DyPolySeg: Taylor Series-Inspired Dynamic Polynomial Fitting Network for Few-shot Point Cloud Semantic Segmentation

Changshuo Wang [1] **Xiang Fang** [2] **Prayag Tiwari** [3]

Abstract

Few-shot point cloud semantic segmentation effectively addresses data scarcity by identifying unlabeled query samples through semantic prototypes generated from a small set of labeled support samples. However, pre-training-based methods suffer from domain shifts and increased training time. Additionally, existing methods using DGCNN as the backbone have limited geometric structure modeling capabilities and struggle to bridge the categorical information gap between query and support sets. To address these challenges, we propose **DyPolySeg**, a pre-training-free Dynamic Polynomial fitting network for few-shot point cloud semantic segmentation. Specifically, we design a unified Dynamic Polynomial Convolution (**DyPolyConv**) that extracts flat and detailed features of local geometry through Low-order Convolution (**LoConv**) and Dynamic High-order Convolution (**DyHoConv**), complemented by Mamba Block for capturing global context information. Furthermore, we propose a lightweight Prototype Completion Module (**PCM**) that reduces structural differences through self-enhancement and interactive enhancement between query and support sets. Experiments demonstrate that DyPolySeg achieves state-of-the-art performance on S3DIS and ScanNet datasets.

1. Introduction

Point cloud semantic segmentation (Xu et al., 2024; Zhang et al., 2023b; 2024b; He et al., 2024) serves as a fundamental task in 3D scene understanding, with critical applications spanning autonomous driving (Zhao et al., 2023; Chib & Singh, 2023), robotics (Soori et al., 2023; Goel & Gupta, 2020), and augmented reality (Devagiri et al., 2022; Sereno et al., 2020). Despite its significance, the acquisition of large-scale annotated point cloud data(Jiang et al., 2023) remains a resource-intensive challenge, requiring substantial human effort and time investment. This limitation significantly constrains the practical deployment of deep learning methods (Ning et al., 2024a; Wang et al., 2021; 2023; Ning et al., 2024b; Yu et al., 2024) in real-world scenarios. To address this fundamental challenge, researchers (Li et al., 2024; Xiong et al., 2024) have increasingly turned to few-shot learning approaches for point cloud segmentation tasks.

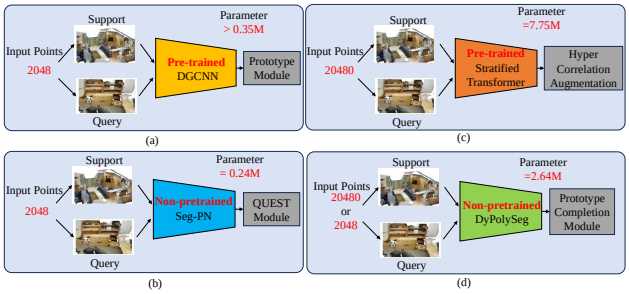


Figure 1. Architectural comparison of representative few-shot point cloud segmentation methods. (a) Conventional approaches (Zhao et al., 2021b; He et al., 2023) rely on pre-trained DGCNN backbones coupled with various prototype alignment modules. (b) SegPN (Zhu et al., 2024) eliminates the need for pre-training but suffers from limited representational capacity. (c) COSeg (An et al., 2024) adopts a pre-trained Straight Transformer architecture and processes 20,480 points to mitigate foreground information leakage. (d) Our proposed DyPolySeg innovatively models local structures through polynomial fitting and incorporates a lightweight prototype completion module for enhanced performance.

Few-shot point cloud semantic segmentation represents an innovative paradigm that leverages semantic prototypes derived from a limited set of labeled support samples to effectively identify and segment unlabeled query samples. As illustrated in Fig. 1(a), this field was pioneered by Zhao et al. (Zhao et al., 2021b), who introduced the AttMPTI method based on a pre-trained DGCNN (Wang et al., 2019). Their groundbreaking work established a foundation that

[1]Cyber Security Research Center, Nanyang Technological University, Singapore [2]ERI@N, Interdisciplinary Graduate Programme, Nanyang Technological University, Singapore [3]School of Information Technology, Halmstad University, Sweden. Correspondence to: Xiang Fang <xfang9508@gmail.com>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

subsequent research has built upon, with various approaches (Mao et al., 2022; Lai et al., 2022a; Zhu et al., 2023) progressively enhancing segmentation performance. While Zhu et al. (Zhu et al., 2024) made significant strides with their pre-training-free SegNN approach (Fig. 1(b)), their model's capability in local structure representation remains limited. An et al. (An et al., 2024) proposed COSeg (Fig. 1(c)), introducing a novel approach to prevent foreground leakage by expanding input points to 20,480 through uniform sampling. However, this methodology can lead to suboptimal scenarios where foreground information either becomes extremely sparse or is entirely absent from the sampled points.

Current few-shot point cloud segmentation methods (Wang et al., 2025b; Zhao et al., 2021b; Mao et al., 2022; Zhu et al., 2023; 2024; He et al., 2023; An et al., 2024) face three critical challenges: First, pre-training on "seen" categories before fine-tuning on "unseen" ones increases computational cost and leads to domain shifts. Second, using DGCNN as backbone limits the model's ability to capture complex 3D geometric structures, particularly crucial in few-shot scenarios. Finally, the limited support samples fail to fully represent category distributions, creating biased prototypes that affect feature matching accuracy between query and support sets.

To address these challenges, we propose DyPolySeg, an innovative pre-training-free dynamic polynomial fitting network. First, we introduce a unified Dynamic Polynomial Convolution (DyPolyConv) that combines geometry-prior-driven low-order convolution (LoConv) and dynamic high-order convolution (DyHoConv) for local geometric feature modeling. LoConv efficiently captures basic flat information through position encoding, while DyHoConv adaptively models complex local details through learned spatial priors, integrating high-order convolution weights and power exponents. Second, we incorporate Mamba Block to establish a robust "local-global" structure through strategic stacking of DyPolyConv and Mamba Block, effectively fusing local geometric features with global semantic information. Finally, we develop a lightweight stackable Prototype Completion Module (PCM) featuring Self-Enhancement Module (SEM) and Interactive Enhancement Module (IEM): the former learns feature distribution patterns within query and support sets, while the latter refines prototype bias through fine-grained feature correspondence. Multiple stacked PCM modules enable gradual reduction of structural differences, enhancing segmentation accuracy.

Our contributions are summarized as follows:

- We propose DyPolySeg, a novel framework that achieves comprehensive scene understanding through an innovative "local-global" structure constructed by strategically integrating dynamic polynomial convolution and Mamba Block.
- We develop a unified Dynamic Polynomial Convolution (DyPolyConv) that precisely captures local geometric structures through the synergistic combination of low-order convolution (LoConv) and dynamic high-order convolution (DyHoConv).
- We introduce an efficient Prototype Completion Module (PCM) that effectively minimizes structural differences through self-enhancement and interactive enhancement modules between query and support sets.

