buildings

MDPI

Article

Multi-Stage Pseudo-Label Guided Weakly Supervised Point Cloud Semantic Segmentation for Building Scenes

Yanfei Su [1], Yunlong Wang [1], Tao Ma [1] and Jing Peng [2,*]

[1] Fujian Key Laboratory of Pattern Recognition and Image Understanding, School of Computer and Information Engineering, Xiamen University of Technology, Xiamen 361024, China; suyanfei@xmut.edu.cn (Y.S.)
[2] School of Economics and Management, Xiamen University of Technology, Xiamen 361024, China
* Correspondence: pjsophie@xmut.edu.cn

Abstract

Point cloud semantic segmentation is a fundamental task for 3D scene understanding. However, acquiring dense point-wise annotations is labor-intensive and expensive. To alleviate the annotation burden, weakly supervised methods using extremely sparse labels (e.g., 0.1%) have attracted increasing attention. While existing methods have demonstrated promising performance, they struggle to generate sufficiently reliable pseudo-labels from unlabeled points, which limits further improvement in segmentation performance under extremely sparse supervision. To address this challenge, we propose a **Multi-stage Pseudo-label Guided (MPG)** training framework, which mines high-quality pseudo-labels through a two-stage training strategy. In the first stage, we introduce a **Prototype-guided Pseudo-label Filter (PPF)** to capture latent semantic information by comparing unlabeled features with class prototypes and removes unreliable pseudo-labels using a predefined threshold. In the second stage, we employ a teacher-student architecture and introduce a **Pseudo-Label Fusion (PLF)** strategy. PLF applies consistency and confidence filtering on two data-augmentation branches of the frozen teacher model to produce refined pseudo-labels, which are then combined with PPF outputs to train the student network. This design allows the student model to inherit distilled knowledge while exploring additional semantic information. Extensive experiments on indoor and outdoor building scenes demonstrate the effectiveness of MPG. Under extremely sparse supervision, MPG achieves 69.5% mIoU on S3DIS and 59.5% on XIAMEN-Buildings, achieving state-of-the-art performance among weakly supervised methods and performance comparable to some fully supervised methods.

Keywords: weak supervision; semantic segmentation; large-scale point clouds

Check for updates

Academic Editor: Majid Movahedi Rad

Received: 18 September 2026
Revised: 23 October 2026
Accepted: 25 October 2026
Published: 27 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Semantic segmentation of large-scale 3D point clouds is a fundamental task in 3D scene understanding [1–4]. By assigning precise semantic labels to points, it enables intelligent systems to perceive the real world effectively. This task plays a critical role in various applications, including autonomous driving [5], human-computer interaction [6], and remote sensing [7]. Although fully-supervised learning has achieved significant progress [8–15], obtaining dense point-wise annotations remains labor-intensive, as a single point cloud scene often contains tens of thousands of points [16,17]. Therefore, developing semantic segmentation methods that rely on sparse labels is crucial for reducing annotation costs and facilitating practical applications [5,18–20].

Recent research has focused on weakly-annotated data to reduce reliance on dense supervision while maintaining competitive performance [5,21,22]. Existing methods can be divided into five categories:

(1) 2D label-based methods [2,23]. Inspired by advances in 2D image understanding, early studies projected point clouds onto 2D planes and leveraged image-level annotations for supervision. However, such projection-based strategies inevitably suffer from information loss and high sensitivity to viewpoint variations. To overcome these limitations, researchers have increasingly shifted their attention toward native 3D learning.

(2) Self-training methods. Unlike projection-based approaches, self-training methods iteratively generate pseudo-labels from model predictions and use them as supervision for subsequent training stages. Representative methods, including CAMs [24], MPRM [16] and J2D3D [25], progressively generate point-wise pseudo-labels from sub-cloud or image-level annotations. However, these methods mainly rely on prediction confidence and fail to fully exploit the rich semantic information embedded in unlabeled points.

(3) Perturbation consistency methods. These approaches are built upon the assumption that network predictions should remain invariant under input perturbations. Various perturbations, such as affine transformations, point jittering, downsampling, or masking, are applied to construct paired point clouds for consistency learning [26,27]. Specifically, CPCM [28] extends this work by adopting region-wise masking as a strong perturbation and optimizing a masked consistency loss to capture contextual information. However, since it does not explicitly generate pseudo-labels for unlabeled points, its ability to provide additional supervisory signals remains limited.

(4) Prototype-Guided Methods. These methods perform feature learning on point clouds by class prototypes. e.g., HybridCR [29], C3 [30] and MulPro [31] add a regularization that pulls the feature of a point toward its own class prototype and pushes it away from the prototypes of the other classes. These methods use prototypes only at the feature level.

(5) Teacher–student pseudo-label learning Methods. These methods employ a teacher model to generate pseudo-labels for unlabeled points and a student model to learn from them, and they often adopt an exponential-moving-average teacher or a frozen first-stage model to stabilize training [32–34]. It is therefore prone to confirmation bias and error accumulation under extremely sparse supervision.

The above studies are primarily conducted under the setting of partial point-wise labeling, which achieves an effective trade-off between annotation cost and segmentation accuracy [20]. Under this setting, only a tiny fraction of points (e.g., 0.1%) are annotated in each scene. However, under extremely sparse supervision, the quality of pseudo-labels remains a major bottleneck, which can be attributed primarily to two factors. First, the generation of pseudo-labels largely relies on model predictions. It does not fully exploit the semantic information embedded in unlabeled points, leading to the underutilization of a large amount of potentially informative data. Second, supervision derived from single-source pseudo-labels is prone to confirmation bias and error accumulation, leading to noisy pseudo-labels that are spatially misaligned. Directly using such unreliable pseudo-labels may further amplify errors during training, thereby hindering performance improvement.

To address these challenges, we propose a **Multi-stage Pseudo-label Guided** (**MPG**) training framework that exploits a multi-source pseudo-label generation strategy. MPG captures latent semantic information from unlabeled point clouds via a two-stage training process with two dedicated components, while further mining reliable pseudo-labels. In the first stage, we introduce a **Prototype-guided Pseudo-label Filter** (**PPF**) to extract latent semantic information from unlabeled point clouds. Specifically, PPF generates coarse-

grained pseudo-labels by calculating the similarity between unlabeled point clouds and class prototypes and refines them by filtering out unreliable pseudo-labels using a predefined threshold. As a result, the network is supplied with more reliable supervisory signals for the first stage training. To generate more reliable pseudo-labels, we employ an additional teacher-student network to improve the segmentation performance. In the second stage, we first freeze the model trained in the first stage as a teacher. Then, we introduce a **Pseudo-Label Fusion** (**PLF**) strategy as another avenue for exploring pseudo-labels. Specifically, PLF applies consistency filtering to the feature distributions output by the two data augmentation branches of the teacher model, retaining only those pseudo-labels with consistent prediction results; It then uses a confidence filter to further remove potentially unreliable pseudo-labels. Finally, the student model is trained using the reliable pseudo-labels derived separately from the teacher model and PPF. This design allows the student model to inherit distilled knowledge from the teacher while leveraging the PPF to further explore the additional semantic information. Comprehensive evaluations verify that our MPG attains state-of-the-art results across the large-scale indoor benchmarks, S3DIS [35] and ScanNet-V2 [36], while consistently delivering excellent performance on the outdoor XIAMEN-Buildings dataset [37]. These outcomes confirm the efficacy of our proposed method.

Our contributions are summarized as follows:

- We propose the Multi-stage Pseudo-label Guided training framework (MPG), a weakly supervised point cloud semantic segmentation network that leverages high-quality pseudo-labels obtained through a multi-source strategy.
- We propose the Prototype-guided Pseudo-label Filter (PPF) that mines latent semantic information from large-scale unlabeled point clouds and provides effective supervision signals for weakly supervised training.
- We introduced the Pseudo-Label Fusion (PLF) strategy, providing another avenue for discovering reliable pseudo-labels.
- Extensive experiments on prevalent benchmark datasets verify that our proposed method achieves state-of-the-art (SOTA) performance and outperforms a variety of fully-supervised competitors.

2. Related Work

This section reviews the existing literature on point cloud segmentation, which primarily branches into two mainstream paradigms: fully-supervised and weakly-supervised methods. We first elaborate on the fully-supervised methods in Section 2.1, followed by a comprehensive review of weakly-supervised counterparts in Section 2.2.

