# CascadeProto: Cascaded Cross-Modal Prototype Purification via Entropy-Aware Learning for Few-Shot 3D Point Cloud Segmentation

Changshuo Wang[1], Weijun Li[2]✉, Fan Mo[3], Zhonghang Liu[4], Shuting He[5], Prayag Tiwari[6], and Dimitrios Kanoulas[1]✉

[1] Department of Computer Science, University College London, London, UK
[2] Institute of Semiconductors, Chinese Academy of Sciences, Beijing, China
[3] School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu, China
[4] SmartMore Corporation Ltd., Shenzhen, China
[5] School of Computing and Artificial Intelligence, Shanghai University of Finance and Economics, Shanghai, China
[6] School of Information Technology, Halmstad University, Sweden
wangchangshuo1@gmail.com; wjli@semi.ac.cn; d.kanoulas@ucl.ac.uk

**Abstract.** Few-shot 3D point cloud semantic segmentation aims to recognize novel object categories with limited labeled examples. Existing methods typically rely on point-based prototypes extracted from support sets, but such prototypes inevitably contain background noise and lack semantic richness, leading to suboptimal segmentation quality. Inspired by chemical distillation that iteratively removes impurities, we propose **CascadeProto**, which employs multi-step refinement to distill high-quality prototypes from noisy initializations. At its core lies the **E**ntropy-aware **P**rototype **P**urification **M**odule (**EPPM**), which leverages information-theoretic principles to suppress high-entropy background features while amplifying low-entropy foreground information. To further enrich prototype representations, we introduce **L**earnable **M**odality **A**dapters (**LMA**) that independently align each of three CLIP modalities — text, audio, and image — with point cloud features through foreground-background decoupled distribution matching, enabling flexible single-modality semantic enrichment that bridges the 2D-3D domain gap. Furthermore, we propose an **A**ttention-based **D**ynamic **R**outing **M**echanism (**ADRM**) that adaptively aggregates predictions from multiple cascade stages, allowing simple regions to benefit from early-stage outputs while complex regions leverage deeper purification. Extensive experiments on S3DIS and ScanNet benchmarks demonstrate the effectiveness and superiority of CascadeProto across all three modalities over state-of-the-art methods. The code is available at `https://github.com/changshuowang/CascadeProto`.

**Keywords:** Point Cloud Segmentation · Few-Shot Learning · Cross-Modal Fusion · Prototype Purification · Entropy-Aware Learning

✉ Corresponding authors.

**Table 1:** Comparison of few-shot point cloud semantic segmentation methods. Green checkmarks (✓) indicate supported or used capabilities, while red crosses (✗) denote unsupported or unused features. "Text", "Audio", "Image", and "Point" refer to the modalities available for prototype generation.

| Method | Pre-training | Modality | | | | Backbone |
|---|---|---|---|---|---|---|
| | | Text | Audio | Image | Point | |
| AttMPTI [34] | ✓ | ✗ | ✗ | ✗ | ✓ | DGCNN |
| PAP3D [8] | ✓ | ✓ | ✗ | ✗ | ✓ | DGCNN |
| SegPN [36] | ✗ | ✗ | ✗ | ✗ | ✓ | SegPN |
| TaylorSeg [25] | ✗ | ✗ | ✗ | ✗ | ✓ | Improved SegPN |
| DyPolySeg [21] | ✗ | ✗ | ✗ | ✗ | ✓ | DyPolySeg |
| VIP-Seg [23] | ✗ | ✗ | ✗ | ✗ | ✓ | VIP-Seg |
| EDS-Net [26] | ✗ | ✓ | ✗ | ✗ | ✓ | VIP-Seg |
| DPR-Net [24] | ✗ | ✗ | ✗ | ✗ | ✓ | VIP-Seg |
| **CascadeProto (Ours)** | ✗ | ✓ | ✓ | ✓ | ✓ | VIP-Seg |

## 1 Introduction

Point cloud semantic segmentation [22] serves as a fundamental task in 3D scene understanding, with broad applications in autonomous driving [4, 33], robotics [7, 20], and augmented reality [6, 18]. Despite remarkable progress in fully-supervised methods [28, 32], acquiring large-scale point-level annotations in 3D space remains prohibitively expensive and time-consuming. This fundamental limitation has motivated the development of few-shot learning paradigms, where models must generalize to novel categories with only a handful of labeled examples.

Few-shot point cloud semantic segmentation typically extracts prototypes from a small support set and matches query points against these prototypes. Recent works [21, 25, 34, 36] have demonstrated promising results by leveraging meta-learning strategies and attention mechanisms. However, these methods face three critical challenges. **First**, initial prototypes extracted from sparse support sets inevitably contain background clutter, as raw feature averages indiscriminately mix foreground objects with surrounding noise, leading to degraded segmentation boundaries. **Second**, many competitive approaches rely on large-scale pre-training [8, 34], which incurs substantial computational overhead and may introduce domain biases inherited from pre-training datasets. **Third**, existing frameworks exploit only point cloud geometry, failing to leverage the rich cross-modal semantic knowledge embedded in vision-language models such as CLIP [17], which could substantially enrich prototype representations.

Our key insight is that high-quality prototypes can be progressively distilled from noisy initializations through cascaded purification, drawing inspiration from the chemical distillation process that iteratively removes impurities. Background features exhibit high Shannon entropy due to their diverse and cluttered nature, while foreground features of specific object categories maintain consistently low

entropy. This observation motivates an entropy-driven purification mechanism that gradually suppresses high-entropy background noise across multiple refinement stages, yielding increasingly discriminative prototypes at each step.

As shown in Table 1, existing methods [8,34] that rely on pre-training impose significant computational overhead, while non-pre-training methods [21, 23–26, 36] remain confined to point cloud geometry alone. In contrast, CascadeProto requires no pre-training yet supports three independent cross-modal inputs — text, audio, and image — each enriching point cloud prototypes through a dedicated modality-specific adapter, and employs cascaded entropy-aware purification to progressively refine prototype quality.

We propose **CascadeProto**, a framework that bridges geometric and semantic knowledge through cascaded cross-modal prototype purification for few-shot 3D point cloud segmentation. CascadeProto consists of three key components: (1) **Learnable Modality Adapters (LMA)** that independently align each CLIP modality — text, audio, or image — with point cloud features via foreground-background decoupled distribution matching using a GMMN-based strategy, effectively bridging the 2D-3D domain gap without any pre-training. (2) **Entropy-aware Prototype Purification Module (EPPM)**, the core purification unit that leverages Shannon entropy to progressively suppress high-entropy background while enhancing low-entropy foreground features across multiple cascade stages. (3) **Attention-based Dynamic Routing Mechanism (ADRM)** that adaptively aggregates predictions from all cascade stages via query-conditioned gating weights, enabling simple regions to benefit from early-stage outputs while complex regions leverage deeper purification.