2. Related Works

2.1. Point Cloud Semantic Segmentation

Point cloud semantic segmentation (Wang et al., 2022a; Wu et al., 2022; He & Ding, 2024; Zhang et al., 2024a; Wang et al., 2022b) represents a fundamental task in 3D scene understanding, focusing on assigning semantic labels to individual points within a point cloud. PointNet (Qi et al., 2017a) pioneered this field by introducing direct point cloud processing through shared MLPs and global max pooling, though its capacity for local feature extraction remained limited. Subsequent research has primarily focused on enhancing local feature representation capabilities. PointNet++ (Qi et al., 2017b) advanced the field by implementing hierarchical sampling and local feature aggregation, while DGCNN (Wang et al., 2019) introduced the innovative EdgeConv operation to capture point relationships through dynamically constructed local graph structures.

Recent years have witnessed significant advancements through three main approaches: Transformer-based architectures (Zhao et al., 2021a; Lai et al., 2022b; Wang et al., 2024), Mamba-based models (Liang et al., 2024b; Han et al., 2024), and self-supervised pre-training strategies (Chen et al., 2024; Pang et al., 2022). Point Transformer (Zhao et al., 2021a) effectively leveraged self-attention mechanisms to capture long-range dependencies, while PointMamba (Liang et al., 2024b) innovatively employed space-filling curves for efficient point tokenization and utilized a non-hierarchical Mamba encoder as its backbone. DAPT (Zhou et al., 2024) introduces a novel dynamic adapter and seamlessly integrates with prompt tuning, substantially reduces training costs while achieving impressive performance. PointGST (Liang et al., 2024a) innovatively proposes fine-tuning in spectral domain, resulting in significantly reduced training parameters and superior performance across diverse point cloud tasks. PointGPT (Chen et al., 2024) introduced a novel point cloud autoregressive generation task for pre-training transformer models, achieving superior performance in downstream point cloud understanding tasks. While these approaches demonstrate impressive performance in fully supervised scenarios(Shi et al., 2023; Ning et al., 2023; Wang et al., 2025d;a;c; Fang et al.,

2025), they heavily rely on large-scale annotated datasets and struggle to generalize effectively to novel categories or data-scarce environments.

2.2. Few-shot Point Cloud Semantic Segmentation

Few-shot point cloud semantic segmentation addresses the challenge of learning semantic segmentation capabilities for new categories with limited annotated data. AttMPTI (Zhao et al., 2021b) established the foundation for this field by achieving few-shot segmentation through multi-prototype generation and label propagation. Subsequent research has advanced along three primary directions: feature enhancement, prototype optimization, and domain adaptation.

In the feature enhancement domain, BFG (Mao et al., 2022) introduced bilateral feature globalization to improve performance through feature interaction between support and query sets, while SCAT (Zhang et al., 2023a) leveraged hierarchical attention mechanisms to capture long-range dependencies. For prototype optimization, PAP (He et al., 2023) enhanced prototype representation through adaptive prototype adaptation and projection strategies.

Recent works have introduced sophisticated approaches: DLE (Li et al., 2024) incorporated structural information for precise target region localization while minimizing background interference, utilizing intra-target similarities for complete target segmentation. DENet (Xiong et al., 2024) comprehensively addressed intra-class diversity and semantic inconsistency through a bilateral mutual aggregation module and consistency purification strategy. GPCPR (Wei et al., 2024) innovatively leveraged LLM-generated content and pseudo-query context to optimize prototypes, effectively mitigating categorical information bias.

Recent architectural innovations include SegNN (Zhu et al., 2024), which introduced both parameter-free and parameterized variants achieving competitive performance without pre-training. COSeg (An et al., 2024) proposed a novel approach to prevent foreground leakage by expanding input points to 20,480, though this strategy can potentially lead to sparse foreground representation or information loss. TaylorSeg (Wang et al., 2025b), a pretraining-free network for few-shot point cloud semantic segmentation, uses TaylorConv, inspired by Taylor series, to fit local structures as polynomials. It includes non-parametric TaylorSeg-NN and parametric TaylorSeg-PN with an Adaptive Push-Pull module to align feature distributions, achieving high performance without pretraining.

3. Method

In this section, we first formalize the few-shot point cloud semantic segmentation problem and establish its theoretical foundations based on Taylor series and dynamic convolution. We then present our Dynamic Polynomial Convolution (DyPolyConv) with its design principles and theoretical guarantees. Subsequently, we introduce our Prototype Completion Module (PCM) and its innovative mechanisms. Finally, we detail the complete DyPolySeg architecture (see 2).

3.1. Preliminary

3.1.1. Problem Definition

In few-shot point cloud semantic segmentation, we adopt an episodic learning paradigm where the dataset is divided into seen classes $\mathcal{C}_{seen}$ and unseen classes $\mathcal{C}_{unseen}$. Each task is formulated as an N-way K-shot problem with a support set $\mathcal{S} = \{(I_s^{n,k}, M_s^{n,k})\}_{n=1,k=1}^{N,K}$ and a query set $\mathcal{Q} = \{I_q^i\}_{i=1}^{H}$. Here, $I_s^{n,k} \in \mathbb{R}^{T\times(3+C)}$ represents the k-th point cloud sample of the n-th class in the support set, where T denotes the number of points, 3 represents 3D coordinates (x, y, z), and C indicates additional feature dimensions. Each support sample is paired with a binary segmentation mask $M_s^{n,k} \in \mathbb{R}^{T\times 1}$. The query set contains H samples, where each $I_q^i \in \mathbb{R}^{T\times(3+C)}$ represents a point cloud to be segmented. The objective is to leverage K labeled samples from N classes in the support set $\mathcal{S}$ to segment query samples into N target classes plus one background class.

3.1.2. Taylor Series

Taylor series (Rudin et al., 1964) serves as a fundamental mathematical tool that enables the representation of a function through an infinite sum of terms derived from its derivatives at a specific point. Formally, for a sufficiently smooth function $f(x)$, its Taylor series expansion at point x_0 is given by:

$$f(x) = f(x_0) + \sum_{n=1}^{\infty} \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n, \tag{1}$$

where $f^{(n)}(x_0)$ denotes the n-th order derivative of function $f(x)$ evaluated at point x_0. This series enables accurate local approximation of function values by leveraging information about the function's derivatives at a single point, making it particularly suitable for modeling complex geometric structures in point clouds.

3.1.3. Dynamic Convolution

Traditional convolution operations with fixed kernels inherently limit the model's adaptability to diverse local structures. Dynamic convolution (Yang et al., 2019) overcomes this limitation by introducing adaptive weight generation based on input characteristics. In the context of point clouds, dynamic convolution is formulated as:

$$g_i = \mathcal{A}(\{K(p_i, p_j)f_j | p_j \in \mathcal{N}_i\}), \tag{2}$$

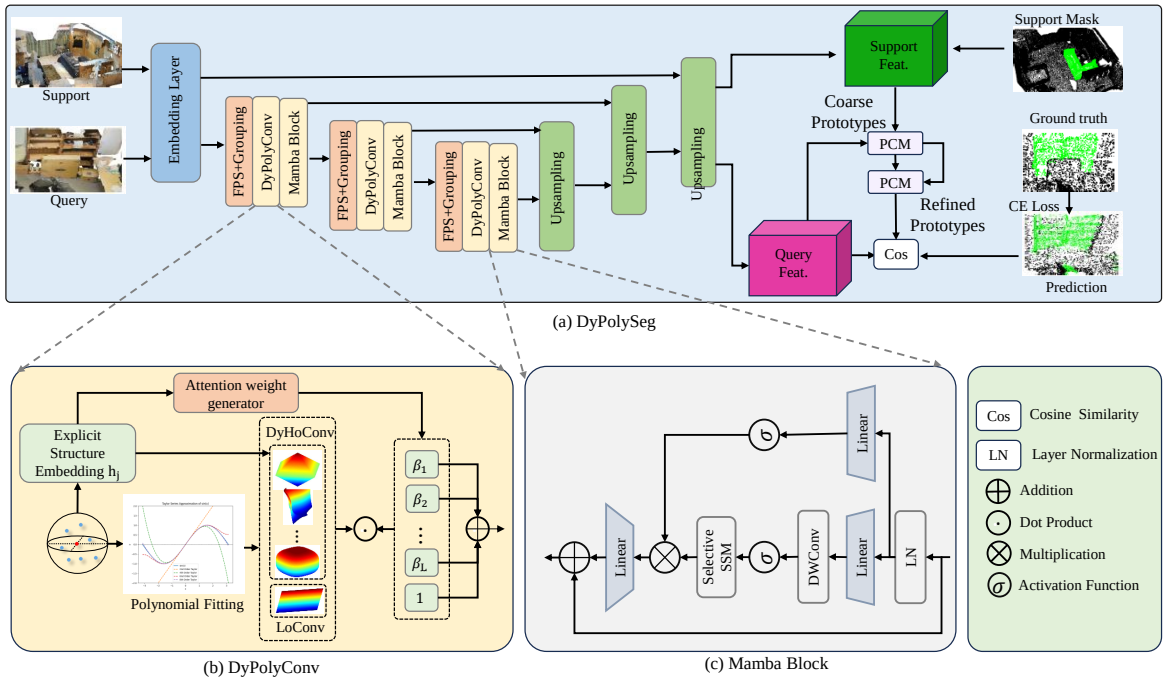


Figure 2. Architectural overview of DyPolySeg. (a) Pipeline architecture: an embedding layer for point cloud representation, feature extraction modules (FPS, DyPolyConv, DyHoConv, and Mamba Block), a Prototype Completion Module (PCM), and a segmentation layer for query point clouds. (b) The DyPolyConv module combines attention-based weight generation with polynomial fitting for local geometric feature extraction. (c) The Mamba Block architecture integrates linear transformations, selective kernel convolutions, and normalization for global context modeling.

where g_i represents the output feature, $\mathcal{A}$ denotes an aggregation function (e.g., max pooling or summation), $\mathcal{N}_i$ defines point p_i's local neighborhood, $K(p_i, p_j)$ represents dynamically generated convolution weights, and f_j indicates the input feature of neighborhood point p_j.

The core innovation lies in the generation of convolution weights $K(p_i, p_j)$. These weights are computed dynamically through a weighted combination of base matrices:

$$K(p_i, p_j) = \sum_{m=1}^{M} \alpha_m(p_i, p_j) W_m, \tag{3}$$

where $\{W_m\}_{m=1}^{M}$ denotes base weight matrices, $\alpha_m(p_i, p_j)$ represents combination coefficients generated by neural networks based on spatial relationships between p_i and p_j, and M indicates the number of base matrices.

3.2. Dynamic Polynomial Convolution

Inspired by Taylor series (Rudin et al., 1964), we propose a novel dynamic polynomial convolution (DyPolyConv) that models local structures through multi-order point convolutions, enabling accurate representation of complex geometric features. Our approach can be formulated as:

$$g_i = \underbrace{\varphi(f_i)}_{LoConv} + \underbrace{\sum_{l=1}^{L} \beta_l \cdot h^l((f_j - f_i)^{n_l})}_{DyHoConv}, \tag{4}$$

where φ denotes a shared MLP, β_l represents attention coefficients, L is the number of high-order convolutions, and n_l denotes the power exponent for the l-th convolution.

The high-order convolution operation h^l is defined as:

$$h((f_j - f_i)^n) = \mathcal{A}(\mathcal{T}(f_i, f_j) | p_j \in \mathcal{N}(p_i)), \tag{5}$$

with transformation function:

$$\mathcal{T}(f_i, f_j) = \left(\frac{\omega_j \cdot (f_j - f_i)}{|\omega_j \cdot (f_j - f_i)|} \right)^s \cdot |\omega_j \cdot (f_j - f_i)|^n, \tag{6}$$

where $|\cdot|$ denotes element-wise absolute value, $s \in \{0, 1\}$ controls the sign function, and n is a learnable parameter.

Our transformation function $\mathcal{T}(\cdot)$ exhibits remarkable generality. It reduces to an Affine basis function (ABF) (Rosenblatt, 1958) when $s = 1$, $n = 1$, and $f_i = 0$:

$$\mathcal{T} = \left(\frac{\omega_j \cdot f_j}{|\omega_j \cdot f_j|} \right)^1 \cdot |\omega_j \cdot f_j|^1 = \omega_j \cdot f_j, \tag{7}$$

Furthermore, it becomes a radial basis function (RBF) (Moody & Darken, 1989) when $s = 0$ and $n = 2$:

$$\mathcal{T} = (f_j - f_i)^2 = \left(\frac{\omega_j \cdot (f_j - f_i)}{|\omega_j \cdot (f_j - f_i)|} \right)^0 \cdot |\omega_j \cdot (f_j - f_i)|^2. \tag{8}$$

3.3. Improving Dynamic Polynomial Convolution

While our basic Dynamic Polynomial Convolution framework provides a solid foundation for capturing complex local geometric structures, practical applications require balancing expressive power with computational efficiency. To address limitations in our initial formulation, we enhance model performance through three complementary optimization strategies:

3.3.1. Enhanced Low-order Convolution

The original low-order convolution primarily focuses on center point features and cannot effectively capture comprehensive local structural information. We extend the functionality by incorporating neighboring point information:

$$g_L = \mathcal{A}(\varphi(f_j)|p_j \in \mathcal{N}(p_i)), \tag{9}$$

where g_L represents the enhanced output features, $\mathcal{A}$ denotes max pooling, and φ is a shared MLP. This improvement enables the model to observe the entire local region's feature distribution, providing richer contextual information.

3.3.2. Explicit Structure Integration

To enhance spatial relationship perception, we explicitly integrate geometric relationships through comprehensive spatial features:

$$\mathbf{h}_j = [p_i, p_j, p_j - p_i, \|p_i, p_j\|] \in \mathbb{R}^{10}, \tag{10}$$

where $\mathbf{h}_j$ contains center point coordinates, neighborhood coordinates, relative displacement, and Euclidean distance. These features generate three key adaptive parameters:

$$\omega_j = \mathbf{h}_j \mathbf{V}_h, \tag{11}$$

$$\beta_l = \frac{\exp(\mathbf{h}_j \mathbf{V}_l)}{\sum_{t=1}^{L} \exp(\mathbf{h}_j \mathbf{V}_t)}, \tag{12}$$

$$s = \begin{cases} 1, & \text{if } \sigma(\mathbf{h}_j \mathbf{V}_s) > 0.5 \\ 0, & \text{otherwise} \end{cases}, \tag{13}$$

where $\mathbf{V}_h, \mathbf{V}_l, \mathbf{V}_s \in \mathbb{R}^{10 \times C_{out}}$ are learnable transformation matrices. The learnable parameter s enables automatic switching between affine and radial basis function behaviors based on local geometric context, significantly enhancing adaptability to different geometric shapes.