2.1. Fully-Supervised Point Cloud Segmentation

Existing fully-supervised methods for point clouds semantic segmentation mainly fall into three categories: point-based methods [38,39], voxel-based methods [9,40,41], and hybrid methods [8,14]. Early studies directly applied 2D convolutions to projected point cloud images. Although this strategy is computationally efficient, the projection step inevitably loses important 3D geometric information. To address this issue, point-based methods were developed to process irregular and unordered points directly. Representative architectures such as PointNet [38] and PointNet++ [39] are naturally suited to point clouds. However, they are generally less effective than 2D convolution in capturing contextual information. Voxel-based methods organize neighboring points into regular grids and often adopt sparse convolution to efficiently process sparse voxel representations [9]. More recent studies combine the advantages of point-based and voxel-based designs, leading to hybrid methods [8,14]. These methods usually achieve stronger performance, but they also require more complex architectures and higher training costs. Despite their progress,

fully-supervised point cloud segmentation methods still rely heavily on densely annotated labels, which limits their practical use in many real-world scenarios.

2.2. Weakly-Supervised Point Cloud Semantic Segmentation

Weakly-supervised point cloud semantic segmentation has emerged as a promising approach to alleviate the high annotation cost of fully-supervised segmentation methods. According to their core supervision strategies and technical principles, existing weakly supervised segmentation methods can be roughly categorized into three main categories: 2D label-based methods, self-training methods, perturbation consistency-based methods, Prototype-Guided methods and teacher–student pseudo-label learning methods.

2.2.1. 2D Label-Based Methods

These methods perform 3D segmentation using only 2D supervision [2]. Drawing on 3D reconstruction and graph convolution techniques, GPFN [23] introduced a graph-based pyramid feature network and a 2D–3D joint optimization module. Their approach uses perspective rendering and semantic fusion to align 2D and 3D features. Similarly, LDLS [42] employs Mask-RCNN [43] for RGB image segmentation, propagating these labels to 3D space via a projection-based graph to generate full point clouds annotations. OBSNet [44] later refined this framework [23] to improve segmentation accuracy. To eliminate the need for 2D image annotations, REAL [45] introduced the PP2S module to link 2D and 3D modalities, incorporating an active learning strategy to update pseudo-labels during training. However, basic projection schemes tend to lose part of point cloud information and suffer from viewpoint restrictions. To overcome the drawbacks of existing 2D methods, researchers have widely explored 3D solutions.

2.2.2. Self-Training Methods

Self-Training typically generates dynamic pseudo-labels from previous predictions to guide subsequent training. For instance, MPRM [16] used a Point CAM alongside a multi-path region mining module to produce pseudo-labels for fully-supervised training. Similarly, J2D3D [25] incorporated 2D and 3D information. Unlike the above CAM-based schemes, OTOC [21] iteratively generates and refines pseudo labels via super-voxel graph propagation and a learned relation network. SSPC-Net [46] generates pseudo labels via dynamic label propagation on superpoint graphs and filters low-quality ones with superpoint dropout. To address the issue of noise pseudo-labels generated by single-scale filtering, RPSC [47] generates fused multi-scale predictions and selects reliable pseudo labels by checking prediction consistency and confidence. Despite these advancements, such methods cannot effectively utilize intrinsic information from raw point clouds.

2.2.3. Perturbation Consistency-Based Methods

Perturbation consistency methods assume that network features should remain invariant under diverse transformations [26,27]. Common approaches include applying affine transforms with point jitter [48], downsampling [22], or masking [28] to generate paired point clouds. For instance, PSD [49] proposed perturbed self-distillation to enforce prediction consistency across different branches. This method uses Jensen-Shannon (JS) divergence as a consistency loss to model graph topology and propagate labels among points. To better handle unreliable pseudo labels, RAC-Net [50] incorporates prediction confidence and model uncertainty to distinguish reliable pseudo-labels from ambiguous ones, applying adaptive constraints to mitigate under-utilization. CPCM [28] extends perturbation consistency by using region-wise masking as a strong perturbation, optimizing a masked consistency loss to capture contextual information. Despite these efforts, such

methods are often limited by sparse supervision signals, which remains a bottleneck for further performance gains.

2.2.4. Prototype-Guided Methods

Prototype-guided methods perform feature learning on point clouds by adding regularization. HybridCR [29] proposes a global guidance contrastive regularization and uses a class prototype to pull each point feature to its own prototype and push it away from the prototypes of the other classes. C3 [30] proposes a cross-scene contrastive regularization and uses class prototypes to pull the prototypes of the same class from two point clouds closer and keep those of different classes apart. MulPro [31] proposes a multi-prototype classifier and uses multiple prototypes per class as classifier weights, which are updated by a subclass-averaging constraint and a prototype-diversity constraint. These methods use the class prototype only to improve the point features at the feature level. In contrast, MPG uses the class prototype as a semantic anchor to generate pseudo-labels.

2.2.5. Teacher–Student Pseudo-Label Learning Methods

Teacher-student pseudo-label learning methods use a teacher model to generate pseudo-labels for unlabeled points. DBCL [32] builds a dual-branch consistency learning framework and uses an exponential-moving-average teacher to provide reliable pseudo-labels for unlabeled points. CASC [33] adds class-agnostic and class-specific consistency modules with an exponential-moving-average teacher, but relies on the confidence magnitude alone to judge the unlabeled points. EviUD [34] improves this by decomposing evidential uncertainty with an exponential-moving-average teacher to split unlabeled points into a reliable set and a fuzzy set that receive different supervision. These methods obtain pseudo-labels from a single source, which easily leads to information bias and error accumulation. In contrast, MPG obtains pseudo-labels from multiple sources.

3. Methods

3.1. Overview

In this paper, we adopt CPCM as the backbone network for weakly supervised point cloud semantic segmentation. Although CPCM achieves promising performance by exploiting contextual information and consistency learning from sparsely annotated point clouds, it still suffers from limited utilization of the rich semantic information contained in unlabeled points. Specifically, the latent supervisory signals within unlabeled data are not sufficiently explored, which restricts the further improvement of segmentation performance. To address this limitation, we propose a novel Multi-stage Pseudo-label Guided (MPG) framework, which aims to progressively mine and exploit reliable pseudo-labels from unlabeled points. As illustrated in Figure 1, MPG is built upon the CPCM backbone and introduces two novel components: the Prototype-guided Pseudo-label Filter (PPF) and the Pseudo-Label Fusion (PLF) strategy. The proposed framework adopts a two-stage training paradigm, where reliable pseudo-labels are progressively generated, filtered, and integrated to enhance semantic representation learning.

During the first stage, we train an initial segmentation model based on the CPCM backbone and introduce the PPF module to exploit the semantic information from unlabeled points. Specifically, PPF generates pseudo-labels by measuring the similarity between point features and class prototypes, and progressively improves the quality of pseudo-labels by removing unreliable predictions with a predefined confidence threshold. This stage aims to establish an initial model with reliable pseudo-label generation capability.

In the second stage, we further introduce a teacher–student learning strategy to enhance pseudo-label utilization. The well-trained model from the first stage is employed as the

teacher network, while a student network initialized from scratch. The student model inherits the CPCM backbone and incorporates both the PPF module and the proposed PLF strategy. By combining prototype-based filtering and complementary pseudo-label information, the student network is able to exploit more reliable supervisory signals from unlabeled points, leading to improved semantic representation learning.

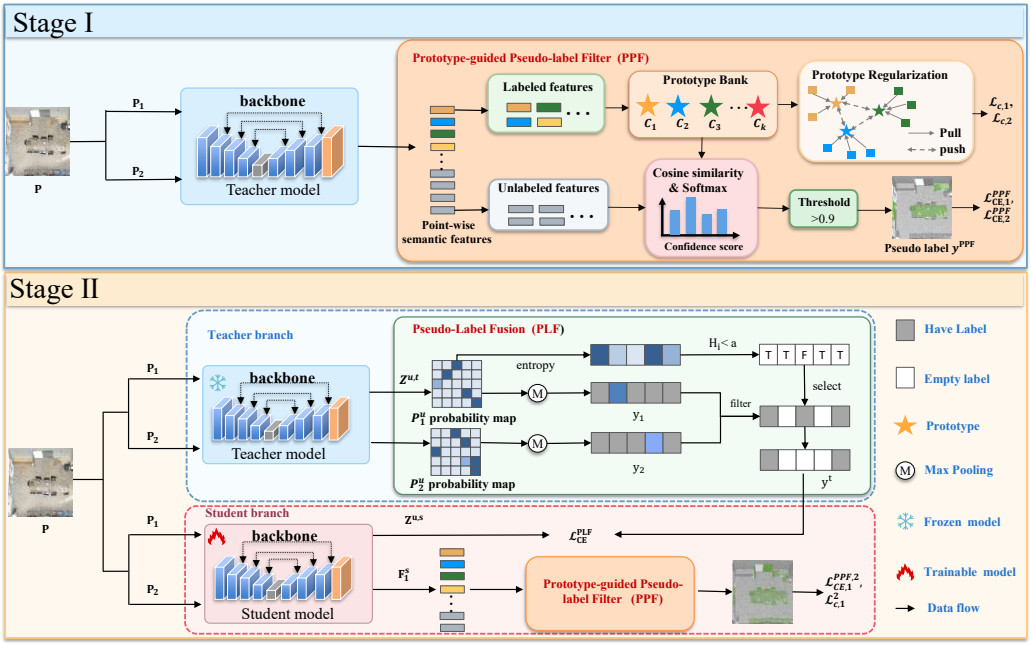


Figure 1. The architecture of MPG. The framework consists of two key components: the Prototype-guided Pseudo-label Filter (PPF) and the Pseudo-label Fusion (PLF) strategy. Training is conducted in two stages with different component configurations. In the first stage, PPF is employed to generate and filter pseudo-labels for network training. In the second stage, the frozen teacher network uses the PLF strategy to generate the pseudo-label for training the student network. Meanwhile, the student applies PPF and integrates the teacher-generated pseudo-labels to obtain more reliable supervision for optimization.

