf{F}_q^i$ and its $\ddot{\mathbb{P}}^i$. Each point is then assigned the label with the most similar prototype to generate $\hat{\mathbf{M}}_q^i$.

C. Heterogeneous Relation Consistency Loss

To further reduce heterogeneous gaps and enhance prototype refinement, inspired by relational knowledge distillation (RKD) loss [49], which unidirectionally transfers structural relations between features from teacher to student in the same image modality, we introduce a Heterogeneous Relation Consistency (HRC) loss. HRC loss facilitates the transfer of more reliable inter-class relations (*i.e.*, inter-prototype relations, IPR) from refined prototypes to heterogeneous prototypes, as shown in Fig. 3. Specifically, we treat the refined prototype as the teacher and the heterogeneous prototypes from different modalities (*e.g.*, 3D, 2D, text) as student prototypes. The HRC loss distills IPRs from the teacher (refined prototype) to each student (3D, 2D and text prototypes), aligning heterogeneous relationships while maintaining each modality's unique characteristics.

Besides, unlike RKD, as shown in Fig. 3, our HRC loss first decomposes each prototype feature into two complementary subspaces based on the channel dimension before transferring IPR, enabling more precise capturing of fine-grained feature relations across multiple subspaces, which is particularly important for complex 3D data. This reduces redundancy and improves relational transfer accuracy, allowing the model to focus on modality-specific information while ensuring precise prototype relation transfer. Moreover, through the interaction between the HPE module and HRC loss during training, our model enables mutually beneficial and bi-directional optimization between heterogeneous and refined prototypes. This process narrows the heterogeneous gap and facilitates bidirectional learning, ultimately improving the refinement of prototypes across different modalities and enabling more effective transfer of relational knowledge.

Specifically, for each prototype set $\mathbb{P}$, we divide it into two subsets according to the channel, *i.e.*, $\mathbb{P} = \mathbb{P}_1 \copyright \mathbb{P}_2$, where each prototype $\mathcal{P} = \mathcal{P}_1 \copyright \mathcal{P}_2$, $\copyright$ denote the channel-wise concatenation operation. Then HRC loss constrains the consistency of distance-wise IPRs and angle-wise IPRs by two loss functions, *i.e.*, distance-wise relation consistency loss and angle-wise relation consistency loss.

1) Distance-wise Relation Consistency Loss: Distance-wise IPR measures the second-order relation between a pair of prototypes $(\mathcal{P}^a, \mathcal{P}^b)$ within the same prototype set $\mathbb{P}$. It can be calculated by the Euclidean distance ψ_D:

$$\begin{aligned} \psi_D(\mathcal{P}^a, \mathcal{P}^b) &= \sum_{c=1}^{2} \frac{1}{\mu_c} \|\mathcal{P}_c^a - \mathcal{P}_c^b\|_2 \\ where \quad \mu_c &= \frac{1}{|\mathbb{P}_c^2|} \sum_{(\mathcal{P}_c^a, \mathcal{P}_c^b) \in \mathbb{P}_c^2} \|\mathcal{P}_c^a - \mathcal{P}_c^b\|_2 \end{aligned}. \tag{8}$$

Using the distance-wise IPR measured in the final refined prototype set $\ddot{\mathbb{P}}$ and each heterogeneous prototype set $\mathbb{P}$, a distance-wise relation consistency loss is defined as:

$$\mathcal{L}_{RCD}(\ddot{\mathbb{P}}, \mathbb{P}) = \sum_{i,a,b} \mathcal{L}_\delta \big(\psi_D(\ddot{\mathcal{P}}^{a,i}, \ddot{\mathcal{P}}^{b,i}), \psi_D(\mathcal{P}^a, \mathcal{P}^b)\big), \tag{9}$$

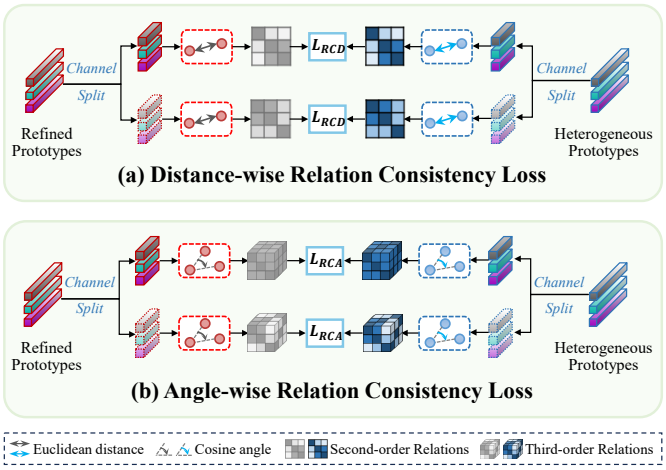


Fig. 3. Our Heterogeneous Relation Consistency (HRC) Loss consists of two parts, *i.e.*, distance-wise relation consistency loss and angle-wise relation consistency loss.

where $a, b \in \{0, ..., N\}$, $\mathbb{P} \in \{\mathbb{P}_{3D}, \mathbb{P}_{text}, \mathbb{P}_{2D}\}$, $\mathcal{L}_\delta$ is the Huber loss.

2) Angle-wise Relation Consistency Loss: Angle-wise IPR measures the third-order relation of the triplet $(\mathcal{P}^a, \mathcal{P}^b, \mathcal{P}^k)$ in the same prototype set, calculated by cosine angle ψ_A:

$$\psi_A(\mathcal{P}^a, \mathcal{P}^b, \mathcal{P}^k) = \sum_{c=1}^{2} \Big\langle \frac{\mathcal{P}_c^a - \mathcal{P}_c^b}{\|\mathcal{P}_c^a - \mathcal{P}_c^b\|_2}, \frac{\mathcal{P}_c^k - \mathcal{P}_c^b}{\|\mathcal{P}_c^k - \mathcal{P}_c^b\|_2} \Big\rangle. \tag{10}$$

Using the angle-wise IPR measured in both the final refined prototype set $\ddot{\mathbb{P}}$ and each heterogeneous prototype set $\mathbb{P}$, an angle-wise relation consistency loss is defined as:

$$\mathcal{L}_{RCA}(\ddot{\mathbb{P}}, \mathbb{P}) = \sum_{i,a,b,c} \mathcal{L}_\delta \big(\psi_A(\ddot{\mathcal{P}}^{a,i}, \ddot{\mathcal{P}}^{b,i}, \ddot{\mathcal{P}}^{c,i}), \psi_A(\mathcal{P}^a, \mathcal{P}^b, \mathcal{P}^c)\big), \tag{11}$$

where $a, b, c \in \{0, ..., N\}$, $\mathbb{P} \in \{\mathbb{P}_{3D}, \mathbb{P}_{text}, \mathbb{P}_{2D}\}$.