3.3.3. Computational Efficiency Optimization

To address computational burden and numerical instability from exponential operations, we employ logarithmic space transformation:

$$\mathcal{T}(f_i, f_j) = \text{sgn}(f_i, f_j)^s \cdot n \cdot \log(|\omega_j \cdot (f_j - f_i)| + \epsilon), \tag{14}$$

where $\epsilon = 10^{-6}$ prevents numerical instability. This transformation reduces computational complexity from $O(n^2)$ to $O(n \log n)$ while improving numerical stability and memory efficiency, making it suitable for large-scale point cloud processing.

3.4. Prototype Completion Module

Few-shot point cloud segmentation faces a critical challenge: limited semantic information in support sets and class discrepancies between support and query sets affect semantic matching accuracy, leading to incorrect segmentation results. To address this problem, we propose a lightweight stackable Prototype Completion Module (PCM) that aims to effectively reduce the feature gap between support and query sets while maintaining computational efficiency. PCM comprises two complementary components: Self-Enhancement Module (SEM) and Interactive Enhancement Module (IEM).

First, we obtain rough prototype features $\mathbf{V}$ through support set features and support set masks. Then, we extract representative scene features from support and query sets through max pooling:

$$\mathbf{F}_q = \text{MaxPool}(\mathcal{Q}), \quad \mathbf{F}_s = \text{MaxPool}(\mathcal{S}), \tag{15}$$

where $\mathbf{F}_q \in \mathbb{R}^{D \times C}$ and $\mathbf{F}_s \in \mathbb{R}^{D \times C}$ represent the pooled features of query set $\mathcal{Q}$ and support set $\mathcal{S}$ respectively, D denotes the pooled feature dimension (typically much smaller than the original point count), and C represents the number of feature channels. This step significantly reduces memory consumption while preserving global semantic information, providing computationally efficient representations for subsequent cross-set feature comparisons.

3.4.1. Self-Enhancement Module

SEM is designed to enhance prototype representational power by exploring intra-set feature distributions, reducing intra-set class diversity. It achieves this goal by computing auto-correlation matrices:

$$\mathbf{G}_q = \mathbf{F}_q^\top \mathbf{F}_q, \quad \mathbf{G}_s = \mathbf{F}_s^\top \mathbf{F}_s, \tag{16}$$

where $\mathbf{G}_q, \mathbf{G}_s \in \mathbb{R}^{C \times C}$ are the auto-correlation matrices for query and support sets respectively, capturing relationships between feature channels within each set. These matrices reveal internal structural patterns in the feature space that can be used to improve prototype discriminability.

SEM subsequently generates attention-based enhanced prototypes:

$$\mathbf{A}_q = \sigma(\mathbf{U}_q\mathbf{G}_q)/\sqrt{D}, \quad \mathbf{A}_s = \sigma(\mathbf{U}_s\mathbf{G}_s)/\sqrt{D}, \quad (17)$$

$$\mathbf{V}_{\text{self}} = \phi_1(\mathbf{A}_q \odot \mathbf{V}) + \phi_1(\mathbf{A}_s \odot \mathbf{V}), \quad (18)$$

where $\mathbf{U}_q, \mathbf{U}_s \in \mathbb{R}^{C\times C}$ are learnable weight matrices used to transform auto-correlation matrices into attention weights, σ denotes the Sigmoid function, $\sqrt{D}$ is a normalization factor, $\mathbf{A}_q, \mathbf{A}_s \in \mathbb{R}^{C\times C}$ are attention weight matrices, $\odot$ represents element-wise multiplication, ϕ_1 is a transformation function, $\mathbf{V} \in \mathbb{R}^{(N+1)\times C}$ represents initial prototype features (including prototypes for N target classes and 1 background class), and $\mathbf{V}_{\text{self}}$ represents the self-enhanced prototype features.

This design enables the model to learn feature distribution patterns within each set and use them to guide prototype enhancement, reducing intra-set class diversity.

3.4.2. Interactive Enhancement Module

While SEM focuses on intra-set feature enhancement, IEM concentrates on eliminating domain gaps by modeling consistency and differences between support and query sets. IEM first computes cross-set correlation matrices:

$$\mathbf{C} = (\mathbf{F}_q^\top \mathbf{F}_s)/\sqrt{D}, \quad (19)$$

where $\mathbf{C} \in \mathbb{R}^{C\times C}$ represents correlations between query and support set features, and $\sqrt{D}$ is a normalization factor. This matrix captures shared information and differences between the two sets, providing important reference for subsequent prototype refinement.

IEM leverages this cross-set correlation information to enrich prototype semantics:

$$\mathbf{A}_{\text{inter}} = \phi_2(\mathbf{A}_c\mathbf{V}), \quad \mathbf{A}_c = \text{softmax}(\mathbf{C}), \quad (20)$$

where $\mathbf{A}_c \in \mathbb{R}^{C\times C}$ is the cross-set correlation matrix normalized through softmax, ϕ_2 is a transformation function, and $\mathbf{A}_{\text{inter}}$ represents interactive enhancement-based prototype features. This step enables the model to learn shared patterns between support and query sets, helping prototypes adapt to feature distributions across different domains.

The final refined prototypes are obtained through residual connections:

$$\mathbf{V}_{\text{out}} = \mathbf{V}_{\text{self}} + \mathbf{A}_{\text{inter}} + \mathbf{V}, \quad (21)$$

where $\mathbf{V}_{\text{out}} \in \mathbb{R}^{(N+1)\times C}$ represents the final prototype features by PCM. The residual connection design ensures lossless information transmission, avoiding feature degradation problems while facilitating gradient backpropagation.

The design of the PCM module allows it to be stacked in multiple layers, forming a progressive prototype refinement process. Each PCM layer can further reduce structural differences between support and query sets, improving prototype quality.

3.5. DyPolySeg Overview

As shown in Figure 2, we propose DyPolySeg, formulating few-shot point cloud semantic segmentation as a dual optimization problem that combines local structure modeling with prototype matching.

Our framework adopts an encoder-decoder architecture. The encoder consists of stacked "local-global" blocks that integrate DyPolyConv for local structural feature extraction and Mamba blocks for global context modeling. The decoder progressively recovers point cloud resolution through inverse interpolation. A lightweight Prototype Completion Module bridges the semantic gap between support and query sets by generating discriminative prototypes for both target classes and background. The final segmentation is obtained by computing similarities between query point features and the refined prototypes.

4. Experiments

4.1. Datasets and Evaluation Metrics

Our experiments utilize three distinct point cloud datasets.

The S3DIS dataset (Armeni et al., 2016) encompasses RGB point clouds collected from 272 rooms distributed across 6 indoor environments, with points categorized into 13 semantic labels (12 categories and clutter). Following established protocols (Zhao et al., 2021b), we process each scene into 1m × 1m blocks, extracting 2048 points per block, yielding 7547 blocks in total.