3.2. Preliminaries

Notations. Formally, a point cloud is a collection of N points $\mathbf{P} = \{p_1, p_2, \dots, p_N\}$,where each point p_n often comprises the geometric coordinates and RGB information, i.e., $p_n = (x_n, y_n, z_n, r_n, g_n, b_n)$. To accomplish the point clouds semantic segmentation task, given a point cloud P and a backbone segmentation network $f_\theta(\cdot)$ parameterized by θ. We expect the model to produce point-wise classification logits $\mathbf{Z} = \mathrm{Softmax}(f_\theta(\mathbf{P}))$ where $\mathbf{Z}[n] \in (0, 1)$, $\mathrm{argmax}(\mathbf{Z}[n]) \in C$ and $c = 0, 1, 2, \dots, C-1$ is a predefined category set with C classes. Unlike the fully-supervised point clouds semantic segmentation that provides the label of every point in P, only sparse annotations are available in weakly-supervised point clouds semantic segmentation. The weakly-labeled point clouds data comprises two parts, the labeled part and the unlabeled part, i.e., $(\mathbf{P}, \mathbf{Y}) = \{(p^l, y^l) \mid l \in \mathbf{L}\} \cup \{(p^u, \emptyset) \mid u \in \mathcal{U}\}$, where $\mathbf{L}, \mathbf{U}$ denote the index sets of the labeled and unlabeled points respectively and $\emptyset$ is a special token denoting the label is unavailable. During model training, a dataset $D = \{(\mathbf{P}, \mathbf{Y})\}$ includes hundreds or thousands of point clouds and weak-label pair is provided. Two random augmentations are applied $\mathbf{P}_1 = \mathrm{Aug}_1(\mathbf{P})$ and $\mathbf{P}_2 = \mathrm{Aug}_2(\mathbf{P})$.

CPCM is employed as the backbone network to extract discriminative point-wise features. CPCM improves point cloud representation learning by introducing contextual masked modeling and consistency regularization, which enables the network to capture both local geometric structures and global contextual information. Based on the representation

ability of CPCM, we further develop our proposed MPG module to enhance semantic understanding of point clouds.

3.2.1. First Stage

To effectively exploit the limited label points and latent semantic cues embedded in unlabeled point clouds, we incorporate the Prototype-guided Pseudo-label Filter (PPF) into the backbone during the first training stage. Given the features $\mathbf{F}_1 = f_\theta(P_1)$ and $\mathbf{F}_2 = f_\theta(P_2)$ extracted from the two branches, PPF processes labeled and unlabeled points through distinct mechanisms to fully leverage their complementary supervisory information. For simplicity, unless otherwise specified, the following descriptions are based on the first augmentation branch.

For labeled points. Sparsely annotated labeled points P^l provide essential categorical prior. We compute class prototypes $\mathbf{c}_k$ as the centroid of all features within class k, which minimizes the average intra-class feature distance. These prototypes serve as reliable anchors in contrastive learning, enabling the network to effectively leverage supervision from labeled points. The class prototype $\mathbf{c}_k$ is computed on a per-batch basis:

$$\mathbf{c}_k = \frac{1}{|P_k^l|} \sum_{P^l \in P_k^l} f_\theta(P^l), \tag{1}$$

where P_k^l denotes the subset of labeled points P^l belonging to category k, $|P_k^l|$ represents the number of points in P_k^l.

To learn more discriminative and robust feature representations, we introduce a prototype-based contrastive learning objective inspired by prior self-supervised representation learning methods. Specifically, the proposed objective encourages feature embeddings to align with their corresponding class prototypes while remaining distinct from prototypes of other classes. Consequently, intra-class compactness and inter-class separability are jointly enhanced. The contrastive loss is formulated as:

$$\mathcal{L}_{c,1} = -\frac{2}{K} \sum_{k=1}^{K} \frac{1}{N_k(N_k-1)} \sum_{1 \le i < j \le N_k} \bar{\mathbf{f}}_{k,i}^\top \bar{\mathbf{f}}_{k,j} + \frac{2}{K(K-1)} \sum_{1 \le i < j \le K} \bar{\mathbf{c}}_i^\top \bar{\mathbf{c}}_j, \tag{2}$$

where K denotes the total number of classes, corresponding to the number of prototypes. $\bar{\mathbf{c}}_i$ represents the ℓ_2-normalized unit vector of the i-th class prototype. $\bar{\mathbf{f}}_{k,i}$ denotes the ℓ_2-normalized unit feature vector of the i-th sample belonging to the k-th class. N_k indicates the number of samples in the c-th class. The contrastive loss $\mathcal{L}_{c,2}$ corresponding to branch P_2 shares an identical calculation formulation with $\mathcal{L}_{c,1}$.

For unlabeled points. For the feature embeddings $\mathbf{f}^u = f_\theta(P^u)$ of unlabeled points P^u, we leverage the class prototypes computed from labeled points to exploit the semantic information contained in unlabeled data. Specifically, we measure the similarity between each unlabeled feature and all class prototypes, treating the prototypes as semantic anchors to propagate sparse supervision throughout the point cloud. The resulting similarity scores are converted into class probabilities, where higher similarity indicates a greater likelihood of belonging to the corresponding class. The prototype-based probability distribution is defined as:

$$S^u = \mathrm{softmax}\left(\bar{\mathbf{f}^u}^\top \cdot \bar{\mathbf{c}}_k\right), \tag{3}$$

where $\bar{\mathbf{f}^u}$ represents the ℓ_2-normalized unit vector of the f^u, $\bar{\mathbf{c}}_k$ denotes the ℓ_2-normalized unit feature vector of the $\mathbf{c}_k$, the operator $\cdot$ denotes the dot product between two feature vectors.

Considering that unlabeled point clouds may contain noisy points, we introduce a filtering mechanism to discard unreliable pseudo-labels. Given a threshold τ, the final predicted label is presented as y^{PPF}. The filtering process is represented as follows:

$$y^u = \arg\max(S^u), \quad \text{s.t.} \quad \max(S^u) \geq \tau, \tag{4}$$

where y^u is set to $\emptyset$ if the confidence score $\max(S^u)$ is below the threshold τ.

With the high-quality pseudo labels $\boldsymbol{y^u}$ screened above as supervision signals, we formulate the cross-entropy loss to optimize the network parameters:

$$\mathcal{L}_{\text{CE},1}^{\text{PPF}} = -\frac{1}{N_{\text{valid}}} \sum_{y^u \neq \emptyset} y^u \log(\hat{y^u}), \tag{5}$$

where $\hat{y^u}$ denotes the predicted probability of y^u, and N_{valid} denotes the number of samples with valid pseudo-labels. The cross-entropy loss $\mathcal{L}_{\text{CE},2}^{\text{PPF}}$ for P_2 is calculated in the same way as that for P_1.

As mentioned above, PPF leverages class prototypes to perform contrastive learning on labeled points and generate reliable unlabeled pseudo-labels for unlabeled points. The overall process of PPF is formulated as follows:

$$\mathcal{L}^{PPF,1} = \mathcal{L}_{\text{c},1} + \mathcal{L}_{CE,1}^{\text{PPF}} + \mathcal{L}_{\text{c},2} + \mathcal{L}_{CE,2}^{\text{PPF}}. \tag{6}$$

Finally, the total loss for the first stage is as follows:

$$\mathcal{L}_{total}^{1} = \mathcal{L}_{CPCM} + \mathcal{L}^{PPF,1}, \tag{7}$$

where $\mathcal{L}_{CPCM}$ is the backbone network loss, referring to [28] for detail.