The main contributions of this work are summarized as follows:

- We propose **CascadeProto**, a novel pre-training-free framework for few-shot 3D point cloud segmentation that supports flexible single-modality semantic enrichment — independently from text, audio, or image — through cascaded cross-modal prototype purification, achieving superior performance without any pre-trained weights.
- We introduce the **EPPM**, which leverages information-theoretic principles to progressively distill clean prototypes from noisy initializations by suppressing high-entropy background features and amplifying low-entropy foreground information across multiple cascade stages.
- We design **LMA** with modality-specific adapters and foreground-background decoupled distribution matching, enabling each of text, audio, and image CLIP embeddings to independently enrich 3D point cloud prototype representations without requiring end-to-end pre-training.
- We propose an **ADRM** that adaptively aggregates multi-stage predictions based on per-region complexity, allowing the model to exploit coarse early-stage prototypes for simple regions and fine-grained late-stage prototypes for cluttered scenes, improving overall robustness and segmentation accuracy.

## 2 Related Work

### 2.1 Point Cloud Semantic Segmentation

3D point cloud semantic segmentation assigns a semantic label to each point in an unstructured set, forming the perceptual backbone of many real-world systems [27, 29]. The seminal PointNet [15] pioneered end-to-end processing of raw point sets via shared MLPs and a global max-pooling aggregator, though its lack of local context limits fine-grained geometric reasoning. PointNet++ [16] addressed this by recursively partitioning the point set into nested local regions, enabling hierarchical feature learning at multiple scales. DGCNN [28] took a graph-based perspective, constructing $k$-NN graphs on-the-fly and introducing EdgeConv to capture inter-point relational structure, which has since established it as a widely adopted encoder backbone.

Subsequent architectures have advanced along three complementary directions. Transformer-based models such as Point Transformer [32] apply self-attention to capture long-range dependencies across the full point set. State-space-model approaches such as PointMamba [11] offer sub-quadratic sequence modeling, improving scalability for large outdoor scenes. Self-supervised methods including PointMAE [14] and PointGPT [3] adopt masked reconstruction objectives to learn transferable representations from unlabeled data, substantially reducing the annotation burden for downstream tasks. Cross-modal methods such as PointCLIP [31] further enrich 3D representations by grounding them in vision-language knowledge from CLIP. Despite impressive progress, all of these fully-supervised paradigms require abundant annotated data, rendering them fundamentally ill-suited for few-shot scenarios where novel categories must be recognized from only a handful of labeled examples.

### 2.2 Few-Shot Point Cloud Semantic Segmentation

Few-shot point cloud semantic segmentation (FS-PCSS) aims to generalize to unseen categories under strict annotation budgets, and is typically formulated as an $N$-way $K$-shot episodic learning problem [8, 13, 34]. The dominant paradigm extracts per-category prototypes from labeled support sets and classifies query points by proximity to these prototypes in feature space. AttMPTI [34] established this direction by combining pre-trained DGCNN features with multi-prototype generation and transductive label propagation. BFG [13] introduced bi-directional globalization to exchange contextual information between support and query branches, improving prototype discriminability. 2CBR [35] exploited cross-branch feature correlation to reduce the support-query distribution gap, while DENet [30] tackled intra-class variation through bi-directional feature aggregation. COSeg [1] rethought the evaluation protocol by expanding the input point set to mitigate information leakage in block-based pipelines.

More recently, the field has shifted toward eliminating pre-training dependencies. SegPN [36] and TaylorSeg [25] deliberately forgo large-scale pre-training to remove domain bias, demonstrating that strong performance is achievable

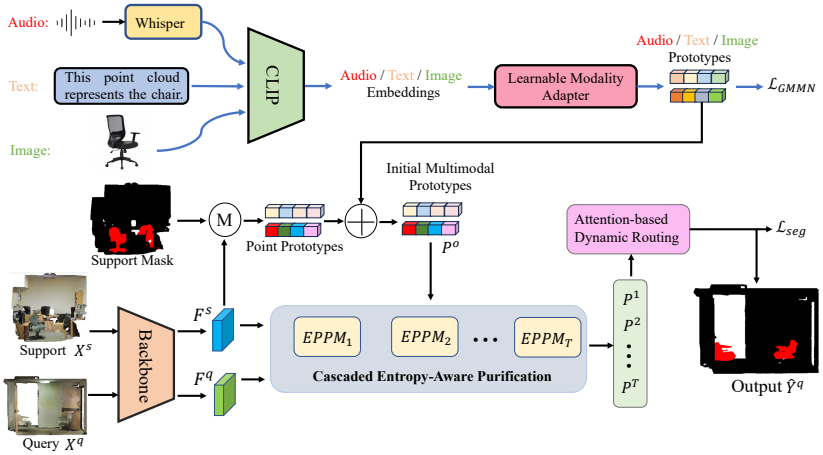


**Fig. 1: Overall architecture of CascadeProto.** Given a support set $\mathcal{S}$ with binary masks and a query point cloud $\mathbf{X}^q$, a shared VIP-Seg encoder extracts per-point features $\mathbf{F}^s$ and $\mathbf{F}^q$. Initial point prototypes $\mathbf{P}_{\text{point}}$ are computed via masked average pooling, while a Learnable Modality Adapter (LMA) projects a single CLIP modality embedding — text, audio, or image — into the point cloud feature space via foreground-background decoupled distribution matching, yielding enriched initial prototypes $\mathbf{P}^0 \in \mathbb{R}^{(N+1)\times D}$. These are then passed through $T$=4 cascaded Entropy-aware Prototype Purification Modules (EPPM), each of which suppresses high-entropy background features via information-theoretic gating and refines the prototype through cross-attention and prototype diffusion, producing a purification trajectory $\{\mathbf{P}^1, \mathbf{P}^2, \mathbf{P}^3, \mathbf{P}^4\}$ with corresponding per-stage logits $\{\mathbf{L}^1, \mathbf{L}^2, \mathbf{L}^3, \mathbf{L}^4\}$. Finally, the Attention-based Dynamic Routing Mechanism (ADRM) adaptively aggregates all stage logits via query-conditioned gating weights to produce the final segmentation output $\hat{\mathbf{Y}}^q$.

from scratch. DyPolySeg [21] combined dynamic polynomial convolution with Mamba-style state-space modeling to capture complementary local and global geometric cues. VIP-Seg [23] further strengthened geometric representation by incorporating visual-inductive priors directly into the encoder. Despite steady progress, all of these methods share a fundamental bottleneck: prototype quality is constrained by the geometric information in point clouds alone. Single-round or shallow fusion strategies leave substantial background noise in the extracted prototypes, and the rich semantic knowledge encoded in vision-language models remains entirely unexploited. CascadeProto directly addresses both limitations through cascaded entropy-aware purification and flexible single-modality cross-modal enrichment.