Total loss. We combine $\mathcal{L}_{RCD}$ and $\mathcal{L}_{RCA}$ with a balancing weight γ to form the total relation consistency loss $\mathcal{L}_{RC}$, formulated as:

$$\mathcal{L}_{RC} = \sum_{\mathbb{P} \in \{\mathbb{P}_{3D}, \mathbb{P}_{text}, \mathbb{P}_{2D}\}} \big(\mathcal{L}_{RCD}(\ddot{\mathbb{P}}, \mathbb{P}) + \gamma \times \mathcal{L}_{RCA}(\ddot{\mathbb{P}}, \mathbb{P})\big). \tag{12}$$

During training, the overall loss is a weighted combination of a standard cross-entropy loss $\mathcal{L}_{SEG}$ and the proposed HRC loss $\mathcal{L}_{RC}$ with a balancing weight λ, represented as:

$$\mathcal{L}_{total} = \mathcal{L}_{SEG} + \lambda \times \mathcal{L}_{RC}. \tag{13}$$

IV. Experiments

A. Datasets & Evaluation Metrics

Dataset. In accordance with AttMPTI [8], we conduct experiments on two indoor datasets: *i.e.*, **S3DIS** [50] and **ScanNet** [51]. To further verify the generalization of the model, we conduct additional experiments on three more datasets, *i.e.*, 1) two indoor datasets (**SceneNN** [52], [53] and **NYU Depth V2** [54]) which pose greater challenges due to increased class diversity, significant class imbalance, occlusion and the presence of small object classes; 2) one outdoor dataset (**Semantic3D [55]**) which contains unstructured objects and

TABLE I
TEST CLASS NAMES FOR EACH SPLIT OF DIFFERENT DATASETS.

	S_0	S_1
S3DIS	(6 classes) beam, board, bookcase, ceiling, chair, column	(6 classes) door, floor, sofa, table, wall, window
ScanNet	(10 classes) bathtub, bed, bookshelf, cabinet, chair, counter, curtain, desk, door, floor	(10 classes) otherfurniture, picture, refrigerator, show curtain, sink, sofa, table, toilet, wall, window
SceneNN	(16 classes) bed, bookshelf, cabinet, chair, counter, curtain, desk, door, floor, pillow, dresser, box, television, lamp, mirror, whiteboard	(17 classes) picture, fridge, sink, sofa, table, wall, window, structure, floor mat, clothes, books, bag, night stand, prop, paper, towel, shelves
	(7 classes) blinds, ceiling, shower curtain, person, toilet, bathtub, furniture (These classes are filtered as *background* because no corresponding annotated point clouds are provided.)	
NYU Depth V2	(20 classes) wall, cabinet, chair, door, bookshelf, counter, desk, curtain, pillow, floor mat, ceiling, refrigerator, towel, box, person, toilet, lamp, bag, otherstructure	(20 classes) floor, bed, sofa, table, window, picture, blinds, shelves, dresser, mirror, clothes, books, television, paper, shower curtain, whiteboard, night stand, sink, bathtub, otherfurniture, otherprop
Semantic3D	(4 classes) buildings, cars, hard scape, high vegetation	(4 classes) low vegetation, man-made terrain, natural terrain, scanning artefacts

complex outdoor scenes. **(1) S3DIS [50]** collects 3D-RGB point clouds from 272 rooms across six indoor scenes. Each point is labeled with 12 semantic categories and the clutter. **(2) ScanNet** [51] contains 1,513 point clouds derived from 707 different indoor scenes. Every point, excluding unannotated areas, is assigned one of 20 semantic classes. **(3) SceneNN [52]** consists of more than 100 indoor scenes. For point cloud semantic segmentation, [53] annotated 76 scenes from the SceneNN dataset with 40 categories defined by the NYU Depth v2 dataset [54]. **(4) NYU Depth V2 [54]** consists of 1,449 RGBD images collected from various commercial and residential buildings in three US cities. The dataset contains 35,064 distinct objects across 894 classes, which are mapped to 41 semantic classes, including 40 foreground classes and one background class. **(5) Semantic3D [55]** contains over 4 billion points, covering diverse outdoor urban scenes. It includes 8 categories and 1 unannotated category.

Data preprocess. Since the original rooms contain an excessive number of points for direct processing, we divide each room into several smaller blocks. For S3DIS and ScanNet datasets, we follow the pre-processing strategy in [8], [10] to divide the rooms into blocks using a non-overlapping sliding window of 1m×1m on the xy plane, yielding 7,547 blocks for S3DIS and 36,350 blocks for ScanNet, respectively. For SceneNN dataset, we follow [53] to use a 2×2 sqm. window with a stride of 0.2 meters and a height of 2 meters to scan the floor area, resulting in a total of 48,714 blocks. For NYU Depth V2 dataset, following [1] we convert each RGBD image to point clouds with a scale of 1, and then divide each point cloud into blocks with a window size of 0.5 and stride of 0.5, resulting in a total of 51,704 blocks. For Semantic3D dataset, we scan each scene with a 5×5 sqm windows and a stride of 5 meters, resulting in a total of 4,825 blocks. During training/testing, we randomly sample M = 2, 048 points from each block. For S3DIS, ScanNet and NYU Depth V2 datasets, each point is represented by a 9D vector, including XYZ, RGB, and normalized spatial coordinates. For SceneNN dataset, each point is represented by a 15D vector, including XYZ, 9 attributes, and normalized spatial coordinates. For Semantic3D dataset, each point is represented by a 10D vector, including XYZ, intensity, RGB and normalized spatial coordinates. For meta-training/testing, following [8], semantic classes are evenly split into two non-overlapping subsets, denoted as S^0 and S^1, as shown in Table I. When training our model on one fold (*e.g.*, S^0), we test the model on another fold (*e.g.*, S^1). Vice versa for cross-validation.

Evaluation Metrics. Following conventions in the 3DSeg community, we report the mean Intersection-over-Union (mIoU) across all test classes.

B. Implementation Details

Framework details. We select QGPA [10] as our FS-3DSeg baseline equipped with a segmentation loss, a self-reconstruction loss, and an alignment loss. Specifically, we utilize DGCNN [5] (with SAN) as 3D point encoder, a pre-trained frozen CLIP [17] (*CLIP rn50* with feature dimension of 1,024) as text encoder and visual encoder. Note that textual features can be pre-computed and stored offline, avoiding redundant computation using CLIP text encoder online training/testing. In contrast, the CLIP visual encoder is essential for extracting visual features online, as it processes dynamically generated multi-view 2D depth maps from 3D point clouds. For 3D to 2D projection, the total projection view V is set to 6, including {*Front, Right, Behind, Left, Top, Down*}. The size of projected images are set to 128×128 and later resized to 224×224 to serve as input to the CLIP visual encoder. Both the semantic projection network and visual projection network adopt a *Linear+LeakyReLU+Dropout+Linear* architecture, with input/hidden/output dimensions of 1024/320/320, respectively. For our proposed HRC loss in Eq. 12, the weight γ for balancing L_{RCD} and L_{RCA} is set to 2 following [49]. For the total loss in Eq. 13, the weight λ for balancing segmentation loss and HRC loss is set to 1.