3.2.2. Second Stage

Despite effective pseudo-label screening via PPF in the first stage, the overreliance on this single source for pseudo-labels hinders further improvements in network performance. To address this limitation, we introduce the Pseudo-Label Fusion(PLF) module to create an extra pseudo-label source and generate higher-quality pseudo-labels in the second stage. Specifically, we adopt a teacher-student training paradigm and train a new model from scratch. At this stage, we only apply pseudo-label supervision to the first branch P_1.

For teacher model. Considering the potential unreliability of labels generated by the model during training, we use the optimized first-stage model $f_\theta^t(\cdot)$ as a frozen teacher to supervise the new model. To further refine the pseudo-labels, we introduce PLF into the teacher model. The PLF takes the category probability distributions from two different enhanced branches of the teacher model as input, and ultimately outputs pseudo-labels as the supervisory signal for the student model. To filter the unreliable pseudo-labels of teachers, we enforce a prediction consistency check between two augmented views of the same input point cloud. Formally, let $y_1 = \text{argmax} f_\theta^t(P_1^u))$ and $y_2 = \text{argmax} f_\theta^t(P_2^u))$ denote the model predictions for the two perturbed data instances, respectively. The teacher model pseudo-label y_t is obtained as follows:

$$y^t = y_1, \quad \text{s.t.} \quad y_1 = y_2, \tag{8}$$

where y^t is set to $\emptyset$ if $y_1 \neq y_2$, thereby avoiding introducing noisy supervision into the training process.

Although the above methods select some high-quality pseudo-labels while filtering out unreliable ones based on consistency, they ignore the information contained in the predicted probability distribution $\mathbf{Z^{u,t}} = \text{Softmax}(f_\theta^t(P_1^u))$. Entropy quantifies the predictive uncertainty of predicted probability distribution; low entropy corresponds to predictions

with high certainty, while high entropy indicates ambiguous predictions. To further improve the quality of pseudo-labels, we filter them using their information entropy, which is calculated as follows:

$$\mathcal{H}_i = -\sum_{k=1}^{K} \mathbf{Z}_{\mathbf{i,k}}^{\mathbf{u,t}} \log_2\left(\mathbf{Z}_{\mathbf{i,k}}^{\mathbf{u,t}} + \epsilon\right), \tag{9}$$

where $\mathcal{H}_i$ denotes the i-th point in $\mathbf{Z}^{\mathbf{u,t}}$ information entropy, K represent the number of categories, respectively, and ϵ is a small constant to prevent numerical instability.

$$y_i^t = y_i^t, \quad \text{s.t.} \quad \mathcal{H}_i < a, \tag{10}$$

where y_i^t is set to $\emptyset$ if the entropy constraint $\mathcal{H}_i < a$ is not satisfied, thereby avoiding introducing uncertain supervision into the training process.

For student model. Two complementary sources constrain the student model during training. First, it is supervised using pseudo-labels refined by the teacher model through PLF, which can be regarded as an approximation of ground-truth annotations. These pseudo-labels provide relatively reliable supervisory signals and facilitate faster convergence at the early stage of student training. Second, the outputs of the student model are fed into the PPF to generate updated pseudo-labels. This mechanism not only introduces an adaptive pseudo-label refinement process but also helps mitigate potential errors introduced by teacher-generated supervision, thereby improving the overall robustness of the training process. These two constraint processes can be formulated as follows:

$$\mathcal{L}_{\mathrm{CE}}^{\mathrm{PLF}} = -\frac{1}{N_{\mathrm{valid}}^t} \sum_{y_i^t \neq \emptyset} y_i^t \log(\mathbf{Z}_{\mathbf{i}}^{\mathbf{u,s}}), \tag{11}$$

$$\mathcal{L}^{\mathrm{PPF},2} = \mathcal{L}_{\mathrm{c},1}^2 + \mathcal{L}_{CE,1}^{\mathrm{PPF},2}, \tag{12}$$

where y_i^{t} represents the ith pseudo-label generated by the PLF, $\mathbf{Z}_{\mathbf{i}}^{\mathbf{u,s}}$ denotes the ith predicted probability output by the student model, and N_{valid}^t denotes the number of samples with valid pseudo-labels (empty labels are ignored during calculation). $\mathcal{L}_{\mathrm{c},1}^2$ and $\mathcal{L}_{CE,1}^{\mathrm{PPF},2}$ represents the loss of the first enhanced branch PPF in the second stage. The total loss in the second stage is as follows:

$$\mathcal{L}_{total}^2 = \mathcal{L}_{CPCM} + \mathcal{L}_{CE}^{\mathrm{PLF}} + \mathcal{L}^{\mathrm{PPF},2}. \tag{13}$$

In summary, the final loss of the proposed MPG is expressed as:

$$\mathcal{L}_{total} = \mathcal{L}_{total}^1 + \mathcal{L}_{total}^2. \tag{14}$$

4. Experiments

4.1. Datasets

We conduct experiments on three widely adopted large-scale point cloud benchmarks: S3DIS [35], ScanNet-V2 [36] and XIAMEN-Buildings [37]. Among them, S3DIS and ScanNet-V2 are designed for indoor scene understanding, while XIAMEN-Buildings targets outdoor building environments.

S3DIS contains 271 rooms collected from six large indoor areas. Each sample represents a medium-sized room (approximately 20 × 15 × 5 m) with densely sampled 3D points. Every point is annotated with one of 13 semantic categories. To ensure a comprehensive evaluation, we strictly follow the official setting and adopt only Area-5 as the validation set to evaluate the performance.

ScanNet-V2 includes 1201 annotated indoor scans for training and 312 scans for validation. In the training set, each point is assigned one of 20 semantic classes. We use the validation set to evaluate the performance.

XIAMEN-Buildings is a large-scale outdoor point clouds dataset designed for 3D building facade semantic segmentation. It contains building facade scenes collected from the streets of Xiamen City. The dataset is divided into six areas and covers eight semantic classes. Following the original protocol in previous work [51], we use each area in turn as the test and validation set, while the other areas are used for training. For a fair comparison, we also report the average results over all six areas.

4.2. Implementation Details

Our method is implemented on PyTorch version 1.10.1 [52] with MinkowskiEngine version 0.5.4 [40], a sparse convolution library, following previous works [17,53]. Regarding the model architecture, a 34-layer Sparse Residual U-Net [54] is adopted, consistent with prior studies [55,56]. For optimization, the SGD optimizer is employed with lr = $1e^{-2}$, weight decay = $1e^{-3}$, and a polynomial learning rate scheduler with decay rate = 0.9. The batch size is set to 2. During training, the voxel size is set to 2 cm, 5 cm and 5 cm for ScanNet-V2, S3DIS and XIAMEN-Buildings, respectively. Our two-stage framework is trained for 300 epochs per stage (600 epochs in total). All experiments are conducted on 2 NVIDIA 3090 GPUs.

For sparse annotation sampling, we adopt class-wise even sampling following PSD [49], which allocates the annotation budget evenly across semantic classes to avoid bias toward dominant categories under extremely low annotation rates. For data augmentation, we follow the same strategy as CPCM [28], a dropout ratio of 0.2, a horizontal-flip probability of 0.95 with a per-axis flip probability of 0.5, an auto-contrast probability of 0.2, and a color-translation range ratio of 0.10 with a probability of 0.95 to the input point clouds during training.

4.3. Evaluation Metrics

The performance of the model is assessed on all points within the test dataset, utilizing Intersection-over-Union (IoU) and mean Intersection-over-Union (mIoU) as the evaluation metrics. K is defined as the total number of classes, and let TP, FP, and FN represent the numbers of true positives, false positives, and false negatives, respectively. The metrics are defined as follows:

$$\mathrm{IoU}_k = \frac{\mathrm{TP}_k}{\mathrm{TP}_k + \mathrm{FP}_k + \mathrm{FN}_k} \tag{15}$$

$$\mathrm{mIoU} = \frac{1}{K} \sum_{k=1}^{K} \frac{\mathrm{TP}_k}{\mathrm{TP}_k + \mathrm{FP}_k + \mathrm{FN}_k} \tag{16}$$

4.4. Quantitative Results on S3DIS

To evaluate the segmentation performance of MPG, we compare it with latest methods on Area-5 of the S3DIS indoor dataset. The detailed results are reported in Table 1. The results demonstrate that under different weak annotation ratios (i.e., 0.1%, 0.02%, and 0.01%), our method achieves competitive performance.