## 3 Methodology

In this section, we first define the problem formulation of few-shot 3D point cloud semantic segmentation, then present the overall architecture of CascadeProto.

We subsequently describe each core component in detail: the Learnable Modality Adapter (LMA), the Entropy-aware Prototype Purification Module (EPPM), the cascaded purification process, and the Attention-based Dynamic Routing Mechanism (ADRM). Finally, we present the training objectives and inference strategy.

### 3.1 Problem Formulation

Few-shot 3D point cloud semantic segmentation adopts the episodic learning paradigm [34], partitioning categories into seen classes $\mathcal{C}_{\text{seen}}$ used for training and unseen classes $\mathcal{C}_{\text{unseen}}$ reserved for evaluation. Each episode is formulated as an $N$-way $K$-shot task comprising a support set $\mathcal{S}$ and a query set $\mathcal{Q}$.

Formally, the support set $\mathcal{S} = \{(\mathbf{X}_i^s, \mathbf{Y}_i^s)\}_{i=1}^{N\times K}$ contains $N$ categories with $K$ labeled samples per category, where $\mathbf{X}_i^s \in \mathbb{R}^{N_s\times 3}$ denotes a point cloud of $N_s$ points and $\mathbf{Y}_i^s \in \{0,1\}^{N_s}$ is the corresponding binary foreground mask. The query set $\mathcal{Q} = \{\mathbf{X}^q, \mathbf{Y}^q\}$ contains an unlabeled point cloud $\mathbf{X}^q \in \mathbb{R}^{N_q\times 3}$ to be segmented into $N$ target categories plus background. Our objective is to learn a mapping:

$$\hat{\mathbf{Y}}^q = f_\theta(\mathcal{S},\, \mathcal{M},\, \mathbf{X}^q), \tag{1}$$

where $\hat{\mathbf{Y}}^q \in \mathbb{R}^{N_q\times(N+1)}$ denotes the predicted per-point logits over $N+1$ classes, $\mathcal{M} = \{c_1, \ldots, c_K\}$ represents optional cross-modal semantic descriptors (text, image, audio), and $f_\theta$ is CascadeProto with parameters $\theta$.

### 3.2 Overall Architecture

As illustrated in Figure 1, CascadeProto processes an episode through three sequential stages. **Stage 1: Multi-Modal Prototype Generation.** A shared VIP-Seg encoder extracts point-wise features from support and query inputs. Initial prototypes $\mathbf{P}_{\text{point}}$ are obtained via masked average pooling, while Learnable Modality Adapters (LMA) project CLIP text and image embeddings into the point cloud feature space. A GMMN-based distribution matching module then fuses both sources into enriched initial prototypes $\mathbf{P}^0 \in \mathbb{R}^{(N+1)\times D}$. **Stage 2: Cascaded Entropy-Aware Purification.** The initial prototypes $\mathbf{P}^0$ are fed into a cascade of $T$ EPPM modules. Each module suppresses high-entropy background features and enhances low-entropy foreground information, producing progressively purer prototypes $\{\mathbf{P}^1, \mathbf{P}^2, \ldots, \mathbf{P}^T\}$ and corresponding per-step logits $\{\mathbf{L}^1, \mathbf{L}^2, \ldots, \mathbf{L}^T\}$. **Stage 3 — Attention-based Dynamic Routing.** The ADRM aggregates logits from all cascade stages via query-conditioned gating weights, producing the final prediction $\hat{\mathbf{Y}}^q$.

### 3.3 Multi-Modal Prototype Generation

**Point Cloud Feature Extraction** Given an input point cloud $\mathbf{X} \in \mathbb{R}^{N_s\times 3}$ (support) or $\mathbf{X} \in \mathbb{R}^{N_q\times 3}$ (query), the shared VIP-Seg encoder $f_{\text{enc}}$ extracts per-point features:

$$\mathbf{F} = f_{\text{enc}}(\mathbf{X}) \in \mathbb{R}^{N_s\times D} \text{ or } \mathbb{R}^{N_q\times D}, \tag{2}$$

where $D = 128$ is the feature dimension.

The encoder is shared between support and query branches. Initial foreground and background prototypes are extracted from the support features via masked average pooling:

$$\mathbf{P}_{\text{fg}}^{(k)} = \frac{1}{|\mathcal{M}_{\text{fg}}^{(k)}|} \sum_{i \in \mathcal{M}_{\text{fg}}^{(k)}} \mathbf{F}_i^s, \quad \mathbf{P}_{\text{bg}} = \frac{1}{|\mathcal{M}_{\text{bg}}|} \sum_{i \in \mathcal{M}_{\text{bg}}} \mathbf{F}_i^s, \tag{3}$$

where $\mathcal{M}_{\text{fg}}^{(k)} = \{i \mid Y_i^s = 1,\ \text{class}(i) = k\}$ and $\mathcal{M}_{\text{bg}} = \{i \mid Y_i^s = 0\}$, with $k = 1, \ldots, N$ ranging over the $N$ categories in the current episode. The stacked point prototype is $\mathbf{P}_{\text{point}} = [\mathbf{P}_{\text{bg}};\ \mathbf{P}_{\text{fg}}^{(1)};\ \ldots;\ \mathbf{P}_{\text{fg}}^{(N)}] \in \mathbb{R}^{(N+1)\times D}$.

**Learnable Modality Adapter** CLIP embeddings are pre-trained on 2D image-text pairs and exist in a feature space misaligned with 3D point cloud geometry. To bridge this 2D-3D domain gap, we design a lightweight Learnable Modality Adapter (LMA) — one per modality — that map each CLIP embedding into the point cloud feature space:

$$\mathbf{E}_{\text{adapted}}^{(m)} = \text{Adapter}^{(m)}(\mathbf{E}_{\text{CLIP}}^{(m)}), \quad m \in \{\text{text},\ \text{image},\ \text{audio}\}, \tag{4}$$

where each adapter is a two-layer MLP with LayerNorm and Dropout, and the adapted embedding is:

$$\text{Adapter}(\mathbf{E}) = \mathbf{W}_2 \cdot \text{ReLU}(\text{LN}(\mathbf{W}_1 \cdot \mathbf{E} + \mathbf{b}_1)) + \mathbf{b}_2. \tag{5}$$

**Foreground-Background Decoupled Distribution Matching** A critical observation is that background features are heterogeneous — comprising walls, floors, and miscellaneous clutter — while foreground features for a specific category are semantically coherent. Treating them with the same alignment loss would force the adapter to compromise between two fundamentally different distributions. We therefore apply decoupled GMMN-based alignment [10] separately for foreground and background.