Training details. Our method is implemented by Pytorch and runs on an NVIDIA RTX A4000 GPU. Before few-shot learning, following [8], [10], we pre-train the point encoder on the training set for 100 epochs with a batch size of 32, using the Adam optimizer with a learning rate of 0.001, a weight decay of 0.0001, and a decay step of 50. In the few-shot training process, we randomly sample 40,000 training

[1]https://github.com/parkie0517/NYUDepthV2_PointCloud_Converter

TABLE II
RESULTS ON S3DIS DATASET USING MEAN-IOU METRIC (%). S^i DENOTES THE SPLIT i IS USED FOR TESTING.

Methods	2-way						3-way					
	1-shot			5-shot			1-shot			5-shot		
	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean
ProtoNet [8]	48.39	49.98	49.19	57.34	63.22	60.28	40.81	45.07	42.94	49.05	53.42	51.24
AttMPTI [8]	53.77	55.94	54.86	61.67	67.02	64.35	45.18	49.27	47.23	54.92	56.79	55.86
BFG [9]	55.60	55.98	55.79	63.71	66.62	65.17	46.18	48.36	47.27	55.05	57.80	56.43
SCAT [15]	54.92	56.74	55.83	64.24	69.03	66.63	-	-	-	-	-	-
QGPNet [12]	56.30	57.62	56.96	65.34	69.01	67.17	47.00	50.12	48.56	55.80	58.54	57.17
2CBR [13]	55.89	61.99	58.94	63.55	67.51	65.53	46.51	53.91	50.21	55.51	58.07	56.79
QGE [11]	58.85	60.29	59.57	66.56	79.46	73.01	-	-	-	-	-	-
QGPA [10]	59.45	66.08	62.76	65.40	70.30	67.85	48.99	56.57	52.78	61.27	60.81	61.04
DPA [14]	66.08	74.30	70.19	71.10	77.03	74.07	50.67	59.53	55.10	64.52	63.34	63.93
RCHP	**67.50**	**74.43**	**70.97**	**72.30**	**77.93**	**75.12**	**61.01**	**68.62**	**64.82**	**64.56**	**66.46**	**65.51**

TABLE III
RESULTS ON SCANNET DATASET USING MEAN-IOU METRIC (%). S^i DENOTES THE SPLIT i IS USED FOR TESTING.

Methods	2-way						3-way					
	1-shot			5-shot			1-shot			5-shot		
	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean
ProtoNet [8]	33.92	30.95	32.44	45.34	42.01	43.68	28.47	26.13	27.30	37.36	34.98	36.17
AttMPTI [8]	42.55	40.83	41.69	54.00	50.32	52.16	35.23	30.72	32.98	46.74	40.80	43.77
BFG [9]	42.15	40.52	41.34	51.23	49.39	50.31	34.12	31.98	33.05	46.25	41.38	43.82
SCAT [15]	45.24	45.90	45.57	55.38	57.11	56.24	-	-	-	-	-	-
QGPNet [12]	44.63	42.18	43.40	54.75	51.81	53.28	37.86	34.50	36.18	47.45	42.74	45.09
2CBR [13]	50.73	47.66	49.20	52.35	47.14	49.75	47.00	46.36	46.68	45.06	39.47	42.27
QGE [11]	43.10	46.79	44.95	51.91	57.21	54.56	-	-	-	-	-	-
QGPA [10]	57.08	55.94	56.51	64.55	59.64	62.10	55.27	55.60	55.44	59.02	53.16	56.09
DPA [14]	62.75	63.04	62.90	67.19	64.62	65.91	61.97	61.72	61.85	66.13	64.67	65.40
RCHP	**71.56**	**70.33**	**70.95**	**76.36**	**73.86**	**75.11**	**66.02**	**66.25**	**66.14**	**72.11**	**72.07**	**72.09**

TABLE IV
RESULTS ON SCENENN DATASET USING MEAN-IOU METRIC (%). S^i DENOTES THE SPLIT i IS USED FOR TESTING. $*$ INDICATES THAT THE EXPERIMENTAL RESULTS WERE REPRODUCED BY US.

Methods	2-way						3-way					
	1-shot			5-shot			1-shot			5-shot		
	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean
ProtoNet* [56]	21.90	21.38	21.64	29.81	26.83	28.32	17.23	16.13	16.68	23.47	20.24	21.86
QGPA* [10]	23.85	24.18	24.02	42.11	43.82	42.97	30.96	30.61	30.79	36.79	43.60	40.20
RCHP	**32.91**	**32.29**	**32.60**	**53.03**	**48.93**	**50.98**	**32.82**	**39.44**	**36.13**	**47.62**	**45.32**	**46.47**

TABLE V
RESULTS ON NYU DEPTH V2 DATASET USING MEAN-IOU METRIC (%). S^i DENOTES THE SPLIT i IS USED FOR TESTING. $*$ INDICATES THAT THE EXPERIMENTAL RESULTS WERE REPRODUCED BY US.

Methods	2-way						3-way					
	1-shot			5-shot			1-shot			5-shot		
	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean
ProtoNet* [56]	23.34	25.99	24.67	51.26	34.16	42.71	18.02	19.71	18.87	24.38	26.54	25.46
QGPA* [10]	27.72	30.09	28.91	64.28	69.10	66.69	32.23	48.33	40.28	60.12	62.07	61.10
RCHP	**33.49**	**34.51**	**34.00**	**67.71**	**77.49**	**72.60**	**53.19**	**63.42**	**58.31**	**66.14**	**64.43**	**65.29**

TABLE VI
RESULTS ON SEMANTIC3D DATASET USING MEAN-IOU METRIC (%). S^i DENOTES THE SPLIT i IS USED FOR TESTING. $*$ INDICATES THAT THE EXPERIMENTAL RESULTS WERE REPRODUCED BY US.

Methods	2-way						3-way					
	1-shot			5-shot			1-shot			5-shot		
	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean	S^0	S^1	mean
ProtoNet* [56]	34.56	39.90	37.23	45.52	39.91	42.72	27.25	34.17	30.71	37.45	40.32	38.89
PAP* [10]	37.09	41.60	39.35	51.41	46.24	48.83	41.61	38.92	40.27	47.26	42.45	44.86
RCHP	**45.17**	**44.35**	**44.76**	**59.82**	**47.61**	**53.72**	**45.22**	**42.55**	**43.89**	**48.41**	**44.13**	**46.27**