Table 1. Quantitative comparisons on S3DIS Area 5 under various weakly supervised settings. Bold indicates the superior performance under the same configuration.

Setting	Method	mIoU (%)	Ceiling	Floor	Wall	Beam	Column	Window	Door	Chair	Table	Bookcase	Sofa	Board	Clutter
100%	PointNet [38]	41.1	88.8	97.3	69.8	**0.1**	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
	SQN [53]	63.7	92.8	96.9	81.8	0.0	25.9	50.5	65.9	79.5	85.3	55.7	72.5	65.8	55.9
	HybridCR [29]	65.8	93.6	98.1	82.3	0.0	24.4	59.5	66.9	79.6	87.9	67.1	73.0	66.8	55.7
	MinkNet [40]	68.2	91.7	**98.7**	83.8	0.0	24.7	56.8	72.1	91.5	83.5	73.3	70.8	**81.3**	58.4
	DeepGCN [57]	60.0	90.8	97.5	76.7	0.0	24.9	51.4	52.7	76.7	83.0	61.1	62.2	58.5	44.6
	PointNeXt [58]	69.2	**94.7**	98.5	82.9	0.0	24.2	59.9	**74.3**	83.0	**91.4**	**76.3**	75.5	78.6	**60.4**
	PointTransV1 [59]	70.4	94.0	98.5	**86.3**	0.0	**38.0**	**63.4**	**74.3**	**89.1**	82.4	74.3	**80.2**	76.0	59.3
	PointTransV2 [60]	**71.6**	-	-	-	-	-	-	-	-	-	-	-	-	-
0.1%	SQN [53]	61.4	91.7	95.6	78.7	0.0	24.2	55.9	63.1	70.5	83.1	60.7	67.8	56.1	50.6
	PointMatch [61]	63.4	-	-	-	-	-	-	-	-	-	-	-	-	-
	AADNet [62]	67.2	93.7	98.0	81.5	0.0	19.4	**59.5**	72.0	80.9	88.5	**78.3**	73.0	72.1	56.1
	DGNet [63]	67.8	**94.3**	**98.4**	81.6	0.0	28.9	57.2	70.5	82.3	**90.7**	74.1	75.2	70.3	58.5
	CPCM [28]	66.3	91.4	95.5	82.0	0.0	30.8	54.1	70.1	87.6	79.4	70.0	67.0	77.8	56.6
	MPG(ours)	**69.5**	92.9	96.8	**83.8**	0.0	**41.1**	57.1	**73.8**	**88.2**	79.0	73.5	**76.9**	**82.5**	**57.6**
0.03%	PSD [49]	48.2	87.9	96.0	62.1	0.0	20.6	49.3	40.9	55.1	61.9	43.9	50.7	27.3	31.1
	HybridCR [29]	51.5	85.4	91.9	65.9	0.0	18.0	51.4	34.2	63.8	78.3	52.4	59.6	29.9	39.0
	DCL [64]	59.6	91.7	95.8	76.4	0.0	21.2	58.3	29.6	72.6	83.3	64.2	69.6	63.4	48.6
0.02%	MILTrans [22]	51.4	86.6	93.2	75.0	0.0	29.3	45.3	46.7	60.5	62.3	56.5	47.5	33.7	32.2
	ERDA [65]	48.4	87.3	**96.3**	61.9	0.0	11.3	45.9	31.7	73.1	65.1	57.8	26.1	36.0	36.4
	MulPro [31]	47.5	90.1	**96.3**	71.8	0.0	6.7	46.7	39.2	67.2	67.4	21.8	39.2	33.0	38.0
	CPCM [28]	62.3	**92.6**	95.6	79.4	0.0	17.8	49.3	59.4	85.7	75.6	69.1	60.7	68.2	55.8
	MPG(ours)	**65.1**	92.2	95.9	**81.6**	0.0	**34.3**	**51.4**	**64.5**	**87.2**	**76.3**	**71.7**	**65.5**	**68.4**	**57.8**
0.01%	SQN [53]	45.3	89.2	93.5	71.3	0.0	4.1	34.7	41.0	54.9	66.9	25.7	55.4	12.8	39.6
	AADNet [62]	60.8	92.5	96.6	77.2	0.0	20.9	**57.0**	61.1	72.2	83.1	60.1	67.8	52.9	49.0
	DGNet [63]	62.4	**93.3**	**98.1**	80.1	0.0	23.3	47.9	53.1	79.4	**87.2**	60.0	**70.6**	65.2	53.1
	CPCM [28]	59.3	-	-	-	-	-	-	-	-	-	-	-	-	-
	MPG(ours)	**63.5**	92.6	95.2	**82.2**	0.0	**33.5**	48.8	**61.8**	**82.2**	77.6	**63.6**	65.4	**65.7**	**55.3**

For the setting of 0.1%, MPG achieves an mIoU of 69.5%, which is 28.4%, 5.8%, 3.7% and higher than the fully-supervised methods PointNet, SQN and HybridCR. It also outperforms the AADNet and DGNet by 2.3% and 1.7%, respectively.

For the setting of 0.02%, Our MPG reaches an mIoU of 65.1%. We first compare our method with weakly supervised approaches reported under the 0.03% label setting. Since the supervision levels differ (0.02% for MPG vs. 0.03% for the compared methods), this comparison is indicative rather than direct. Under this indicative setting, MPG outperforms HybridCR by 13.6% and DCL by 5.5%. It also achieves 2.8% higher mIoU than the backbone CPCM. Compared with fully supervised methods. Our result is 24.0% and 5.1% higher than PointNet and DeepGCN respectively. We further evaluate against more advanced fully supervised networks, PointNeXt and PointTransV1. Our approach has only 4.1% and 5.3% performance gaps against the two models.

For 0.01% extremely sparse label setting, MPG achieves an overall average mIoU of 63.5%, significantly outperforming all recent comparison methods. Specifically, it exceeds the second-ranked DGNet by 1.1%, and achieves an 18.2% improvement over the baseline SQN. These results verify that our method effectively solves the severe supervision scarcity problem in weakly supervised point cloud segmentation, achieving remarkable performance even under the extremely sparse label setting.

Moreover, visual comparisons are provided in Figure 2. The panels sequentially illustrate the raw point clouds, ground truth (GT), MPG outputs, and the backbone results. The regions where our method surpasses the backbone are denoted by red circles. Benefiting from the multi-stage pseudo-label refinement, our method effectively suppresses noisy pseudo-labels. It can be seen that our MPG better recognizes windows, bookcase and columns with more accurate and complete predictions.

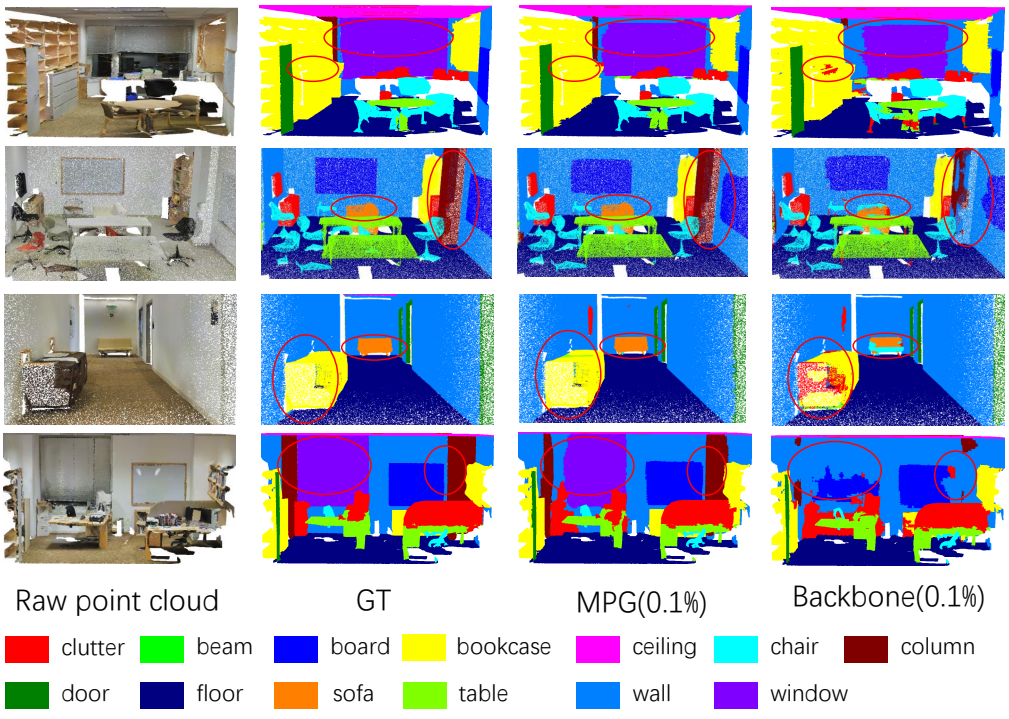


Figure 2. Visualization results on the test set of S3DIS Area-5. Raw point clouds, ground truth (GT), MPG and the backbone are presented separately, from left to right, and we use red circles to highlight where we outperform backbone.