For additional validation, we employ the ScanNet dataset (Dai et al., 2017), comprising 1513 scanned indoor scenes. The dataset features comprehensive point-wise annotations across 20 semantic categories, excluding unannotated regions. Our preprocessing pipeline (Zhao et al., 2021b) generates 36350 blocks, maintaining consistent 2048-point sampling per block.

To evaluate classification capabilities, we incorporate ScanObjectNN (Uy et al., 2019), a real-world point cloud dataset that surpasses ModelNet40 in complexity through its inclusion of background elements and occlusion effects. This collection encompasses 2902 objects across 15 categories, with our evaluation focusing on its most challenging variant (PB_T50_RS).

Evaluation Metrics. we adopt the standard mean Intersection over Union (mIoU) metric across all categories.

Table 1. **Few-shot Results (%) on S3DIS.** S_i denotes the split i is used for testing. *Avg* is their average mIoU.

Methods	Param.	2-way						3-way					
		1-shot			5-shot			1-shot			5-shot		
		S_0	S_1	*Avg*	S_0	S_1	*Avg*	S_0	S_1	*Avg*	S_0	S_1	*Avg*
DGCNN (Wang et al., 2019)	0.62 M	36.34	38.79	37.57	56.49	56.99	56.74	30.05	32.19	31.12	46.88	47.57	47.23
ProtoNet (Snell et al., 2017)	0.27 M	48.39	49.98	49.19	57.34	63.22	60.28	40.81	45.07	42.94	49.05	53.42	51.24
MPTI (Zhao et al., 2021b)	0.29 M	52.27	51.48	51.88	58.93	60.56	59.75	44.27	46.92	45.60	51.74	48.57	50.16
AttMPTI (Zhao et al., 2021b)	0.37 M	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 (Mao et al., 2022)	-	55.60	55.98	55.79	63.71	66.62	65.17	46.18	48.36	47.27	55.05	57.80	56.43
2CBR (Zhu et al., 2023)	0.35 M	55.89	61.99	58.94	63.55	67.51	65.53	46.51	53.91	50.21	55.51	58.07	56.79
PAP3D (He et al., 2023)	2.45 M	59.45	66.08	62.76	65.40	70.30	67.85	48.99	56.57	52.78	61.27	60.81	61.04
Seg-PN(Zhu et al., 2024)	0.24 M	64.84	67.98	66.41	67.63	71.48	69.56	59.11	60.42	59.77	59.48	64.72	62.10
DyPolySeg	2.64 M	**72.02**	**73.82**	**72.92**	**75.99**	**75.32**	**75.66**	**64.54**	**67.93**	**66.24**	**65.61**	**70.22**	**67.92**
Improvement	-	+7.18	+5.84	+6.51	+8.36	+3.84	+6.10	+5.43	+7.51	+6.47	+6.13	+5.50	+5.82

4.2. Implementation Details

We implement DyPolySeg using PyTorch on an NVIDIA RTX 4090 GPU. For model configuration, we use k-NN with 16 nearest neighbors in DyPolyConv, set max pooling stride to 32 in PCM, and stack three basic blocks combining DyPolyConv and Mamba Block for the encoder. Our experiments focus on N-way K-shot settings ($N \in \{2, 3\}$, $K \in \{1, 5\}$) with 100 test episodes.

For training, we divide dataset classes into seen and unseen subsets, constructing episodes with randomly selected support and query samples. We use cross-entropy loss and AdamW optimizer with learning rate 0.001, halved every 7000 iterations. Data augmentation includes random rotation, translation, and point jittering for improved robustness.

4.3. Comparison with Existing Methods

To evaluate our method, we conduct comparisons with state-of-the-art approaches including DGCNN (Wang et al., 2019), ProtoNet (Snell et al., 2017), MPTI (Zhao et al., 2021b), AttMPTI (Zhao et al., 2021b), BFG (Mao et al., 2022), 2CBR (Zhu et al., 2023), PAP3D (He et al., 2023), and Seg-PN (Zhu et al., 2024).

As shown in Table 1, experimental results on the S3DIS dataset demonstrate the significant performance advantages of our proposed DyPolySeg across various few-shot settings. In the 2-way 1-shot scenario, our method achieves 72.92% mIoU, substantially outperforming the previous state-of-the-art method Seg-PN (66.41%) by 6.51%. The performance gains become more pronounced in the 5-shot settings, where DyPolySeg achieves 75.66% mIoU in the 2-way scenario, surpassing Seg-PN (69.56%) by 6.10%. In the more challenging 3-way settings, our method continues to demonstrate exceptional performance - achieving 66.24% mIoU in the 1-shot scenario and 67.92% in the 5-shot setting, outperforming Seg-PN by significant margins of 6.47% and 5.82% respectively. These consistent improvements validate the effectiveness of our proposed modules.

Results on ScanNet Dataset. As shown in Table 2, experimental results on ScanNet dataset further validate the exceptional performance of DyPolySeg. In the 2-way 1-shot setting, our method achieves 71.89% mIoU, significantly outperforming Seg-PN (63.74%) by 8.15%. This improvement demonstrates our method's robust capability with limited labeled samples. In the 2-way 5-shot scenario, DyPolySeg achieves 72.46% mIoU, surpassing Seg-PN (68.07%) by 4.39%. The performance remains strong in more challenging 3-way settings, where DyPolySeg reaches 69.45% mIoU in 1-shot and 69.18% in 5-shot configurations, exceeding Seg-PN by 5.87% and 3.58% respectively. These consistent improvements across different settings further validate the effectiveness of our approach.

4.4. Ablation experiments

4.4.1. Impact of Different Modules

Table 3 demonstrates the critical contributions of each module in our DyPolySeg framework. The model with only LoConv achieves a modest average mIoU of 48.68%, while introducing DyHoConv improves performance to 50.25%, highlighting its potential in capturing local geometric features. Adding the Mamba Block further enhances performance to 53.28% mIoU, showing its effectiveness in capturing global context. Notably, incorporating PCM dramatically boosts performance to 70.48% mIoU, revealing its crucial role in bridging semantic information gaps between support and query sets. The optimal performance of 72.92% mIoU is achieved when all modules are combined, validating the synergistic effect of our complete architecture.

4.4.2. Impact of Different Numbers of Encoder Layers

As shown in Figure 3, while Seg-NN and Seg-PN show modest improvements with increasing depth, DyPolySeg

Table 2. Few-shot Results (%) on ScanNet. S_i denotes the split i is used for testing. *Avg* is their average mIoU.