Given adapted cross-modal embeddings $\mathbf{E}_{\text{fused}}$ and random noise $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, a three-layer MLP generator $G$ produces cross-modal prototypes:

$$\mathbf{P}_{\text{modal}} = G(\mathbf{E}_{\text{fused}},\ \mathbf{z}) \in \mathbb{R}^{(N+1)\times D}. \tag{6}$$

The foreground and background alignment losses are defined using the Maximum Mean Discrepancy (MMD) with a multi-scale Gaussian kernel $k_\sigma$:

$$\text{MMD}(P, Q) = \left\| \frac{1}{|P|} \sum_{\mathbf{x} \in P} \phi(\mathbf{x}) - \frac{1}{|Q|} \sum_{\mathbf{y} \in Q} \phi(\mathbf{y}) \right\|_{\mathcal{H}}^2, \tag{7}$$

where $k(\mathbf{x}, \mathbf{y}) = \sum_{\sigma \in \{2,5,10,20,40,80\}} \exp(-\|\mathbf{x} - \mathbf{y}\|^2 / 2\sigma^2)$. The decoupled alignment loss is:

$$\mathcal{L}_{\text{GMMN}} = 0.1 \cdot \underbrace{\text{MMD}(\mathbf{P}_{\text{modal}}^{\text{bg}},\ \mathbf{P}_{\text{point}}^{\text{bg}})}_{\mathcal{L}_{\text{bg}}} + 1.0 \cdot \underbrace{\text{MMD}(\mathbf{P}_{\text{modal}}^{\text{fg}},\ \mathbf{P}_{\text{point}}^{\text{fg}})}_{\mathcal{L}_{\text{fg}}}. \tag{8}$$

The lower weight on $\mathcal{L}_{\text{bg}}$ reflects background heterogeneity: forcing the background cross-modal prototype to match the background point prototype would impede generalization. The fused initial prototype used for training is:

$$\mathbf{P}^0 = \mathbf{P}_{\text{point}} + \mathbf{P}_{\text{modal}}. \tag{9}$$

### 3.4 Entropy-Aware Prototype Purification Module

The EPPM is the core building block of the cascade. It takes as input the current prototype $\mathbf{P}^{t-1}$ along with support features $\mathbf{F}^s$ and query features $\mathbf{F}^q$, and outputs a refined prototype $\mathbf{P}^t$.

**Information-Theoretic Gating** Our key observation is that background features in 3D scenes exhibit high Shannon entropy because they encompass diverse and heterogeneous structures, whereas foreground features of a specific category maintain low entropy due to their semantic consistency. Formally, for a feature vector $\mathbf{x} \in \mathbb{R}^D$, we compute its per-channel entropy after sigmoid normalization:

$$p_i = \sigma(x_i), \quad H_i = -p_i \log(p_i + \epsilon) - (1 - p_i) \log(1 - p_i + \epsilon), \tag{10}$$

where $\epsilon = 10^{-8}$ ensures numerical stability. A learnable threshold $\theta$ (initialized to 0.5) governs a soft gate:

$$g_i = \sigma(2(\theta - H_i)), \tag{11}$$

which assigns values close to 1 for low-entropy (foreground-like) channels and close to 0 for high-entropy (background-like) channels. The gated feature is:

$$\mathbf{x}_{\text{gated}} = \mathbf{x} \odot \mathbf{g}. \tag{12}$$

**Cross-Attention Refinement** After entropy gating, we apply cross-attention between the query and support to further refine the prototype. Query and support features are projected into a low-dimensional space of dimension $d = 72$:

$$\mathbf{Q}' = \varphi(\mathbf{F}^q) \in \mathbb{R}^{N_q \times d}, \quad \mathbf{S}' = \varphi(\mathbf{F}^s) \in \mathbb{R}^{N_s \times d}, \tag{13}$$

where $\varphi$ is a $1 \times 1$ convolution. The cross-correlation attention matrix and cross-attended prototype are:

$$\mathbf{A} = \text{softmax}\left(\frac{\mathbf{Q}'\mathbf{S}'^{\top}}{\sqrt{d}}\right) \in \mathbb{R}^{N_q \times N_s}, \quad \mathbf{P}_{\text{cross}} = \mathbf{A} \cdot \psi(\mathbf{P}^{t-1}), \tag{14}$$

where $\psi$ is a learnable linear projection.

**Prototype Diffusion** To propagate prototype information across both branches, we introduce a prototype diffusion mechanism. Channel-level activations are computed for the query and support:

$$\mathbf{q}_{\text{ch}} = \sigma(\text{mean}(\mathbf{F}^q, \text{dim} = 1)), \quad \mathbf{s}_{\text{ch}} = \sigma(\text{mean}(\mathbf{F}^s, \text{dim} = 1)). \tag{15}$$

Channels active in both branches form the common activation mask $\mathbf{m}_{\text{common}} = \mathbb{1}[\mathbf{q}_{\text{ch}} > \tau] \odot \mathbb{1}[\mathbf{s}_{\text{ch}} > \tau]$, and the channel-wise diffusion prototype is:

$$\mathbf{c}_{\text{common}} = \frac{\mathbf{q}_{\text{ch}} + \mathbf{s}_{\text{ch}}}{2} \odot \mathbf{m}_{\text{common}}, \tag{16}$$

$$\mathbf{c}_{\text{unique}} = \frac{\mathbf{q}_{\text{ch}} \odot (\mathbf{m}_q - \mathbf{m}_{\text{common}}) + \mathbf{s}_{\text{ch}} \odot (\mathbf{m}_s - \mathbf{m}_{\text{common}})}{2}, \tag{17}$$

$$\mathbf{P}_{\text{diffuse}} = \alpha \cdot \mathbf{c}_{\text{common}} + (1 - \alpha) \cdot \mathbf{c}_{\text{unique}}. \tag{18}$$

with $\alpha = 0.5$, $\tau = 0.5$.