In summary, our method, tested on Area-5 of the indoor dataset S3DIS, with 0.1%, 0.02% and 0.01% labeled data, outperforms the weakly supervised latest methods under the

same conditions and achieves results comparable to fully-supervised methods with only a small amount of labeled data.

4.5. Quantitative Results on ScanNet-V2

We further evaluated the performance of MPG on the ScanNet-V2 dataset, as shown in Table 2. Comprehensive experimental results on the ScanNet-V2 benchmark fully demonstrate that our proposed method effectively addresses the core challenge of insufficient supervision in weakly supervised point cloud semantic segmentation. Our method obtains state-of-the-art performance. We evaluate it under 1% and 0.1% weakly supervised label ratios. These results demonstrate the effectiveness and strong generalization ability of our approach.

Under the 1% weakly supervised setting, our proposed method achieves an mIoU of 68.2%, which ranks first among all comparison methods and shows significant superiority over existing approaches. Specifically, our method outperforms the second-ranked DGNet by 0.8%. It also consistently outperforms other mainstream weakly supervised methods including GaIA, ERDA, AADNet, and CPCM, demonstrating its superior effectiveness in sparse label scenarios.

Under the 0.1% extremely weakly supervised setting, our method still maintains outstanding performance, achieving an mIoU of 67.1%, which outperforms all comparison methods. It outperforms the second-ranked CPCM by 3.3%, and also outperforms SQN and EviUD by 8.7% and 7.4%, verifying its excellent robustness and generalization ability even under extremely sparse label conditions.

Additionally, Figure 3 illustrates the qualitative performance of our method on the ScanNet-V2 validation set. The visual comparison is organized as follows: raw point clouds, GT, baseline (backbone), and our MPG results (from left to right). Notably, the red circles indicate the specific regions. As observed, our MPG achieves superior recognition of sofas and chairs compared to the baseline.

Table 2. Quantitative comparisons on ScanNet-V2. Bold indicates the superior performance under the same configuration. * denotes results based on our reimplementation.

Setting	Method	mIoU (%)
100%	PointNet [38]	33.9
	HybridCR [29]	59.9
	PointNeXt [58]	71.2
	PointTransV2 [60]	**75.4**
1%	Zhang et al. [48]	51.1
	PSD [49]	54.7
	HybridCR [29]	56.8
	DCL [64]	59.6
	c3 [30]	58.1
	GaIA [66]	65.2
	ERDA [65]	63.0
	AADNet [62]	66.8
	DGNet [63]	67.4
	CPCM * [28]	65.4
	MPG(ours)	**68.2**
0.1%	EviUD [34]	59.7
	SQN [53]	58.4
	CPCM [28]	63.8
	MPG(ours)	**67.1**

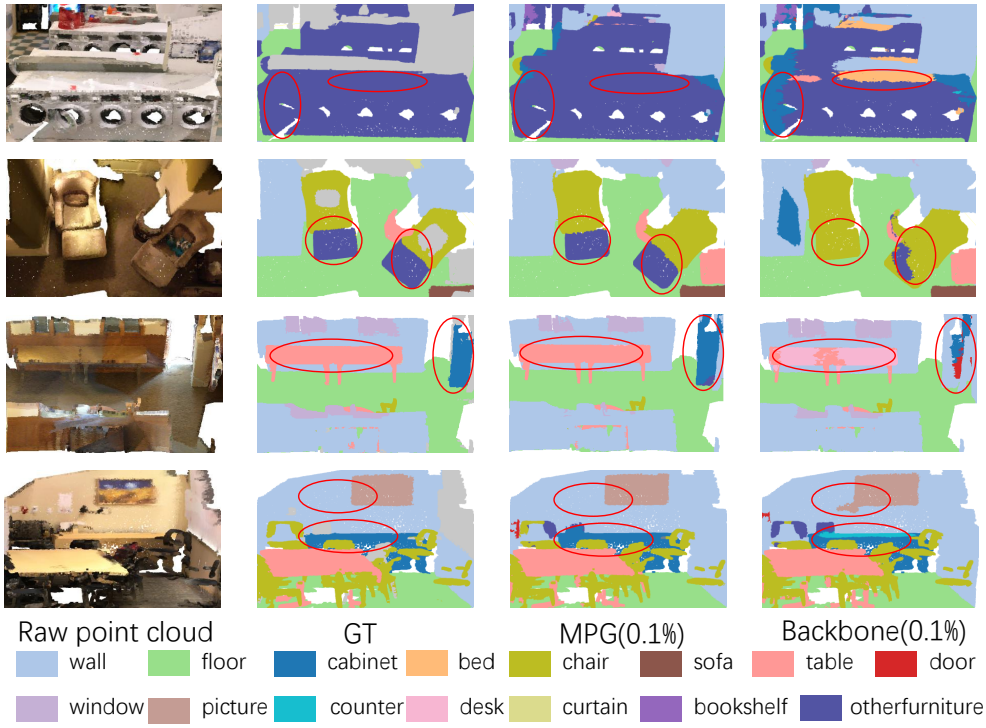


Figure 3. Visualization results on the validation set of Scannet-V2. Raw point clouds, GT, MPG and the backbone are presented separately, from left to right, and we use red circles to highlight where we outperform backbone.

As demonstrated above, our MPG achieves state-of-the-art performance on the ScanNet-V2 dataset under same settings. Benefiting from the high-quality pseudo-labels generated by the multi-stage refinement mechanism, our method even outperforms several fully supervised approaches.

4.6. Quantitative Results on XIAMEN-Buildings

To further evaluate the generalization performance of the proposed MPG, we conducted comparative experiments on the XIAMEN-Buildings dataset, which contains building facades with unique structural characteristics. We conducted six-fold cross-validation for fair comparison between MPG and existing methods. In this scheme, we use each area as the validation set in sequence and take the rest as training sets. Detailed test results are presented in Table 3, and it shows our approach outperforms recent weakly supervised methods under the same label ratio.

Under the 0.1% sparse label setting, our MPG method reaches an overall average mIoU of 60.2%, which ranks first among all weakly supervised methods, and achieves performance comparable to fully-supervised medtheds. Compared with the second-ranked UCL, our method improves the overall mIoU by 2.1%, and outperforms the baseline PSD by 8.0%.

Across 6 test areas, our method achieves the highest mIoU in 4 areas. It delivers prominent gains on complex building scenes and keeps competitive performance across all areas. These results verify that our method effectively alleviates the supervision scarcity problem in weakly supervised point clouds segmentation, achieving remarkable performance breakthroughs with extremely sparse labels.

Table 4 reports the per-class IoU on XIAMEN-Buildings under 6-fold cross-validation. MPG achieves the highest mIoU among weakly supervised methods (59.5%), surpassing the backbone CPCM (56.0%) by 3.5 points and approaching the fully supervised method, DLA-Net (59.7%), with only 0.1% annotations.

Table 3. Semantic segmentation performance on XIAMEN-Buildings. Bold indicates the superior performance under the same configuration. * denotes results based on our reimplementation.

Settings	Methods	Totall Average	mIoU (%)					
			Area 1	Area 2	Area 3	Area 4	Area 5	Area 6
100%	Point2Node [67]	54.6	63.1	63.8	56.5	48.8	52.7	42.5
	KPConv [68]	57.9	68.3	66.8	56.9	51.2	52.4	51.7
	SCF-Net [69]	59.1	66.5	62.5	56.8	53.7	62.2	**53.0**
	DLA-Net [51]	**60.3**	**68.2**	**66.9**	**58.4**	**54.3**	**63.6**	50.4
0.1%	PSD [49]	52.2	54.6	53.5	52.2	52.0	56.7	44.3
	SQN [53]	53.7	62.0	55.4	55.0	48.2	57.3	44.3
	DLA-Net [51]	54.0	61.1	59.1	53.7	48.4	57.2	44.3
	SCF-Net [69]	54.8	60.4	57.4	53.9	50.4	57.4	49.5
	SAF-C3 [37]	55.7	61.6	57.3	55.0	54.4	57.6	48.1
	UCL [70]	58.1	**65.0**	62.0	**57.9**	53.9	59.4	50.3
	CPCM * [28]	56.2	60.3	59.7	56.2	52.1	58.3	50.8
	MPG(ours)	**60.2**	62.3	**62.3**	57.7	**60.1**	**63.6**	**55.2**

Table 4. Semantic segmentation performance on XIAMEN-Buildings with Six-Fold Cross-Validation. Bold indicates the superior performance under the same configuration. * denotes results based on our reimplementation.