Methods	Param.	2-way						3-way					
		1-shot			5-shot			1-shot			5-shot		
		S_0	S_1	*Avg*	S_0	S_1	*Avg*	S_0	S_1	*Avg*	S_0	S_1	*Avg*
DGCNN (Wang et al., 2019)	1.43 M	31.55	28.94	30.25	42.71	37.24	39.98	23.99	19.10	21.55	34.93	28.10	31.52
ProtoNet (Snell et al., 2017)	0.27 M	33.92	30.95	32.44	45.34	42.01	43.68	28.47	26.13	27.30	37.36	34.98	36.17
MPTI (Zhao et al., 2021b)	0.29 M	39.27	36.14	37.71	46.90	43.59	45.25	29.96	27.26	28.61	38.14	34.36	36.25
AttMPTI (Zhao et al., 2021b)	0.37 M	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 (Mao et al., 2022)	-	42.15	40.52	41.34	51.23	49.39	50.31	34.12	31.98	33.05	46.25	41.38	43.82
2CBR (Zhu et al., 2023)	0.35 M	50.73	47.66	49.20	52.35	47.14	49.75	47.00	46.36	46.68	45.06	39.47	42.27
PAP3D (He et al., 2023)	2.45 M	57.08	55.94	56.51	64.55	59.64	62.10	55.27	55.60	55.44	59.02	53.16	56.09
Seg-PN (Zhu et al., 2024)	0.24 M	63.15	64.32	63.74	67.08	69.05	68.07	61.80	65.34	63.57	62.94	68.26	65.60
DyPolySeg	2.64 M	**71.05**	**72.73**	**71.89**	**71.25**	**73.66**	**72.46**	**67.65**	**71.24**	**69.45**	**68.73**	**69.62**	**69.18**
Improvement	-	+7.90	+8.41	+8.15	+4.17	+4.61	+4.39	+5.85	+5.90	+5.87	+5.79	+1.36	+3.58

Table 3. Effect of different modules on S3DIS under 2-way-1-shot settings on the S_0 and S_1 split.

LoConv	DyHoConv	Mamba	PCM	S_0	S_1	*Avg*
✓	✗	✗	✗	47.32	50.05	48.68
✗	✓	✗	✗	48.64	51.86	50.25
✓	✓	✗	✗	49.75	52.97	51.36
✓	✓	✓	✗	52.21	54.35	53.28
✓	✓	✗	✓	70.12	70.84	70.48
✓	✓	✓	✓	**72.02**	**73.82**	**72.92**

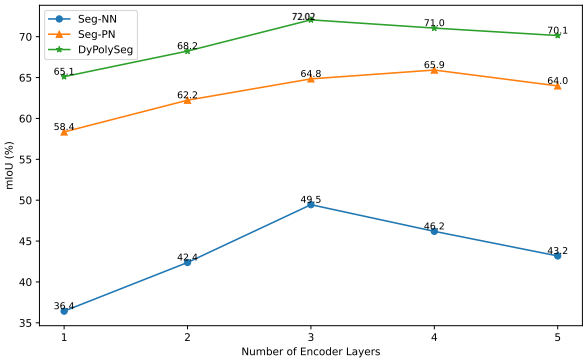


Figure 3. Performance comparison with varying encoder layers under 2-way-1-shot setting on S3DIS dataset.

demonstrates substantial and consistent performance gains. Starting from 36.4% mIoU with one layer, our method progressively improves to 58.4%, 65.1%, and peaks at 72.02% with four encoder layers. The performance plateaus beyond four layers, suggesting an optimal balance between model capacity and complexity.

4.4.3. Ablation Study of DyHoConv

As shown in Table 4, our model demonstrates consistent improvement with increasing number of HoConvs. Starting from a single convolution (69.33% mIoU in 2-way, 63.79% in 3-way), we observe progressive gains: a 0.44% increase

Table 4. Effect of increasing HoConv numbers in DyHoConv module. Results (%) are reported on S3DIS dataset

Number	2-way-1-shot			3-way-1-shot		
	S_0	S_1	*Avg*	S_0	S_1	*Avg*
1	68.76	69.89	69.33	62.11	65.46	63.79
2	69.47	70.07	69.77	62.66	65.95	64.31
4	70.28	71.13	70.71	63.45	66.54	65.00
6	71.14	72.06	71.60	64.01	67.17	65.59
8	**72.02**	**73.82**	**72.92**	**64.54**	**67.93**	**66.24**

Table 5. Effect of different composition of PCM on S3DIS under 2/3-way-1-shot settings on the S_0 and S_1 split.

SEM	IEM	2-way-1-shot			3-way-1-shot		
		S_0	S_1	*Avg*	S_0	S_1	*Avg*
✗	✗	49.17	52.32	50.75	43.34	46.12	44.73
✓	✗	70.15	71.03	70.59	63.67	68.42	66.05
✗	✓	68.44	71.66	70.05	62.74	67.35	65.05
✓	✓	**72.02**	**73.82**	**72.92**	**64.54**	**67.93**	**66.24**

with two convolutions, a 1.38% improvement with four convolutions, and reaching 71.60% mIoU with six convolutions in the 2-way setting. The optimal performance is achieved with eight HoConvs, yielding 72.92% mIoU for 2-way and 66.24% for 3-way settings. This trend indicates that additional HoConvs enhance the capture of local geometric features, while the diminishing returns suggest eight layers as an optimal configuration.

4.4.4. Ablation Study of PCM

As shown in Table 5, removing both SEM and IEM modules causes significant performance degradation, with mIoU dropping to 50.75% and 44.73% for 2-way and 3-way settings respectively. Each individual module demonstrates substantial effectiveness: SEM alone achieves 70.59% and 66.05% mIoU, while IEM alone reaches 70.05% and 65.05% mIoU in 2-way and 3-way settings. The combination of both

Table 6. Effect of the number of PCM. We report the results (%) under 2/3-way-1-shot settings on S3DIS datasets.

Number	2-way-1-shot			3-way-1-shot		
	S_0	S_1	Avg	S_0	S_1	Avg
0	49.17	52.32	50.75	43.34	46.12	44.73
1	68.42	70.18	69.30	61.47	66.89	64.18
2	**72.02**	**73.82**	**72.92**	**64.54**	**67.93**	**66.24**
3	70.02	71.24	70.63	64.32	67.24	65.78

modules yields optimal performance with 72.92% mIoU in 2-way and 66.24% in 3-way settings, validating the synergistic effect of our SEM and IEM design.

4.4.5. Impact of Number of PCM

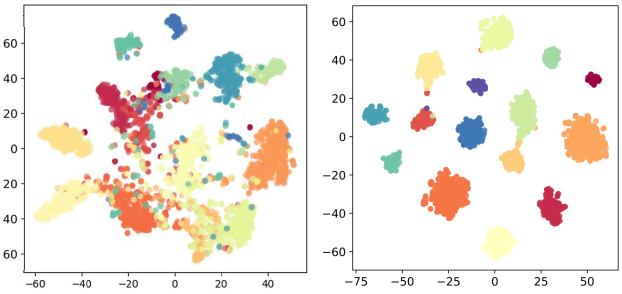
As shown in Table 6, without any PCM, the model's performance is significantly low, with an average mIoU of 50.75% in 2-way and 44.73% in 3-way settings. Introducing a single PCM dramatically improves performance to 69.30% and 64.18%, respectively. The optimal performance is achieved with 2 PCM layers, reaching 72.92% mIoU in the 2-way setting and 66.24% in the 3-way setting. Interestingly, adding a third PCM slightly reduces performance to 70.63% and 65.78%, suggesting that two layers represent the optimal balance for prototype refinement. This indicates that while initial prototype completion significantly enhances feature representation, excessive stacking may introduce unnecessary complexity or feature redundancy.

Table 7. The accuracy (%) of DyPolySeg on ScanObjectNN. "-" means unknown.