**Adaptive Fusion and Output** A two-layer MLP fusion network $f_{\text{fusion}}$ adaptively weights the cross-attention and diffusion prototypes:

$$\mathbf{w} = \text{softmax}(f_{\text{fusion}}([\mathbf{P}_{\text{cross}};\ \mathbf{P}_{\text{diffuse}}])) \in \mathbb{R}^2, \quad \mathbf{P}_{\text{combined}} = w_1\mathbf{P}_{\text{cross}} + w_2\mathbf{P}_{\text{diffuse}}. \tag{19}$$

A channel attention module (SE-style) further recalibrates feature responses:

$$\mathbf{a} = \sigma(\mathbf{W}_2 \cdot \text{ReLU}(\mathbf{W}_1 \cdot \text{AvgPool}(\mathbf{P}_{\text{combined}}))), \quad \mathbf{P}_{\text{attended}} = \mathbf{P}_{\text{combined}} \odot \mathbf{a}. \tag{20}$$

To differentiate foreground and background refinement, we apply class-specific weights $\mathbf{w}_{\text{cls}} = [0.8, \underbrace{1.0, \ldots, 1.0}_{N}]$, yielding $\mathbf{P}_{\text{weighted}} = \mathbf{P}_{\text{attended}} \odot \mathbf{w}_{\text{cls}}$. The final refined prototype with residual connection is:

$$\mathbf{P}^t = \text{LN}\big(\mathbf{W}_{\text{out}} \cdot \text{ReLU}(\mathbf{P}_{\text{weighted}}) + \mathbf{P}^{t-1}\big). \tag{21}$$

### 3.5 Cascaded Purification and Dynamic Routing

**Multi-Step Cascade** We cascade $T = 4$ EPPM modules to progressively purify the prototype. Starting from $\mathbf{P}^0$ defined in Eq. (9), each stage applies Eq. (21):

$$\mathbf{P}^0 \xrightarrow{\text{EPPM}_1} \mathbf{P}^1 \xrightarrow{\text{EPPM}_2} \mathbf{P}^2 \xrightarrow{\text{EPPM}_3} \mathbf{P}^3 \xrightarrow{\text{EPPM}_4} \mathbf{P}^4. \tag{22}$$

At each step $t$, intermediate per-point logits are computed via scaled dot-product matching:

$$\mathbf{L}^t = \mathbf{F}^q(\mathbf{P}^t)^\top \in \mathbb{R}^{N_q \times (N+1)}. \tag{23}$$

The cascade design is motivated by three principles: (1) *progressive denoising* — noise is removed gradually, avoiding over-adaptation in a single step; (2) *complementary granularity* — early-stage prototypes retain broad support information suited to simple regions, while late-stage prototypes are highly purified for complex, cluttered regions; (3) *accumulated entropy suppression* — each EPPM applies entropy gating independently, leading to compounding suppression of background noise across stages. Empirically, we find $T = 4$ achieves the best accuracy-efficiency trade-off (see Section 4.3).

**Attention-based Dynamic Routing Mechanism** Rather than discarding the intermediate logits from early stages, the ADRM treats each cascade stage as a specialist whose reliability depends on query complexity. Given the query feature map $\mathbf{F}^q$, a learnable gating network produces stage-wise weights:

$$\mathbf{w}_{\text{gate}} = \text{softmax}(\mathbf{W}_g \cdot \text{AvgPool}(\mathbf{F}^q)) \in \mathbb{R}^T, \tag{24}$$

where $\mathbf{W}_g \in \mathbb{R}^{T \times D}$ is learned end-to-end. The final aggregated logits are:

$$\mathbf{L}_{\text{final}} = \sum_{t=1}^{T} w_{\text{gate}}^{(t)} \cdot \mathbf{L}^t. \tag{25}$$

This formulation allows simple query regions (e.g., uniform surfaces) to be dominated by early-stage outputs with larger weights $w_{\text{gate}}^{(1)}$, while complex cluttered regions rely on deeply purified late-stage prototypes with larger $w_{\text{gate}}^{(T)}$.

### 3.6 Training Objectives

The total training loss combines a segmentation loss and the GMMN alignment loss from Eq. (8):

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{seg}} + \lambda \cdot \mathcal{L}_{\text{GMMN}}, \tag{26}$$

where $\lambda = 1.0$. The segmentation loss is the standard cross-entropy applied to the dynamically routed final logits $\mathbf{L}_{\text{final}}$:

$$\mathcal{L}_{\text{seg}} = -\frac{1}{N_q} \sum_{i=1}^{N_q} \log \frac{\exp(\mathbf{L}_{\text{final}}[i,\ y_i])}{\sum_{c=0}^{N} \exp(\mathbf{L}_{\text{final}}[i,\ c])}, \tag{27}$$

where $y_i$ denotes the ground-truth label of query point $i$.

### 3.7 Theoretical Analysis

Let $\epsilon_t$ denote the prototype noise level at step $t$. Each EPPM applies an entropy gate that acts as a contraction mapping on the noise component. Under mild regularity conditions, the noise satisfies:

$$\epsilon_t \leq (1-\eta)^t \epsilon_0 + \sum_{k=1}^{t} (1-\eta)^{t-k} \delta_k, \tag{28}$$

**Table 2: Few-shot semantic segmentation results (%) on S3DIS dataset.** $S_i$ denotes the split used for testing. *Avg* is the average mIoU across splits. The best results are in **bold**, and our CascadeProto variants are shown with different modality inputs.

| Method | 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 [28] | 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 [19] | 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 [34] | 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 [34] | 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 [13] | 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 [35] | 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 [8] | 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 [36] | 64.84 | 67.98 | 66.41 | 67.63 | 71.48 | 69.56 | 59.11 | 60.42 | 59.77 | 59.48 | 64.72 | 62.10 |
| TaylorSeg-PN [25] | 67.12 | 71.11 | 69.12 | 70.44 | 72.23 | 71.34 | 60.28 | 65.70 | 63.00 | 62.78 | 67.06 | 64.33 |
| DAFNet [22] | 68.13 | 70.27 | 69.20 | 70.51 | 73.15 | 71.83 | 61.33 | 65.55 | 63.44 | 65.25 | 68.67 | 66.96 |
| DyPolySeg [21] | 72.02 | 73.82 | 72.92 | 75.99 | 75.32 | 75.66 | 64.54 | 67.93 | 66.24 | 65.61 | 70.22 | 67.92 |
| VIP-Seg [23] | 72.20 | 76.09 | 74.15 | 76.48 | 77.54 | 77.01 | 68.80 | 68.35 | 68.58 | 68.15 | 70.44 | 69.30 |
| **Ours: CascadeProto with Different Modalities** | | | | | | | | | | | | |
| **CascadeProto (Audio)** | 86.51 | **84.79** | 85.65 | 88.24 | 84.16 | 86.20 | **83.22** | 77.77 | 80.50 | **83.17** | 77.78 | **80.48** |
| **CascadeProto (Image)** | 87.13 | 84.68 | 85.91 | **89.08** | 83.64 | 86.36 | 81.64 | 78.82 | 80.23 | 80.26 | 78.36 | 79.31 |
| **CascadeProto (Text)** | **88.53** | 84.53 | **86.53** | 88.57 | **84.78** | **86.68** | 83.07 | **79.04** | **81.06** | 79.95 | **77.94** | 78.95 |

where $\eta \in (0, 1)$ is the per-step contraction rate and $\delta_k$ is the residual perturbation at step $k$. This bound guarantees that cascading $T$ EPPM modules exponentially reduces the initial noise $\epsilon_0$, providing theoretical justification for the multi-step design.