Settings	Methods	mIoU (%)	Balu.	Balc.	Advb.	Wall	Eave	Colu.	Wind.	Clutter
Fully	KPConv [68]	58.0	49.7	35.7	77.5	67.9	58.7	83.7	60.4	30.4
	Point2Node [67]	53.5	41.9	35.1	74	66.1	58.3	80.7	59.1	12.8
	SCF-Net [69]	58.9	55.4	40.8	81.0	66.8	61.3	79.4	61.8	25.0
	DLA-Net [51]	**59.7**	58.2	43.3	79.1	67.5	58.0	79.0	62.0	30.2
0.1%	SCF-Net [69]	54.3	53.4	36.7	76.5	61.6	54.6	76.2	56.0	19.7
	DLA-Net [51]	53.2	49.3	33.7	76.2	62.2	54.5	72.7	55.3	21.6
	PSD [49]	51.1	47.0	31.9	71.5	55.6	53.8	72.7	52.8	23.5
	SQN [53]	53.1	43.0	38.3	74.3	61.2	55.7	75.6	55.8	21.3
	SAF-C3 [37]	54.8	52.0	34.1	**76.7**	60.4	**57.3**	74.4	56.2	**27.9**
	MSC	56.5	56.3	38.7	79.7	62.4	55.6	75.7	58.4	25.0
	CPCM * [28]	56.0	54.3	43.7	74.2	61.7	51.5	73.2	64.3	24.8
	MPG(ours)	**59.5**	**55.7**	**46.5**	76.3	**69.1**	52.3	**81.9**	**68.4**	25.6

At the category level, MPG achieves the highest IoU on four of the eight classes: column (81.9%), window (68.4%), wall (69.1%), and balcony (46.5%). The largest margin is on column, where MPG exceeds the second-best method by 5.7 points (81.9% vs. 76.2%). These classes are geometrically regular and visually consistent across building instances, which makes prototype estimation stable and prototype-based filtering effective.

In contrast, the lowest IoU is observed on clutter (25.6%), a semantically ambiguous class whose prototype is poorly defined. On balustrade, advertising board, and eave, MPG does not achieve the highest IoU, indicating that these classes benefit from other design choices in competing methods. Overall, the results suggest that the effectiveness of MPG depends on the geometric regularity and semantic coherence of the class rather than on its sample frequency.

Figure 4 illustrates the qualitative results on the XIAMEN-Buildings Area5. From left to right, the columns display the raw point clouds, GT, MPG, and the backbone results. Red circles indicate regions where our approach achieves better segmentation than backbone. Notably, our method generates clearer and more precise segmentation boundaries

than the backbone network, effectively preserving the fine-grained structural details of building contours.

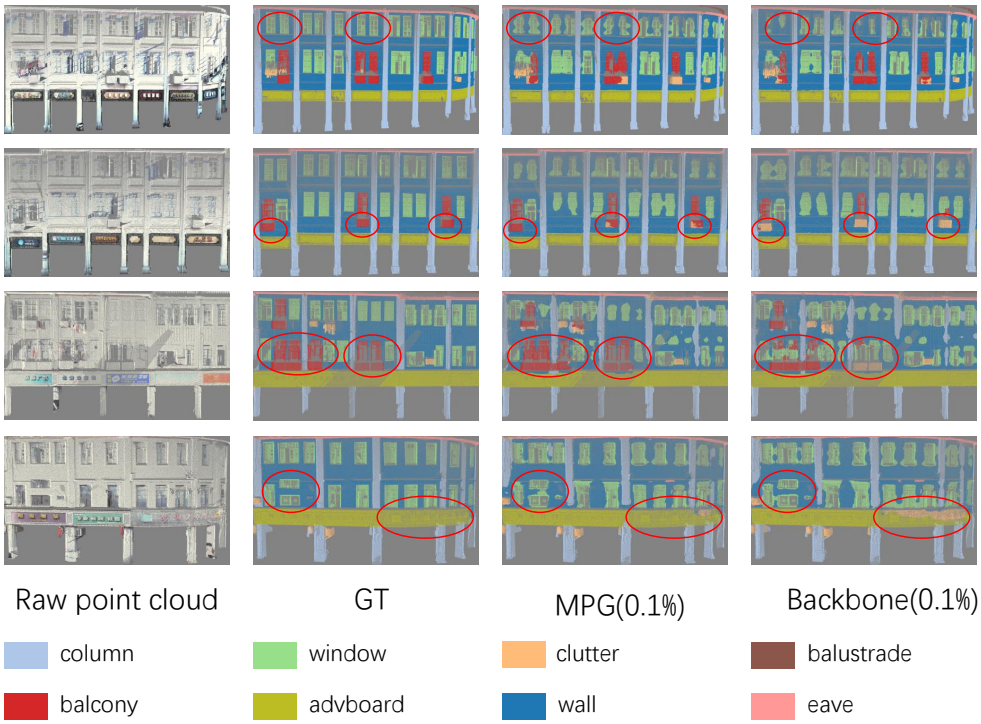


Figure 4. Visualization results on the Area-5 of XIAMEN-Buildings. Raw point clouds, GT, MPG and the backbone are presented separately, from left to right, and we use red circles to highlight where we outperform backbone.

5. Ablation Study

5.1. Ablation Study on Essential Component of MPG

To verify the contribution of each key module in MPG—namely the contrastive loss $\mathcal{L}_c^1 = \mathcal{L}_{c,1} + \mathcal{L}_{c,2}$, the loss $\mathcal{L}_{ce}^{PPF,1} = \mathcal{L}_{CE,1}^{\text{PPF}} + \mathcal{L}_{CE,2}^{\text{PPF}}$, and the second-stage training—a series of ablation studies were performed on Area-5 of S3DIS using 0.1% and 0.01% label ratio. Table 5 summarizes the outcomes. Among these, the configuration in row 6, which integrates all three components, achieves the best performance. It should be noted that the baseline results from the publicly available CPCM code are listed in row 1, whereas row 6 reports the performance of our proposed MPG. The other rows present the results of models using only individual or partial components.

Table 5. Ablation results of different components on Area-5 of S3DIS under 0.1% and 0.01% supervision. $\mathcal{L}_c^1$ and $\mathcal{L}_{ce}^{PPF,1}$ are losses in the PPF during the first stage. All results are reported as mean ± standard deviation over 3 independent runs with different random seeds.

	$\mathcal{L}_c^1$	$\mathcal{L}_{ce}^{PPF,1}$	Second-Stage Training	mIoU (%) @0.1%	mIoU (%) @0.01%
1	∘	∘	∘	66.53 ± 0.25	59.02 ± 0.27
2	✓	∘	∘	66.87 ± 0.25	59.47 ± 0.29
3	✓	✓	∘	67.70 ± 0.36	61.50 ± 0.32
4	∘	✓	∘	67.13 ± 0.31	60.83 ± 0.33
5	∘	∘	✓	67.63 ± 0.23	61.47 ± 0.25
6	✓	✓	✓	69.17 ± 0.31	63.50 ± 0.34

✓indicates the module is adopted; ∘ indicates the module is not adopted.

Effectiveness of Contrastive loss $\mathcal{L}_c^1$. Comparative experiments were carried out to assess the performance of the contrastive loss $\mathcal{L}_c^1$. These trials involved the baseline (row 1), the variant with only $\mathcal{L}_c^1$ (row 2), and the variant incorporating solely the $\mathcal{L}_{ce}^{PPF,1}$ (row 4). Based on the obtained mean mIoU scores, row 2 achieves 0.34% and 0.45% gain over row 1, while row 3 (PPF module) yields 0.57% and 0.67% improvement over row 4. These quantitative results demonstrate that $\mathcal{L}_c^1$ effectively complements other modules, leading to a consistent enhancement in segmentation accuracy.