Methods	Venue	Accuracy (%)
PointNet (Qi et al., 2017a)	CVPR2017	68.2
PointNet++ (Qi et al., 2017b)	NIPS2017	77.9
DGCNN (Wang et al., 2019)	ACM TOG2019	78.1
Point-BERT (Yu et al., 2022)	CVPR2022	83.1
Point-MAE (Pang et al., 2022)	ECCV2022	85.2
PointMLP (Ma et al., 2022)	ICLR2022	85.4
RespSurf-U (Ran et al., 2022)	CVPR2022	86.0
PointNeXt-S (Qian et al., 2022)	NIPS2022	87.7
Point-PN (Zhang et al., 2023c)	CVPR2023	87.1
PCM (Zhang et al., 2024c)	AAAI2025	88.1
PointMamba (Liang et al., 2024b)	NeurIPS2024	89.3
DyPolySeg (**our**)	ICML2025	**90.8**

4.5. Other Point Cloud Understanding Tasks

As shown in Table 7, DyPolySeg achieves 90.8% accuracy on ScanObjectNN with only 2.64M parameters, demonstrating competitive performance against recent methods like PCM (34.2M) and Mamba3D (16.9M). This efficiency validates the effectiveness of our architecture design.



(a) Point-BERT (b) DyPolySeg

Figure 4. t-SNE visualization comparing the feature spaces of Point-BERT and DyPolySeg on the ScanObjectNN.

Visualization in Figure 3 compares feature distributions between Point-BERT (Yu et al., 2022) and DyPolySeg on ScanObjectNN dataset. Our method achieves more compact within-class clustering and clearer between-class separation, indicating superior shape-aware representation learning.

5. Conclusion

In this paper, we present DyPolySeg, a novel framework for few-shot point cloud semantic segmentation. Our method addresses two critical challenges through: (1) a dynamic polynomial fitting network with DyPolyConv for effective local geometric modeling, and (2) a lightweight Prototype Completion Module (PCM) for enhanced prototype representation. The integration of DyPolyConv and Mamba Block in an encoder-decoder architecture enables comprehensive capture of both local geometric details and global semantic context. Extensive experiments demonstrate that DyPolySeg not only advances the state-of-the-art in few-shot point cloud segmentation but also provides valuable insights for future 3D vision research.

Impact Statement

The proposed DyPolySeg framework advances few-shot point cloud semantic segmentation, contributing to robust and efficient 3D scene understanding. This technology has broad applications in autonomous navigation, assistive robotics, and augmented reality, potentially accelerating their deployment for societal benefit.

While promising, we acknowledge potential privacy concerns regarding point cloud data usage and the need to address potential biases in training data that could affect different populations. We encourage the research community to proactively consider these broader impacts in future developments.

References

An, Z., Sun, G., Liu, Y., Liu, F., Wu, Z., Wang, D., Van Gool, L., and Belongie, S. Rethinking few-shot 3d point cloud semantic segmentation. In *CVPR*, pp. 3996–4006, 2024.

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

Chen, G., Wang, M., Yang, Y., Yu, K., Yuan, L., and Yue, Y. Pointgpt: Auto-regressively generative pre-training from point clouds. *Advances in Neural Information Processing Systems*, 36, 2024.

Chib, P. S. and Singh, P. Recent advancements in end-to-end autonomous driving using deep learning: A survey. *IEEE Transactions on Intelligent Vehicles*, 2023.

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

Devagiri, J. S., Paheding, S., Niyaz, Q., Yang, X., and Smith, S. Augmented reality and artificial intelligence in industry: Trends, tools, and future challenges. *Expert Systems with Applications*, 207:118002, 2022.

Fang, X., Fang, W., Wang, C., Liu, D., Tang, K., Dong, J., Zhou, P., and Li, B. Multi-pair temporal sentence grounding via multi-thread knowledge transfer network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 2915–2923, 2025.

Goel, R. and Gupta, P. Robotics and industry 4.0. *A Roadmap to Industry 4.0: Smart Production, Sharp Business and Sustainable Development*, pp. 157–169, 2020.

Han, X., Tang, Y., Wang, Z., and Li, X. Mamba3d: Enhancing local features for 3d point cloud analysis via state space model. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 4995–5004, 2024.

He, S. and Ding, H. Refmask3d: Language-guided transformer for 3d referring segmentation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 8316–8325, 2024.

He, S., Jiang, X., Jiang, W., and Ding, H. Prototype adaption and projection for few-and zero-shot 3d point cloud semantic segmentation. *TIP*, 32:3199–3211, 2023.

He, S., Ding, H., Jiang, X., and Wen, B. Segpoint: Segment any point cloud via large language model. In *European Conference on Computer Vision*, pp. 349–367. Springer, 2024.

Jiang, L., Wang, C., Ning, X., and Yu, Z. Lttpoint: a mlp-based point cloud classification method with local topology transformation module. In *2023 7th Asian Conference on Artificial Intelligence Technology (ACAIT)*, pp. 783–789. IEEE, 2023.

Lai, L., Chen, J., Zhang, C., Zhang, Z., Lin, G., and Wu, Q. Tackling background ambiguities in multi-class few-shot point cloud semantic segmentation. *Knowledge-Based Systems*, 253:109508, 2022a.

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

Li, Z., Wang, Y., Li, W., Sun, R., and Zhang, T. Localization and expansion: A decoupled framework for point cloud few-shot semantic segmentation. *arXiv preprint arXiv:2408.13752*, 2024.

Liang, D., Feng, T., Zhou, X., Zhang, Y., Zou, Z., and Bai, X. Parameter-efficient fine-tuning in spectral domain for point cloud learning. *arXiv preprint arXiv:2410.08114*, 2024a.

Liang, D., Zhou, X., Xu, W., Zhu, X., Zou, Z., Ye, X., Tan, X., and Bai, X. Pointmamba: A simple state space model for point cloud analysis. *arXiv preprint arXiv:2402.10739*, 2024b.

Ma, X., Qin, C., You, H., Ran, H., and Fu, Y. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *arXiv preprint arXiv:2202.07123*, 2022.

Mao, Y., Guo, Z., Xiaonan, L., Yuan, Z., and Guo, H. Bidirectional feature globalization for few-shot semantic segmentation of 3d point cloud scenes. In *2022 International Conference on 3D Vision (3DV)*, pp. 505–514. IEEE, 2022.

Moody, J. and Darken, C. J. Fast learning in networks of locally-tuned processing units. *Neural computation*, 1(2): 281–294, 1989.

Ning, E., Zhang, C., Wang, C., Ning, X., Chen, H., and Bai, X. Pedestrian re-id based on feature consistency and contrast enhancement. *Displays*, 79:102467, 2023.

Ning, E., Wang, C., Zhang, H., Ning, X., and Tiwari, P. Occluded person re-identification with deep learning: a survey and perspectives. *Expert systems with applications*, 239:122419, 2024a.

Ning, E., Wang, Y., Wang, C., Zhang, H., and Ning, X. Enhancement, integration, expansion: Activating representation of detailed features for occluded person re-identification. *Neural Networks*, 169:532–541, 2024b.

Pang, Y., Wang, W., Tay, F. E., Liu, W., Tian, Y., and Yuan, L. Masked autoencoders for point cloud self-supervised learning. In *European conference on computer vision*, pp. 604–621. Springer, 2022.