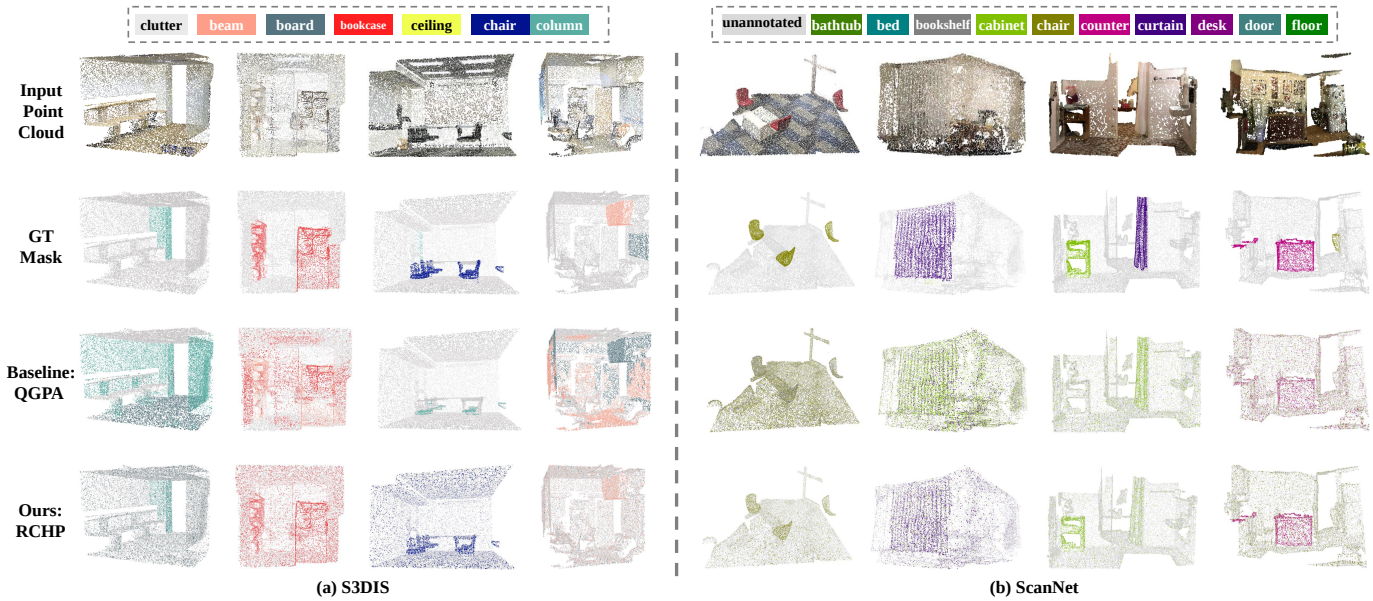


Fig. 4. Qualitative results of our method in '*2-way 1-shot* S^0' point cloud semantic segmentation on the S3DIS and ScanNet datasets in comparison to the GT mask and QGPA [10]. Best viewed in color.

episodes from the training set to train our model using the Adam optimizer with a learning rate of 0.001, a decay step of 5000, and a decay ratio of 0.5. For the query set, the number of query point clouds T is set to 1 for each class. During testing, we sample 100 episodes from the S3DIS, ScanNet and Semantic3D datasets for each feasible class combination from the testing set and report the average results across all episodes. Note that for datasets like SceneNN and NYU Depth V2, which contain much more classes, we test more efficiently by setting the number of test episodes to 100 for the SceneNN dataset under the 2-way setting and to 20 episodes across all other settings.

C. Comparison With State-of-the-Art Methods

Results on S3DIS. Table II shows the experimental results of our method compared with state-of-the-art (SOTA) FS-3DSeg methods on the S3IDS dataset. Our method substantially improves the baseline QGPA [10] by 3.29%∼12.05% across different split settings, marking a significant improvement on FS-3DSeg. Furthermore, our approach consistently outperforms all SOTA methods, particularly surpassing DPA [14] by up to 10.34% under '3-way 1-shot S^0' setting, showcasing our effectiveness and superiority. We analyze that, unlike 3D prototypes that only capture limited geometric information, our HPE module and HRC loss enhance FS-3DSeg by integrating rich semantic information from texts and visual cues from depth maps into 3D prototypes, effectively addressing the issue of limited support information.

Results on ScanNet. Table III shows the results of our method compared with SOTA FS-3DSeg methods on ScanNet. We were very pleased to observe that our method significantly outperforms baseline QGPA [10] by a large margin of 10.65%∼18.91%, marking a significant enhancement for the challenging FS-3DSeg problem. Especially for the *'2-way 1-shot'* setting, RCHP improves the baseline by 14.44% (70.95% of ours *vs.* 56.51% of QGPA), respectively. For *'3-way 5-shot'* setting, our RCHP even surpass SOTA by 16.00% (72.09% *vs.* 56.09%). Besides, we significantly outperform the SOTA methods, especially surpassing DPA [14] by large margins of 4.05%∼9.24%. Notably, our model shows even greater improvement on ScanNet compared to S3DIS because ScanNet's wider range of classes and more complex scenarios challenge 3D prototype quality, our HPE module and HRC loss effectively improve prototype quality by coping with the lack of inherent support information, thus improving performance.

Generalization validation. We reproduce our method and two SOTA baselines, ProtoNet [56] and QGPA [10], on SceneNN, NYU Depth V2 and Semantic3D datasets. Results in Tables IV, V and VI show that our method consistently outperforms ProtoNet and QGPA across all settings, demonstrating its robustness and effectiveness in both indoor and outdoor segmentation datasets. Besides, we find that all models perform lower on SceneNN compared to other datasets, likely due to differences in input feature dimensions. While S3DIS, ScanNet, and NYU Depth V2 use 9D features (XYZ, RGB, normalized coordinates) that effectively capture essential semantics, SceneNN's 15D features include additional attributes that may introduce noise or redundancy, making it more challenging for the model to learn effectively, thereby affecting its overall performance.

Comparison with CLIP-based Method. We reproduce PointCLIP [16] in our FS-3DSeg task by training the model on the few-shot support set and testing on the query set. To segment point clouds by PointCLIP, we project each point cloud into depth maps, extract pixel-wise feature maps, and match pixel features with text features. As shown in Table VII, PointCLIP performs much worse than ours by 13.69%∼21.75% and 30.81%∼35.98% on S3DIS and ScanNet, respectively. This is because the sparsity and irregularity of the point cloud cause geometric information loss during projection, and

TABLE VII
COMPARISON WITH CLIP-BASED METHOD USING MEAN-IOU METRIC (%). S^i DENOTES THE SPLIT i IS USED FOR TESTING. THE BEST RESULTS ARE MASKED IN BOLD.

Dataset	Methods	2-way 1-shot			3-way 1-shot		
		S^0	S^1	mean	S^0	S^1	mean
S3DIS	PointCLIP [16]	51.21	52.68	51.95	47.32	47.65	47.49
	RCHP	**67.50**	**74.43**	**70.97**	**61.01**	**68.62**	**64.82**
ScanNet	PointCLIP [16]	40.75	37.22	38.99	31.50	30.27	30.89
	RCHP	**71.56**	**70.33**	**70.95**	**66.02**	**66.25**	**66.14**

TABLE VIII
ANALYSIS OF COMPUTATIONAL COST AND EXPERIMENTAL RESULTS UNDER 2-WAY 1-SHOT SETTING. OUR METHOD BETTER BALANCES BETWEEN COMPUTATION COST AND EXPERIMENT RESULTS. HERE *TinyCLIP-8M* REFERS TO *TinyCLIP-ViT-8M-16-Text-3M* MODEL.