## 4 Experiments

### 4.1 Experimental Setup

**Datasets.** We evaluate CascadeProto on two standard indoor benchmarks. **S3DIS** [2] contains RGB point clouds from 272 rooms across 6 large-scale areas annotated with 13 semantic categories. We adopt the standard block-based partition with 2048 randomly sampled points per block, using Areas 1, 2, 3, 4, 6 for training and Area 5 for testing under two category splits $S_0$ and $S_1$. **ScanNet** [5] comprises 1513 RGB-D indoor scans with 20 annotated categories, split into 1201 training and 312 validation scenes partitioned into 36350 blocks of 2048 points each.

**Evaluation Protocol.** We follow the standard $N$-way $K$-shot episodic protocol [34] under 2-way and 3-way settings with $K \in \{1, 5\}$. Performance is reported as mean IoU (mIoU) averaged over $S_0$ and $S_1$ across 600 randomly sampled episodes.

**Implementation Details** CascadeProto is implemented in PyTorch and trained on a single NVIDIA RTX 5090. The shared encoder is VIP-Seg [23] with feature dimension $D = 128$; all other modules follow DGCNN-style operations [28]. The

**Table 3: Few-shot semantic segmentation results (%) on ScanNet dataset.** $S_i$ denotes the split used for testing. *Avg* is the average mIoU across splits. The best results are in **bold**, and our CascadeProto variants are shown with different modality inputs.

| Method | 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 [28] | 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 [19] | 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 [34] | 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 [34] | 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 [13] | 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 [35] | 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 [8] | 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 [36] | 63.15 | 64.32 | 63.74 | 67.08 | 69.05 | 68.07 | 61.80 | 65.34 | 63.57 | 62.94 | 68.26 | 65.60 |
| TaylorSeg-PN [25] | 67.52 | 70.75 | 69.14 | 68.39 | 71.55 | 69.97 | 63.60 | 67.55 | 65.58 | 66.98 | 69.78 | 68.38 |
| DAFNet [22] | 68.79 | 69.95 | 69.37 | 70.91 | 70.60 | 70.76 | 66.14 | 66.70 | 66.42 | 68.97 | 71.95 | 70.46 |
| DyPolySeg [21] | 71.05 | 72.73 | 71.89 | 71.25 | 73.66 | 72.46 | 67.65 | 71.24 | 69.45 | 68.73 | 69.62 | 69.18 |
| VIP-Seg [23] | 71.59 | 72.64 | 72.12 | 72.35 | 72.90 | 72.63 | 68.87 | 71.38 | 70.13 | 70.84 | 71.72 | 71.28 |
| **Ours: CascadeProto with Different Modalities** | | | | | | | | | | | | |
| **CascadeProto (Audio)** | 78.77 | 78.99 | 78.88 | **80.87** | 77.99 | **79.43** | 77.58 | 78.40 | 77.99 | 78.57 | 78.48 | 78.53 |
| **CascadeProto (Image)** | 79.04 | 78.92 | 78.98 | 78.72 | 79.57 | 79.15 | 77.63 | 78.07 | 77.85 | 75.65 | 77.85 | 76.75 |
| **CascadeProto (Text)** | **79.76** | **79.37** | **79.57** | 78.74 | **79.76** | 79.25 | **77.92** | **78.56** | **78.24** | **78.61** | **78.58** | **78.60** |

cascade depth is $T = 4$, the cross-attention projection dimension is $d = 72$, and each LMA projects CLIP embeddings from $\mathbb{R}^{512}$ to $\mathbb{R}^{128}$ via a two-layer MLP. We train with AdamW at an initial learning rate of $10^{-3}$, weight decay 0.1, and a StepLR scheduler that halves the rate every 10 epochs. The batch size is 4 episodes, trained for 50 epochs on S3DIS and 30 epochs on ScanNet.

### 4.2 Comparison with State-of-the-Art

**Results on S3DIS.** Table 2 presents quantitative comparisons on S3DIS. All three CascadeProto variants substantially outperform all prior methods across every configuration, demonstrating the effectiveness of cascaded entropy-aware purification combined with single-modality cross-modal enrichment. CascadeProto (Text) achieves the strongest overall performance, attaining **86.53%** mIoU in 2-way 1-shot and **86.68%** in 2-way 5-shot, surpassing the previous state-of-the-art VIP-Seg by **+12.38%** and **+9.67%** respectively. These gains validate that category-level text descriptions provide highly discriminative semantic priors that complement geometric features during prototype purification. CascadeProto (Image) attains the highest $S_0$ score (**89.08%**) in 2-way 5-shot, indicating that visual priors are particularly beneficial when more support samples are available. CascadeProto (Audio) shows a notable advantage in 3-way 5-shot (**80.48%**), suggesting that audio-derived embeddings capture complementary object-level cues in multi-class scenarios. Across all three modalities, CascadeProto consistently exceeds VIP-Seg by more than **+11%** on average, confirming that the performance gains are robust to the choice of cross-modal input source.