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

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NIPS*, 30, 2017b.

Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H., Elhoseiny, M., and Ghanem, B. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *NIPS*, 35: 23192–23204, 2022.

Ran, H., Liu, J., and Wang, C. Surface representation for point clouds. In *CVPR*, pp. 18942–18952, 2022.

Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Rudin, W. et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1964.

Sereno, M., Wang, X., Besançon, L., Mcguffin, M. J., and Isenberg, T. Collaborative work in augmented reality: A survey. *TVCG*, 28(6):2530–2549, 2020.

Shi, J., Zhang, Y., Yin, X., Xie, Y., Zhang, Z., Fan, J., Shi, Z., and Qu, Y. Dual pseudo-labels interactive self-training for semi-supervised visible-infrared person re-identification. In *ICCV*, pp. 11218–11228, 2023.

Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. *NIPS*, 30, 2017.

Soori, M., Arezoo, B., and Dastres, R. Artificial intelligence, machine learning and deep learning in advanced robotics, a review. *Cognitive Robotics*, 3:54–70, 2023.

Uy, M. A., Pham, Q.-H., Hua, B.-S., Nguyen, T., and Yeung, S.-K. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *ICCV*, pp. 1588–1597, 2019.

Wang, C., Wang, C., Li, W., and Wang, H. A brief survey on rgb-d semantic segmentation using deep learning. *Displays*, 70:102080, 2021.

Wang, C., Ning, X., Sun, L., Zhang, L., Li, W., and Bai, X. Learning discriminative features by covering local geometric space for point cloud analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022a.

Wang, C., Wang, H., Ning, X., Tian, S., and Li, W. 3d point cloud classification method based on dynamic coverage of local area. *J. Softw*, 34(4):1962–1976, 2022b.

Wang, C., Ning, X., Li, W., Bai, X., and Gao, X. 3d person re-identification based on global semantic guidance and local feature aggregation. *IEEE transactions on circuits and systems for video technology*, 34(6):4698–4712, 2023.

Wang, C., Wu, M., Lam, S.-K., Ning, X., Yu, S., Wang, R., Li, W., and Srikanthan, T. Gpsformer: A global perception and local structure fitting-based transformer for point cloud understanding. In *European Conference on Computer Vision*, pp. 75–92. Springer, 2024.

Wang, C., Cao, R., and Wang, R. Learning discriminative topological structure information representation for 2d shape and social network classification via persistent homology. *Knowledge-Based Systems*, pp. 113125, 2025a.

Wang, C., He, S., Fang, X., Wu, M., Lam, S.-K., and Tiwari, P. Taylor series-inspired local structure fitting network for few-shot point cloud semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7527–7535, 2025b.

Wang, C., He, S., Wu, M., Lam, S.-K., Tiwari, P., and Gao, X. Looking clearer with text: A hierarchical context blending network for occluded person re-identification. *IEEE Transactions on Information Forensics and Security*, 2025c.

Wang, R., Lam, S.-K., Wu, M., Hu, Z., Wang, C., and Wang, J. Destination intention estimation-based convolutional encoder-decoder for pedestrian trajectory multimodality forecast. *Measurement*, 239:115470, 2025d.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38 (5):1–12, 2019.

Wei, L., Lang, C., Chen, Z., Wang, T., Li, Y., and Liu, J. Generated and pseudo content guided prototype refinement for few-shot point cloud segmentation. In *NIPS*, 2024.

Wu, X., Lao, Y., Jiang, L., Liu, X., and Zhao, H. Point transformer v2: Grouped vector attention and partition-based pooling. *NIPS*, 35:33330–33342, 2022.

Xiong, G., Wang, Y., Li, Z., Yang, W., Zhang, T., Zhou, X., Zhang, S., and Zhang, Y. Aggregation and purification: Dual enhancement network for point cloud few-shot segmentation. In *IJCAI*, 2024.

Xu, J., Yang, S., Li, X., Tang, Y., Hao, Y., Hu, L., and Chen, M. A probability-driven framework for open world 3d point cloud semantic segmentation. In *CVPR*, pp. 5977–5986, 2024.

Yang, B., Bender, G., Le, Q. V., and Ngiam, J. Condconv: Conditionally parameterized convolutions for efficient inference. *NIPS*, 32, 2019.

Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., and Lu, J. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *CVPR*, pp. 19313–19322, 2022.

Yu, Z., Li, L., Xie, J., Wang, C., Li, W., and Ning, X. Pedestrian 3d shape understanding for person re-identification via multi-view learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(7):5589–5602, 2024.

Zhang, C., Wu, Z., Wu, X., Zhao, Z., and Wang, S. Few-shot 3d point cloud semantic segmentation via stratified class-specific attention based transformer network. In *AAAI*, volume 37, pp. 3410–3417, 2023a.

Zhang, H., Wang, C., Tian, S., Lu, B., Zhang, L., Ning, X., and Bai, X. Deep learning-based 3d point cloud classification: A systematic survey and outlook. *Displays*, 79:102456, 2023b.

Zhang, H., Ning, X., Wang, C., Ning, E., and Li, L. Deformation depth decoupling network for point cloud domain adaptation. *Neural Networks*, 180:106626, 2024a.

Zhang, H., Wang, C., Yu, L., Tian, S., Ning, X., and Rodrigues, J. Pointgt: A method for point-cloud classification and segmentation based on local geometric transformation. *IEEE Transactions on Multimedia*, 2024b.

Zhang, R., Wang, L., Guo, Z., Wang, Y., Gao, P., Li, H., and Shi, J. Parameter is not all you need: Starting from non-parametric networks for 3d point cloud analysis. *arXiv preprint arXiv:2303.08134*, 2023c.

Zhang, T., Li, X., Yuan, H., Ji, S., and Yan, S. Point could mamba: Point cloud learning via state space model. *arXiv preprint arXiv:2403.00762*, 2024c.

Zhao, H., Jiang, L., Jia, J., Torr, P. H., and Koltun, V. Point transformer. In *ICCV*, pp. 16259–16268, 2021a.

Zhao, J., Zhao, W., Deng, B., Wang, Z., Zhang, F., Zheng, W., Cao, W., Nan, J., Lian, Y., and Burke, A. F. Autonomous driving system: A comprehensive survey. *Expert Systems with Applications*, pp. 122836, 2023.

Zhao, N., Chua, T.-S., and Lee, G. H. Few-shot 3d point cloud semantic segmentation. In *CVPR*, pp. 8873–8882, 2021b.

Zhou, X., Liang, D., Xu, W., Zhu, X., Xu, Y., Zou, Z., and Bai, X. Dynamic adapter meets prompt tuning: Parameter-efficient transfer learning for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14707–14717, 2024.

Zhu, G., Zhou, Y., Yao, R., and Zhu, H. Cross-class bias rectification for point cloud few-shot segmentation. *TMM*, 25:9175–9188, 2023.

Zhu, X., Zhang, R., He, B., Guo, Z., Liu, J., Xiao, H., Fu, C., Dong, H., and Gao, P. No time to train: Empowering non-parametric networks for few-shot 3d scene segmentation. In *CVPR*, pp. 3838–3847, 2024.