Methods	#Params	FLOPs (G)	FPS	S3DIS	ScanNet
attMPTI	357.82K	152.65	1.47	54.86	41.69
QGPA	2.79M	16.30	38.68	62.76	56.51
DPA	4.85M	15.49	32.35	70.19	62.90
RCHP (*CLIP_rn50*)	38.32M + 3.95M	164.20	35.17	**70.97**	**70.95**
RCHP (*TinyCLIP-8M*)	8.28M + 3.46M	55.41	36.97	70.37	68.96

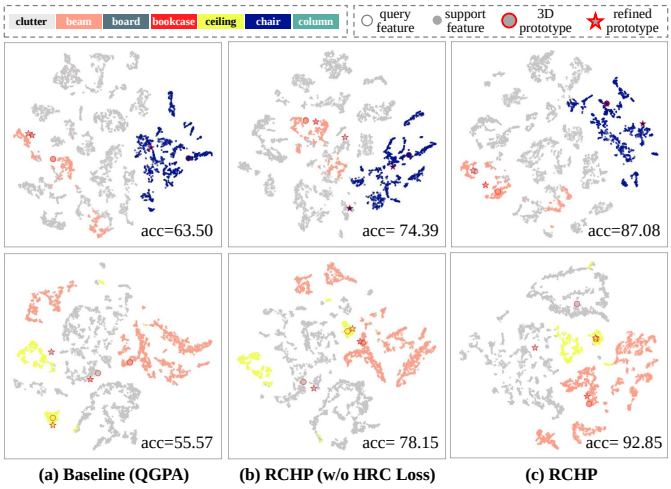


Fig. 5. Comparison of feature distribution and support prototype distribution. Take the *'2-way 1-shot S^0'* setting on S3DIS as an example. 'acc' denotes the segmentation accuracy for the selected episode. Best viewed in color.

extracting 2D feature maps further loses visual detail. In contrast, our method effectively preserves geometric structure while incorporating auxiliary guidance (*i.e.*, visual cues and semantic information), leading to superior performance.

Computational Complexity. In Table VIII, we present the number of parameters and computational complexity of our proposed method. Compared to prior methods, our approach (using *CLIP_rn50*) contains more parameters, primarily due to the CLIP visual encoder, which is more parameter-heavy. Nevertheless, other modules remain relatively moderate in size compared to state-of-the-art (SOTA) methods. To optimize computational efficiency, we use lightweight CLIP, *i.e.*, TinyCLIP [57], to replace standard *CLIP_rn50*. Notably, TinyCLIP effectively reduces the size and computational cost while maintaining competitive performance. Experimental results show that our model (using *TinyCLIP-ViT-8M-16-Text-3M*) provides a favorable trade-off between computational cost and performance, making it ideal for resource-constrained environments. In conclusion, our model effectively balances performance with computational complexity, offering superior segmentation results with reasonable efficiency.

Qualitative Results. We visualize the segmentation results on S3DIS and ScanNet under '2-way 1-shot S^0' setting. We compare our segmentation results with the GT mask and predictions from QGPA [10]. As shown in Fig. 4, QGPA often yields inaccurate segmentation results across various regions, notably incorrectly distinguishing background classes as foreground and confusing distinct foreground classes. In contrast, our method equipped with the HPE module and HRC loss, gradually corrects these errors. This success demonstrates the effectiveness of integrating heterogeneous prototypes.

D. Ablation Study

We conduct ablation experiments on the S3DIS dataset to evaluate the proposed components, including the HPE module and HRC loss. Additionally, we analyze the impact of different texts, as well as visual information and view selection strategies. For the HRC loss, we analyze its weight and compare it with other knowledge distillation (KD) methods. For the HPE module, we compare it with other multimodal fusion variants. Finally, we verify the effectiveness of our model under different baseline settings.

Effects of Different Components. We adopt QGPA as the baseline and verify the benefits of the proposed HPE module and HRC loss, as shown in Table IX.

1) Benefits of HRC loss: Compared with the baseline, applying $\mathcal{L}_{RCD}$ and $\mathcal{L}_{RCA}$ respectively can increase the model performance, with the latter one being more effective on '3-way 1-shot' setting because the third-order angle relations can better model more complex relations between more classes. Joint using $\mathcal{L}_{RCD}$ and $\mathcal{L}_{RCA}$ further boost segmentation results, exceeding the baseline by 1.38%$\sim$7.41%, demonstrating the effectiveness of HRC loss to reduce heterogeneous gaps.

2) Benefits of HPE module: When the HPE module only integrates text prototypes or 2D prototypes into refined prototypes, our model outperforms the Baseline by a margin of 1.09%$\sim$7.22% and 0.38%$\sim$9.94% in terms of mIoU, respectively. Here, compared to 2D prototypes, text prototypes achieve a larger improvement in most settings, as 2D prototypes are still generated from the projections of incomplete point clouds, while label text contains more comprehensive semantic information. The collaborative effect of three types of prototypes further improves the baseline by a large margin of 2.09%$\sim$11.44%.

3) Benefits of combining HPE module and HRC loss: After applying both HPE module and HRC loss to learn refined prototypes, RCHP achieves very significant improvement compared to the baseline, with a margin of 3.58%$\sim$11.88%, showing that HPE module and HRC loss mutually benefit each other to learning relation-consistent heterogeneous prototypes, thus improving performance.

4) Effects of different modalities during inference: Using only 2D information with HRC loss results in poor performance due to limited semantics and loss of geometric details in 3D-to-2D projection. Text information with HRC loss

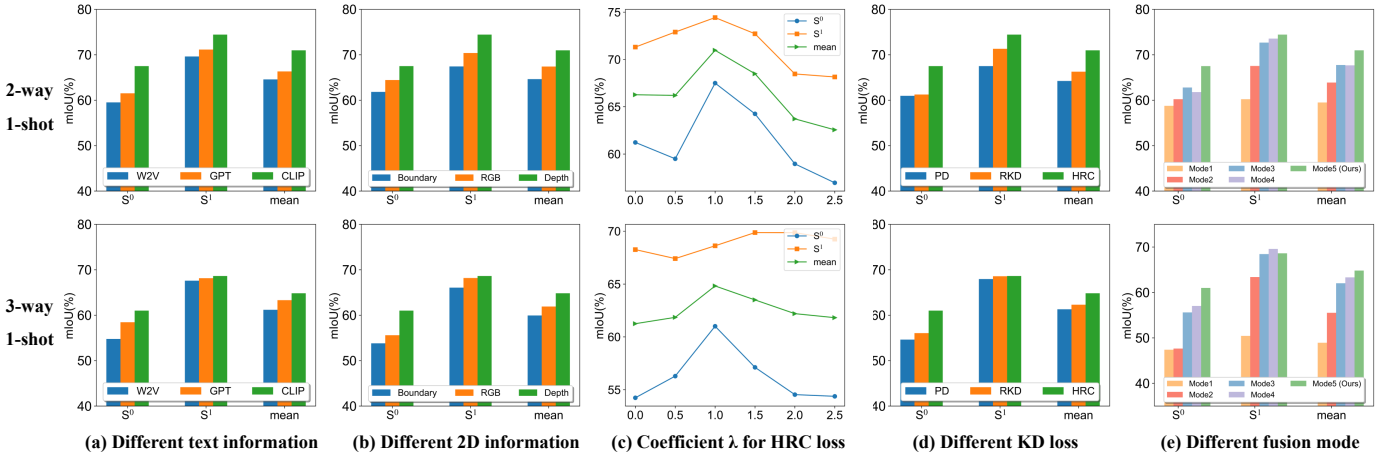


Fig. 6. Ablation study of different heterogeneous support information, hyper-parameters, distillation loss, and heterogeneous fusion strategies on S3DIS dataset under 2-way 1-shot setting.