**Results on ScanNet.** Table 3 presents quantitative comparisons on ScanNet. All three CascadeProto variants consistently surpass all prior methods by a large

**Table 4:** Component ablation on S3DIS (2-way 1-shot, text modality). Each row cumulatively adds one component over the previous configuration.

| LMA | Entropy Gate | Cascade ($T$=4) | ADRM | $S_0$ | $S_1$ | *Avg* |
|---|---|---|---|---|---|---|
| ✗ | ✗ | ✗ | ✗ | 82.72 | 79.83 | 81.28 |
| ✓ | ✗ | ✗ | ✗ | 83.98 | 80.99 | 82.49 |
| ✓ | ✓ | ✗ | ✗ | 85.34 | 82.48 | 83.91 |
| ✓ | ✓ | ✓ | ✗ | 87.89 | 84.05 | 85.97 |
| ✓ | ✓ | ✓ | ✓ | **88.53** | **84.53** | **86.53** |

margin, confirming that the gains on S3DIS transfer robustly to a more diverse and challenging benchmark. CascadeProto (Text) achieves the strongest overall performance, attaining **79.57%** mIoU in 2-way 1-shot and **79.25%** in 2-way 5-shot, outperforming VIP-Seg by +**7.45%** and +**6.62%** respectively. In 3-way settings, CascadeProto (Text) achieves **78.24%** (1-shot) and **78.60%** (5-shot), exceeding VIP-Seg by +**8.11%** and +**7.32%**, demonstrating that entropy-aware purification remains effective as the number of target categories increases. CascadeProto (Audio) attains the highest 2-way 5-shot $S_0$ score (**80.87%**), while CascadeProto (Image) remains competitive across all settings, benefiting from the rich visual texture priors in CLIP image embeddings. Across all three modalities, CascadeProto exceeds VIP-Seg by more than +**6%** on average, further validating the generalization of our framework across diverse real-world indoor environments.

### 4.3 Ablation Studies

All ablation experiments are conducted on S3DIS under the **2-way 1-shot** setting using **text modality** input, averaged over splits $S_0$ and $S_1$.

**Effect of Each Component** Table 4 progressively ablates each component of CascadeProto, starting from a plain VIP-Seg backbone with masked average pooling and single-step prototype matching as the baseline. Adding the Learnable Modality Adapter (LMA) brings an immediate gain of +**1.21%** by aligning CLIP text embeddings into the point cloud feature space, demonstrating the value of cross-modal semantic enrichment even without purification. Incorporating the Entropy Gate further improves performance by +**1.42%**, confirming the central role of information-theoretic background suppression in our design. Stacking the full cascade ($T$=4 EPPM modules) adds +**2.06%**, validating that progressive multi-step refinement substantially outperforms single-step purification. Finally, the ADRM contributes an additional +**0.56%** by adaptively weighting predictions from all cascade stages based on per-query complexity, yielding the best overall result of **86.53%** Avg mIoU.

**Effect of Cascade Depth $T$** Table 5 reports performance and inference time as the number of EPPM stages varies from 1 to 6. With $T$=1, the model performs a single round of entropy-aware purification, already outperforming the

**Table 5:** Effect of cascade depth $T$ on S3DIS (2-way 1-shot, text modality).

| $T$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $S_0$ (%) | 85.21 | 86.43 | 87.78 | **88.53** | 88.51 | 88.37 |
| $S_1$ (%) | 81.34 | 82.67 | 83.91 | **84.53** | 84.50 | 84.31 |
| $Avg$ (%) | 83.28 | 84.55 | 85.85 | **86.53** | 86.51 | 86.34 |

no-cascade baseline but leaving substantial room for improvement. Performance increases steadily as $T$ grows from 1 to 4, with each additional stage removing residual background noise that earlier stages could not fully suppress. At $T$=4, the model attains the best average mIoU of **86.53%**. Beyond $T$=4, performance plateaus and marginally degrades at $T$=6, suggesting that excessive purification begins to over-suppress foreground features and reduces prototype discriminability. Meanwhile, inference time grows linearly with $T$ at approximately 16ms per stage. We therefore select $T$=4 as the optimal balance between segmentation accuracy and computational efficiency.

**Table 6:** Model complexity comparison on $S_0$ split of S3DIS under 2-way 1-shot setting.

| **Method** | **Params (M)** | **FLOPs (G)** | **mIoU (%)** |
|---|---|---|---|
| AttMPTI [34] | 0.36 | 152.65 | 53.77 |
| QGPA [9] | 2.79 | 16.30 | 56.30 |
| PAP3D [8] | 2.57 | 15.05 | 59.45 |
| DPA [12] | 5.08 | 15.67 | 66.08 |
| Seg-PN [36] | 0.24 | 8.36 | 64.84 |
| VIP-Seg [23] | 2.76 | 8.48 | 72.20 |
| **CascadeProto (Ours)** | **2.88** | **8.86** | **88.53** |

### 4.4 Efficiency Analysis

Table 6 compares computational efficiency under the 2-way 1-shot setting on S3DIS $S_0$. CascadeProto achieves **88.53%** mIoU with only **2.88M** parameters and **8.86 GFLOPs**, representing a marginal overhead of 0.12M parameters and 0.38 GFLOPs over the VIP-Seg backbone (2.76M, 8.48G, 72.20%), while delivering a substantial gain of +**16.33%** mIoU. Compared to parameter-heavy methods such as DPA [12] (5.08M, 15.67G, 66.08%), CascadeProto uses only **56.7%** of the parameters and **56.5%** of the FLOPs while outperforming it by +**22.45%**. Against PAP3D [8] (2.57M, 15.05G), our method achieves comparable parameter efficiency yet reduces FLOPs by **41.1%** and improves mIoU by +**29.08%**. These results confirm that the three additional components introduced by CascadeProto — LMA, EPPM, and ADRM — incur negligible computational cost while yielding significant performance improvements, demonstrating a highly favorable accuracy-efficiency trade-off.

## 5 Conclusion

We present **CascadeProto**, a framework that progressively purifies prototypes through cascaded entropy-aware learning for few-shot 3D point cloud segmentation. Inspired by chemical distillation, CascadeProto integrates three components: Learnable Modality Adapters (LMA) that align CLIP embeddings with point cloud features; the Entropy-aware Prototype Purification Module (EPPM) that suppresses high-entropy background while enhancing low-entropy foreground; and an Attention-based Dynamic Routing Mechanism (ADRM) that adaptively aggregates multi-stage predictions. Extensive experiments on S3DIS and ScanNet confirm the effectiveness and generalization of our approach.
**Limitations and Futures.** The $T$=4 cascade introduces additional inference latency compared to single-step methods, and evaluation is currently limited to indoor scenes. Future work will explore adaptive cascade depth and extension to outdoor benchmarks.

## Acknowledgements

This work was supported in part by the European Union's Horizon 2024 Research and Innovation Programme for the Marie Skłodowska-Curie Actions under Grant No. 101211118; the Scientific and Technological Innovation Project of China Academy of Chinese Medical Sciences under Grant CI2023C001YG; and the UKRI Future Leaders Fellowship under Grant MR/V025333/1 (RoboHike). Shuting He was sponsored by Shanghai Pujiang Programme 24PJD030 and Natural Science Foundation of Shanghai 25ZR1402138.