TABLE IX
EFFECTS OF DIFFERENT COMPONENTS ON S3DIS USING MEAN-IOU METRIC (%). S^i DENOTES THE SPLIT i IS USED FOR TESTING.

Methods	HPE module			HRC loss		2-way 1-shot			3-way 1 shot		
	3D	Text	2D	L_{RCD}	L_{RCA}	S^0	S^1	mean	S^0	S^1	mean
Baseline	✓					58.96	63.08	61.02	47.86	59.48	53.67
+ HRC	✓			✓		60.18	66.89	63.54	49.44	63.31	56.38
	✓				✓	59.65	65.14	62.40	48.94	64.32	56.63
	✓			✓	✓	60.34	67.24	63.79	52.50	66.89	59.70
		✓		✓	✓	54.00	60.61	57.31	43.86	65.74	54.80
			✓	✓	✓	62.52	71.89	67.21	54.39	72.02	63.21
+ HPE	✓	✓				60.05	70.30	65.18	52.34	64.21	58.28
	✓		✓			59.96	67.10	63.53	49.15	65.25	57.20
	✓	✓	✓			61.23	71.31	66.27	54.24	68.26	61.25
RCHP	✓	✓	✓	✓	✓	**67.50**	**74.43**	**70.97**	**61.01**	**68.62**	**64.82**

outperforms individual 2D or 3D modalities by offering rich semantic knowledge. Combining 3D, 2D, and text prototypes with HRC loss achieves the best performance by integrating 3D's geometric precision, 2D's spatial context, and text's semantic richness.

5) Comparing feature and prototype distribution: In order to more intuitively compare the effects of different components, we also compare the feature distribution and class prototype distributions with the t-SNE visualization tool [58], as shown in Fig. 5. Here, prediction accuracy is also given for clearer comparison. We observe that the prototypes produced by QGPA exhibit confusion between different classes, and gaps persist between 3D prototypes and their refined prototypes. In contrast, our method not only ensures a continuous improvement in accuracy with the refined prototypes but also reduces the gap between the 3D prototype and refined prototypes for each class. This indicates that our refined prototypes are more discriminative, and HRC loss can effectively reduce the heterogeneous gap, facilitating a more effective feature space.

Effects of Different Text Information. In RCHP, we also explore other types of text information, such as word2vec [59] and diverse text descriptions generated by large language models (e.g., GPT-3 [60]). As shown in Fig. 6 (a), using *word2vec* gains limited improvement compared to CLIP text, due to its lack of visual context and less robust semantic representation, which are crucial for enhancing 3D point cloud segmentation. Using *LLM-generated descriptions* significantly improves the baseline but is still less effective than CLIP text, as LLMs tend to generate longer paragraphs with potentially class-unrelated details, thus introducing noise into the prototypes. Consequently, we opt for choosing CLIP texts for their stability and reliability as a text representation.

Effects of Different 2D Visual Information. In RCHP, we also investigate the impact of different types of 2D visual information projected from support 3D point clouds, including boundary maps, RGB images and depth maps. As shown in Fig. 6 (b), using *boundary maps* exhibit limited improvement over the baseline, due to their sparse and chaotic nature. *RGB images* notably improve the baseline but are less effective than depth maps, as lighting changes, shadows, and reflections lead to more noise to RGB information. Using *depth maps* achieves the best performance among the different 2D visual representations, as they exhibit smaller heterogeneous gaps to the 3D point cloud. Hence, we choose *depth maps* as a more stable and reliable 2D representation.

Ablation of 2D View Selection. As shown in Table X, fusing individual views yields limited and relatively similar improvements, as each view only captures specific visual details and lacks a holistic perspective. By contrast, jointly fusing all views achieves the best performance, as it combines the strengths of each view to provide the most comprehensive visual information. This fusion allows the model to leverage a more complete representation of the 3D structure. Thus we choose to fuse all views rather than relying on specific views.

Effects of λ for HRC Loss. Fig. 6 (c) illustrates the impact of the weight λ of the HRC loss in Eq. 13. Compared to $\lambda = 0$ (*i.e.*, HRC loss is deprecated), a rather small λ (*e.g.*, 0.5) results in performance degradation. This is because relation consistency with a rather small weight may introduce slight noise into heterogeneous prototype fusion to some extent, interfering with prototype discrimination. As λ increases, the consistency of IPR starts to enhance heterogeneous prototype fusion, resulting in significant performance improvement, and achieving the best performance when $\lambda = 1$. This demonstrates that HRC loss effectively alleviates heterogeneous gaps between prototypes. However, a larger λ (*e.g.*, 2) does not yield

TABLE X
EFFECTS OF FUSING DIFFERENT VIEWS OF 2D DEPTH MAPS ON S3DIS USING MEAN-IOU METRIC (%).

Settings	3D	2D						
		Front	Right	Behind	Left	Top	Down	**All**
2-way 1-shot	61.02	62.93	63.26	62.10	62.89	63.29	62.62	**63.53**
3-way 1-shot	53.67	56.45	56.73	57.10	54.85	56.78	56.88	**57.20**

TABLE XI
EFFECTIVENESS OF OUR METHOD UNDER DIFFERENT BASELINES.

Methods	2-way 1-shot			3-way 1-shot		
	S^0	S^1	mean	S^0	S^1	mean
ProtoNet [8]	48.39	49.98	49.19	40.81	45.07	42.94
RCHP	54.08	60.88	57.48	51.89	62.23	57.06
2CBR [13]	55.89	61.99	58.94	46.51	53.91	50.21
RCHP	62.93	67.54	65.24	56.33	66.83	61.58
QGPA [10]	58.96	63.08	61.02	47.86	59.48	53.67
RCHP	**71.56**	**70.33**	**70.95**	**66.02**	**66.25**	**66.14**

higher performance and may even harm model performance. In summary, an appropriate value of $\lambda = 1$ yields the best results.

Ablation of Different Prototype Distillation Methods. We compare the proposed HRC loss with other knowledge distillation methods, such as feature-level **p**rototype **d**istillation (PD) [14] and **r**elational **k**nowledge **d**istillation (RKD) [49]. As shown in Fig. 6 (d), PD performs worst due to its inability to capture and transfer complex relational knowledge between heterogeneous prototypes. Our HRC loss outperforms the original RKD loss due to enabling finer-grained feature interaction, highlighting important features, and enhancing feature discriminability.

Ablation of Heterogeneous Fusion Modes. We also design other heterogeneous information fusion modes. *Mode1:* Project heterogeneous support information into a shared feature space, and extract class prototypes to match with query points. *Mode2:* Extract heterogeneous class prototypes, match query points with each set of prototypes separately, and ensemble the matching results. *Mode3* and *Mode4* replace the prototype averaging operation in our HPE module with cross attention and learnable weighted sum, respectively. As shown in Fig. 6 (e), *Mode1* and *Mode2* are greatly affected by heterogeneous gaps, resulting in much poorer performance. *Mode3* and *Mode4* yield similar results to our prototype averaging operation in our HPE module because simple averaging is already effective for integrating class-wise prototypes across three modalities. Hence, we opt for average fusion for its simplicity and clarity.

Effectiveness on Different Baselines. To further verify the robustness and generalization of our method, in addition to QGPA [10] as the baseline, we also verify the effectiveness of the proposed HPE module and HRC loss on more baselines, including ProtoNet [8] and 2CBR [13]. As depicted in Table XI, our method demonstrates remarkable improvements on ProtoNet [8] and 2CBR [13], up to 5.55%∼17.16%, as we fully exploit the support set to provide powerful guidance information. Based on QGPA [10] as the baseline, our method achieves the best performance on all settings.

V. CONCLUSION

In this paper, we propose a Relation Consistency-guided Heterogeneous Prototype Learning Framework (RCHP) to address the FS-3DSeg challenge. RCHP effectively utilizes heterogeneous information from the original support set, generating and fusing diverse prototypes to enhance segmentation performance. To bridge the heterogeneous gap, we introduce a heterogeneous relation consistency loss. Extensive experiments on two indoor 3D segmentation datasets further demonstrate significant improvements over the baseline, achieving up to 12.05% and 18.91% higher mIoU on each dataset, outperforming state-of-the-art methods. Besides, experiments on SceneNN and NYU Depth V2 datasets further demonstrate the generalization of the proposed method. Our approach offers a novel and effective solution for FS-3DSeg. Main limitation of our works lies in that integrating CLIP into the framework introduces additional computational overhead, which could be a limitation for large-scale applications. Future work will focus on optimizing computational efficiency by model quantization and pruning, and leveraging large visual-language models to further enhance performance.

REFERENCES

[1] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *ICCV*, 2021, pp. 10 012–10 022.

[2] H. Ding, C. Liu, S. He, X. Jiang, P. H. Torr, and S. Bai, "Mose: A new dataset for video object segmentation in complex scenes," in *ICCV*, 2023, pp. 20 224–20 234.

[3] H. Ding, C. Liu, S. He, X. Jiang, and C. C. Loy, "Mevis: A large-scale benchmark for video segmentation with motion expressions," in *ICCV*, 2023, pp. 2694–2703.

[4] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *CVPR*, 2017, pp. 652–660.

[5] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[6] X. Lai, J. Liu, L. Jiang, L. Wang, H. Zhao, S. Liu, X. Qi, and J. Jia, "Stratified transformer for 3d point cloud segmentation," in *CVPR*, 2022, pp. 8500–8509.

[7] N. Zhang, Z. Pan, T. H. Li, W. Gao, and G. Li, "Improving graph representation for point cloud segmentation via attentive filtering," in *CVPR*, 2023, pp. 1244–1254.

[8] N. Zhao, T.-S. Chua, and G. H. Lee, "Few-shot 3d point cloud semantic segmentation," in *CVPR*, 2021, pp. 8873–8882.

[9] Y. Mao, Z. Guo, L. Xiaonan, Z. Yuan, and H. Guo, "Bidirectional feature globalization for few-shot semantic segmentation of 3d point cloud scenes," in *3DV*, 2022, pp. 505–514.

[10] S. He, X. Jiang, W. Jiang, and H. Ding, "Prototype adaption and projection for few-and zero-shot 3d point cloud semantic segmentation," *IEEE TIP*, 2023.

[11] Z. Ning, Z. Tian, G. Lu, and W. Pei, "Boosting few-shot 3d point cloud segmentation via query-guided enhancement," in *ACM MM*, 2023, pp. 1895–1904.

[12] D. Hu, S. Chen, H. Yang, and G. Wang, "Query-guided support prototypes for few-shot 3d indoor segmentation," *IEEE TCSVT*, 2023.

[13] G. Zhu, Y. Zhou, R. Yao, and H. Zhu, "Cross-class bias rectification for point cloud few-shot segmentation," *IEEE TMM*, 2023.

[14] J. Liu, W. Yin, H. Wang, Y. CHen, J.-J. Sonke, and E. Gavves, "Dynamic prototype adaptation with distillation for few-shot point cloud segmentation," in *CVPR*, 2024.

[15] C. Zhang, Z. Wu, X. Wu, Z. Zhao, and S. Wang, "Few-shot 3d point cloud semantic segmentation via stratified class-specific attention based transformer network," in *AAAI*, 2023, pp. 3410–3417.

[16] R. Zhang, Z. Guo, W. Zhang, K. Li, X. Miao, B. Cui, Y. Qiao, P. Gao, and H. Li, "Pointclip: Point cloud understanding by clip," in *CVPR*, 2022, pp. 8552–8562.

[17] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning (ICML)*, 2021, pp. 8748–8763.

[18] B. Graham, M. Engelcke, and L. Van Der Maaten, "3d semantic segmentation with submanifold sparse convolutional networks," in *CVPR*, 2018, pp. 9224–9232.

[19] H.-Y. Meng, L. Gao, Y.-K. Lai, and D. Manocha, "Vv-net: Voxel vae net with group convolutions for point cloud segmentation," in *ICCV*, 2019, pp. 8500–8508.

[20] T. Feng, W. Wang, F. Ma, and Y. Yang, "Lsk3dnet: Towards effective and efficient 3d perception with large sparse kernels," in *CVPR*, 2024, pp. 14 916–14 927.

[21] T. Feng, W. Wang, R. Quan, and Y. Yang, "Shape2scene: 3d scene representation learning through pre-training on shape data," in *ECCV*, 2025, pp. 73–91.

[22] T. Feng, W. Wang, X. Wang, Y. Yang, and Q. Zheng, "Clustering based point cloud representation learning for 3d analysis," in *ICCV*, 2023, pp. 8283–8294.

[23] J. Yin, D. Zhou, L. Zhang, J. Fang, C.-Z. Xu, J. Shen, and W. Wang, "Proposalcontrast: Unsupervised pre-training for lidar-based 3d object detection," in *ECCV*, 2022, pp. 17–33.

[24] Q. Meng, W. Wang, T. Zhou, J. Shen, L. Van Gool, and D. Dai, "Weakly supervised 3d object detection from lidar point cloud," in *ECCV*, 2020, pp. 515–531.

[25] A. J. Ratner, H. Ehrenberg, Z. Hussain, J. Dunnmon, and C. Ré, "Learning to compose domain-specific transformations for data augmentation," *NeurIPS*, vol. 30, 2017.

[26] Z. Chen, Y. Fu, Y. Zhang, Y.-G. Jiang, X. Xue, and L. Sigal, "Multi-level semantic feature augmentation for one-shot learning," *IEEE TIP*, vol. 28, no. 9, pp. 4594–4605, 2019.