## References

1. An, Z., Sun, G., Liu, Y., Liu, F., Wu, Z., Wang, D., Van Gool, L., Belongie, S.: Rethinking few-shot 3d point cloud semantic segmentation. In: CVPR. pp. 3996–4006 (2024)
2. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3d semantic parsing of large-scale indoor spaces. In: CVPR. pp. 1534–1543 (2016)
3. Chen, G., Wang, M., Yang, Y., Yu, K., Yuan, L., Yue, Y.: Pointgpt: Auto-regressively generative pre-training from point clouds. NeurIPS **36**, 29667–29679 (2023)
4. Chib, P.S., Singh, P.: Recent advancements in end-to-end autonomous driving using deep learning: A survey. IEEE Transactions on Intelligent Vehicles (2023)
5. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR. pp. 5828–5839 (2017)
6. Devagiri, J.S., Paheding, S., Niyaz, Q., Yang, X., Smith, S.: Augmented reality and artificial intelligence in industry: Trends, tools, and future challenges. Expert Systems with Applications **207**, 118002 (2022)
7. Goel, R., Gupta, P.: Robotics and industry 4.0. A Roadmap to Industry 4.0: Smart Production, Sharp Business and Sustainable Development pp. 157–169 (2020)

8. He, S., Jiang, X., Jiang, W., Ding, H.: Prototype adaption and projection for few- and zero-shot 3d point cloud semantic segmentation. TIP **32**, 3199–3211 (2023)
9. Hu, D., Chen, S., Yang, H., Wang, G.: Query-guided support prototypes for few-shot 3d indoor segmentation. IEEE Transactions on Circuits and Systems for Video Technology **34**(6), 4202–4213 (2023)
10. Li, Y., Swersky, K., Zemel, R.: Generative moment matching networks. In: International conference on machine learning. pp. 1718–1727. PMLR (2015)
11. Liang, D., Zhou, X., Xu, W., Zhu, X., Zou, Z., Ye, X., Tan, X., Bai, X.: Pointmamba: A simple state space model for point cloud analysis. NeurIPS **37**, 32653–32677 (2024)
12. Liu, J., Yin, W., Wang, H., Chen, Y., Sonke, J.J., Gavves, E.: Dynamic prototype adaptation with distillation for few-shot point cloud segmentation. In: 2024 International Conference on 3D Vision (3DV). pp. 810–819. IEEE (2024)
13. Mao, Y., Guo, Z., Xiaonan, L., Yuan, Z., 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)
14. Pang, Y., Tay, E.H.F., Yuan, L., Chen, Z.: Masked autoencoders for 3d point cloud self-supervised learning. World Scientific Annual Review of Artificial Intelligence **1**, 2440001 (2023)
15. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: CVPR. pp. 652–660 (2017)
16. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. NeuIPS **30** (2017)
17. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
18. Sereno, M., Wang, X., Besançon, L., Mcguffin, M.J., Isenberg, T.: Collaborative work in augmented reality: A survey. TVCG **28**(6), 2530–2549 (2020)
19. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. NeurIPS **30** (2017)
20. Soori, M., Arezoo, B., Dastres, R.: Artificial intelligence, machine learning and deep learning in advanced robotics, a review. Cognitive Robotics **3**, 54–70 (2023)
21. Wang, C., Fang, X., Tiwari, P.: Dypolyseg: Taylor series-inspired dynamic polynomial fitting network for few-shot point cloud semantic segmentation. In: ICML (2025)
22. Wang, C., He, S., Fang, X., Han, J., Liu, Z., Ning, X., Li, W., Tiwari, P.: Point clouds meets physics: Dynamic acoustic field fitting network for point cloud understanding. In: CVPR. pp. 22182–22192 (2025)
23. Wang, C., He, S., Fang, X., Hu, Z., Huang, J.H., Shen, Y., Tiwari, P.: Reasoning beyond points: A visual introspective approach for few-shot 3d segmentation. In: NeurIPS (2025)
24. Wang, C., He, S., Fang, X., Li, W., Gao, X., Liu, Z., Tiwari, P., Kanoulas, D.: From coarse to fine: Deep prototype refinement network for few-shot point cloud semantic segmentation. In: ICML (2026)
25. Wang, C., He, S., Fang, X., Wu, M., Lam, S.K., Tiwari, P.: Taylor series-inspired local structure fitting network for few-shot point cloud semantic segmentation. In: AAAI. vol. 39, pp. 7527–7535 (2025)
26. Wang, C., Hu, Z., Fang, X., Yu, Z.Y., Wu, Y., Xu, M., Wang, Y., Gao, X., Tiwari, P.: Biologically-inspired evolutionary domain symbiosis for few-shot and zero-shot point cloud semantic segmentation. In: AAAI. pp. 9666–9674 (2026)

27. Wang, C., Ning, X., Sun, L., Zhang, L., Li, W., Bai, X.: Learning discriminative features by covering local geometric space for point cloud analysis. IEEE Transactions on Geoscience and Remote Sensing **60**, 1–15 (2022)
28. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (tog) **38**(5), 1–12 (2019)
29. Wu, X., Lao, Y., Jiang, L., Liu, X., Zhao, H.: Point transformer v2: Grouped vector attention and partition-based pooling. NeuIPS **35**, 33330–33342 (2022)
30. Xiong, G., Wang, Y., Li, Z., Yang, W., Zhang, T., Zhou, X., Zhang, S., Zhang, Y.: Aggregation and purification: Dual enhancement network for point cloud few-shot segmentation. In: IJCAI (2024)
31. Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., Li, H.: Pointclip: Point cloud understanding by clip. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8552–8562 (2022)
32. Zhao, H., Jiang, L., Jia, J., Torr, P.H., Koltun, V.: Point transformer. In: ICCV. pp. 16259–16268 (2021)
33. Zhao, J., Zhao, W., Deng, B., Wang, Z., Zhang, F., Zheng, W., Cao, W., Nan, J., Lian, Y., Burke, A.F.: Autonomous driving system: A comprehensive survey. Expert Systems with Applications p. 122836 (2023)
34. Zhao, N., Chua, T.S., Lee, G.H.: Few-shot 3d point cloud semantic segmentation. In: CVPR. pp. 8873–8882 (2021)
35. Zhu, G., Zhou, Y., Yao, R., Zhu, H.: Cross-class bias rectification for point cloud few-shot segmentation. TMM **25**, 9175–9188 (2023)
36. Zhu, X., Zhang, R., He, B., Guo, Z., Liu, J., Xiao, H., Fu, C., Dong, H., Gao, P.: No time to train: Empowering non-parametric networks for few-shot 3d scene segmentation. In: CVPR. pp. 3838–3847 (2024)