[27] F. Pahde, O. Ostapenko, P. J. Hnichen, T. Klein, and M. Nabi, "Self-paced adversarial training for multimodal few-shot learning," in *WACV*, 2019, pp. 218–226.

[28] Y. Meng, M. Michalski, J. Huang, Y. Zhang, T. Abdelzaher, and J. Han, "Tuning language models as training data generators for augmentation-enhanced few-shot learning," in *ICML*, 2023, pp. 24 457–24 477.

[29] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *ICML*, 2017, pp. 1126–1135.

[30] Q. Cai, Y. Pan, T. Yao, C. Yan, and T. Mei, "Memory matching networks for one-shot image recognition," in *CVPR*, 2018, pp. 4080–4088.

[31] Z. Hu, Z. Li, X. Wang, and S. Zheng, "Unsupervised descriptor selection based meta-learning networks for few-shot classification," *Pattern Recognition*, vol. 122, p. 108304, 2022.

[32] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra *et al.*, "Matching networks for one shot learning," *NeurIPS*, vol. 29, 2016.

[33] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," *Advances in neural information processing systems*, vol. 30, 2017.

[34] B. Zhang, X. Li, Y. Ye, and S. Feng, "Prototype completion for few-shot learning," *TPAMI*, 2023.

[35] X. Huang and S. H. Choi, "Sapenet: self-attention based prototype enhancement network for few-shot learning," *Pattern Recognition*, vol. 135, p. 109170, 2023.

[36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[37] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *ICML*, 2022, pp. 12 888–12 900.

[38] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *NeurIPS*, 2023.

[39] Z. Yang, L. Li, K. Lin, J. Wang, C.-C. Lin, Z. Liu, and L. Wang, "The dawn of lmms: Preliminary explorations with gpt-4v (ision)," *arXiv preprint arXiv:2309.17421*, vol. 9, no. 1, p. 1, 2023.

[40] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "Minigpt-4: Enhancing vision-language understanding with advanced large language models," *arXiv preprint arXiv:2304.10592*, 2023.

[41] Z. Guo, R. Zhang, X. Zhu, Y. Tang, X. Ma, J. Han, K. Chen, P. Gao, X. Li, H. Li *et al.*, "Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following," *arXiv preprint arXiv:2309.00615*, 2023.

[42] Y. Tang, X. Han, X. Li, Q. Yu, Y. Hao, L. Hu, and M. Chen, "Minigpt-3d: Efficiently aligning 3d point clouds with large language models using 2d priors," *arXiv preprint arXiv:2405.01413*, 2024.

[43] D. Liu, X. Huang, Y. Hou, Z. Wang, Z. Yin, Y. Gong, P. Gao, and W. Ouyang, "Uni3d-llm: Unifying point cloud perception, generation and editing with large language models," *arXiv preprint arXiv:2402.03327*, 2024.

[44] X. Zhu, R. Zhang, B. He, Z. Guo, Z. Zeng, Z. Qin, S. Zhang, and P. Gao, "Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning," pp. 2639–2650, 2023.

[45] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra *et al.*, "Matching networks for one shot learning," *Advances in neural information processing systems*, vol. 29, 2016.

[46] Z. Zhuang, R. Li, K. Jia, Q. Wang, Y. Li, and M. Tan, "Perception-aware multi-sensor fusion for 3d lidar semantic segmentation," in *ICCV*, 2021, pp. 16 280–16 290.

[47] Y. Lu, Q. Jiang, R. Chen, Y. Hou, X. Zhu, and Y. Ma, "See more and know more: Zero-shot point cloud segmentation via multi-modal visual data," in *ICCV*, 2023, pp. 21 674–21 684.

[48] X. Chu, W. Ouyang, H. Li, and X. Wang, "Structured feature learning for pose estimation," in *CVPR*, 2016, pp. 4715–4723.

[49] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *CVPR*, 2019, pp. 3967–3976.

[50] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese, "3d semantic parsing of large-scale indoor spaces," in *CVPR*, 2016, pp. 1534–1543.

[51] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *CVPR*, 2017, pp. 5828–5839.

[52] B.-S. Hua, Q.-H. Pham, D. T. Nguyen, M.-K. Tran, L.-F. Yu, and S.-K. Yeung, "Scenenn: A scene meshes dataset with annotations," in *3DV*, 2016, pp. 92–101.

[53] B.-S. Hua, M.-K. Tran, and S.-K. Yeung, "Pointwise convolutional neural networks," in *CVPR*, 2018, pp. 984–993.

[54] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012, pp. 746–760.

[55] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3d.net: A new large-scale point cloud classification benchmark," in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-1-W1, 2017, pp. 91–98.

[56] V. G. Satorras and J. B. Estrach, "Few-shot learning with graph neural networks," in *ICLR*, 2018.

[57] K. Wu, H. Peng, Z. Zhou, B. Xiao, M. Liu, L. Yuan, H. Xuan, M. Valenzuela, X. S. Chen, X. Wang, H. Chao, and H. Hu, "Tinyclip: Clip distillation via affinity mimicking and weight inheritance," in *ICCV*, 2023, pp. 21 970–21 980.

[58] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[59] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *International Conference on Learning Representations (ICLR)*, 2013.

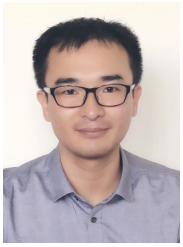
[60] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," vol. 33, pp. 1877–1901, 2020.

Lili Wei received the BSc degree in Software Engineering from the School of Computer and Communication, Lanzhou University of Technology, Gansu, China, in 2018. Currently, she is working toward a PhD degree at the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. Her research interests include computer vision and machine learning.

Congyan Lang received the Ph.D. degree from the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China, in 2006. She was a Visiting Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, from 2010 to 2011. From 2014 to 2015, she visited the Department of Computer Science, University of Rochester, Rochester, NY, USA, as a Visiting Researcher. She is currently a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. Her current research interests include multimedia information retrieval and analysis, machine learning, and computer vision.

Zheming Xu received the M.S. degree in advanced computer science and the M.S. degree in computer science from the University of York, UK, and Beijing Jiaotong University, Beijing, China, in 2020 and 2021, respectively. She is now a Ph.D. student at Beijing Jiaotong University. Her research interests focus on machine learning and computer vision, especially in multi-view learning.

Liqian Liang received the BSc degree and PhD degree in Computer Science from the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China, in 2014 and 2021. She has been a visiting scholar in the School of Computer Science, The University of Adelaide, Australia, from 2016 to 2017. Her research interests include computer vision and machine learning.



Jun Liu received the PhD degree from Nanyang Technological University, the MSC degree from Fudan University, and the BEng degree from Central South University. He is currently a Chair and Full Professor at Lancaster University. His research interests include computer vision, artificial intelligence, and digital health. He is an Associate Editor of IEEE Transactions on Image Processing and IEEE Transactions on Biometrics, Behavior, and Identity Science, and serves/has served as an Area Chair of CVPR, ECCV, ICML, NeurIPS, ICLR, MM, and WACV, etc.