Effectiveness of the $\mathcal{L}_{CE}^{PPF,1}$. The $\mathcal{L}_{CE}^{PPF,1}$ represents the cross-entropy loss calculated by PPF using pseudo-labels generated from class prototypes. Row 4, which only uses this strategy, shows 0.60% and 1.81% mIoU improvement compared to the baseline in row 1. Furthermore, row 3, incorporating both contrastive loss and $\mathcal{L}_{CE}^{PPF,1}$, improves by 0.83% and 2.03% compared to row 2. These results indicate that the $\mathcal{L}_{CE}^{PPF,1}$ significantly enhances the quality of pseudo-labels and improves overall model performance.

Effectiveness of the Second-stage training. The second-stage training further supplies pseudo-labels to the model, contributing to higher segmentation accuracy. To precisely quantify the impact of this module, comparative experiments were performed. The quantitative results reveal that row 5, which directly trains the model using the second-stage approach (i.e., the teacher model employs the original network without PPF-based training), yields 1.10% and 2.45% mIoU gain compared to row 1. Moreover, comparing the full framework (row 6) with the variant lacking second-stage training (row 3) reveals a significant 1.47% and 2.00% mIoU increase, which validates the capability of PLF in boosting segmentation accuracy. Notably, row 6 attains the peak mean segmentation accuracy of 69.17% and 63.50%, confirming that the other two components can further unleash the optimization potential of the PLF module.

5.2. Ablation Study on Different Entropy Thresholds

To investigate the impact of the information entropy threshold on the model performance, we conduct ablation studies with different threshold values, and the results are summarized in Table 6.

Table 6. Ablation studies on different entropy thresholds on S3DIS (0.1%).

∅	2.3	2.5	2.7	3.0
68.0%	68.3%	68.7%	69.5%	68.9%

∅ denotes that no information entropy threshold is used for filtering.

Effectiveness of the entropy threshold filter. The baseline setting, where no entropy threshold filtering is applied (denoted by ∅), yields a model accuracy of 68.0%. When we introduce the entropy threshold filtering with a threshold of 2.3, the accuracy immediately rises to 68.3%, demonstrating the initial effectiveness of filtering out low-confidence samples with high entropy. As we further increase the threshold to 2.5 and 2.7, the model performance continues to improve, reaching the highest accuracy of 69.5% at the threshold of 2.7. This corresponds to a 1.5% absolute improvement over the baseline, which validates that the entropy threshold can effectively filter out noisy samples and enhance the quality of pseudo labels for model training.

However, we observe that the model performance starts to decline when the threshold is set too high: with a threshold of 3.0, the accuracy drops to 68.9%. This is because an overly permissive entropy threshold retains a large number of high-entropy, unreliable pseudo-labels that pass the consistency check but carry noisy supervision. This noise propagates into the fused pseudo-labels and degrades the model's segmentation performance. A threshold

of 2.7 provides a better balance between retaining sufficient pseudo-labels and suppressing unreliable ones.

Based on these ablation results, we select 2.7 as the optimal information entropy threshold for our final model, which achieves the better trade-off between filtering out noisy samples and retaining sufficient valid training data.

5.3. Ablation Study on Different PPF Confidence Threshold τ

To investigate the impact of the information confidence threshold on the model performance, we conduct ablation studies with different threshold values, and the results are summarized in Table 7.

Table 7. Ablation studies on PPF confidence threshold τ on S3DIS (0.1%).

$\emptyset$	**0.7**	**0.8**	**0.9**	**0.95**
65.2%	66.8%	67.2%	67.8%	66.5%

$\emptyset$ denotes that no threshold is used for filtering. All results are obtained under the single-stage setting (Stage I only, with PLF disabled) to isolate the effect of the PPF confidence threshold.

Effectiveness of the τ in PPF. We conduct this ablation under the first stage setting to isolate the effect of PPF, as evaluating it within the full two-stage pipeline would conflate its contribution with that of PLF and the second training stage. Table 7 reports the effect of the PPF confidence threshold τ. Without filtering, the model achieves 65.2% mIoU. Applying a threshold consistently improves performance, confirming that removing low-confidence pseudo-labels is beneficial. As τ increases from 0.7 to 0.9, mIoU improves monotonically from 66.8% to 67.8%, indicating that stricter filtering yields cleaner pseudo-labels and better supervision. However, further increasing τ to 0.95 degrades performance to 66.5%, as an overly strict threshold discards a large number of valid pseudo-labels, leaving insufficient supervision for training. We therefore set $\tau = 0.9$, which achieves the best balance between pseudo-label purity and retention.

5.4. Ablation Study on the Contrastive Loss Weight

Effectiveness of the contrastive loss weight. We denote the weight of the contrastive loss as α. Table 8 reports the sensitivity of α under 0.1% supervision on S3DIS. The performance is highest at $\alpha = 1.0$ (69.5%) and drops slightly at both lower and higher values: 69.1% at $\alpha = 0.5$ and 68.9% at $\alpha = 2.0$. At $\alpha = 0.5$, the contrastive loss provides insufficient guidance for aligning features with class prototypes, weakening the effectiveness of PPF filtering. At $\alpha = 2.0$, the contrastive loss dominates the optimization and slightly degrades performance. The default value $\alpha = 1.0$ therefore achieves the best balance among the loss terms.

Table 8. Sensitivity analysis of the contrastive loss weight α on S3DIS under 0.1% supervision. The other loss terms are kept at their default values.

α	0.5	1.0	2.0
mIoU (%)	69.1	69.5	68.9

5.5. Ablation Study on Consistency Filter and PPF

To validate the effectiveness of our proposed core components on the S3DIS dataset with an extremely low annotation rate of 0.1%, we conduct comprehensive ablation studies on the consistency filter and the PPF module, with the results summarized in Table 9.

The baseline model without any filtering module achieves an accuracy of 67.1, which serves as the reference for evaluating the contribution of each component.

Effectiveness of the consistency filter. The consistency filter is designed to purify the pseudo label set by removing samples with inconsistent predictions from different data views, which effectively reduces the risk of introducing noisy supervision into the model training process. When only the consistency filter is enabled, the model accuracy improves from 67.1% to 67.4%, achieving a 0.3% performance gain over the baseline. Although the individual improvement of the consistency filter is relatively modest in this low-annotation scenario, it plays a critical complementary role when combined with the PPF module, as it further refines the pseudo label set after the initial confidence-based filtering.

Table 9. Ablation studies on consistency filter and PPF on S3DIS (0.1%).

Consistency Filter	PPF	mIoU (%)
∘	∘	67.1
✓	∘	67.4
∘	✓	68.4
✓	✓	69.5

PPF means the module used in the second stage. ✓indicates the module is adopted; ∘ indicates the module is not adopted.

Effectiveness of the PPF on second stage. The PPF filters low-confidence pseudo labels based on the distance between unlabeled data and class prototypes, which directly targets the reliable samples in the unlabeled dataset. When only the PPF is enabled, the model accuracy rises significantly from 67.1% to 68.4%, achieving a 1.3% improvement over the baseline, which outperforms the individual contribution of the consistency filter by a large margin. The result reveals that pseudo-label quality plays a key role at extremely low annotation rates, and PPF alleviates performance degradation by generating high-quality pseudo labels.

More importantly, when both the consistency filter and the PPF are enabled simultaneously, the model reaches the highest accuracy of 69.5%, which is 2.4% higher than the baseline model, 2.1% higher than the setting with only the consistency filter, and 1.1% higher than the setting with only the PPF module. Our method achieves notable performance gains. This proves the cooperative effect between the two modules. First, PPF generate high-quality pseudo labels. Then the consistency filter eliminates samples with inconsistent multi-view predictions. Jointly, they purify pseudo-labels respectively in different ways. This greatly improves performance for 3D scene segmentation under sparse annotations.

5.6. Ablation Study on Different Branch Constraints

To explore the influence of two augmentation branches ($\mathbf{P}_1$ and $\mathbf{P}_2$) with identical augmentation implementation on model training via pseudo-label supervision, we conduct ablation experiments on the S3DIS dataset for the second training stage, as listed in Table 10.

Table 10. Ablation results of different branch constraints on the S3DIS dataset in the second stage.

$\mathbf{P}_1$	$\mathbf{P}_2$	mIoU (%)
∘	✓	68.9
✓	∘	69.5
✓	✓	67.6

✓indicates the module is adopted; ∘ indicates the module is not adopted.

Effectiveness of constraints for different branches. When only $\mathbf{P}_2$ is activated and $\mathbf{P}_1$ is disabled, the model obtains an accuracy of 68.9%. Replacing the active branch to sole $\mathbf{P}_1$ improves the performance to 69.5%. This performance gap arises because the teacher model leverages class prototypes from $\mathbf{P}_1$ for supervision. As a result, these prototypes provide less reliable guidance for the $\mathbf{P}_2$ branch, thereby contributing to the performance advantage of $\mathbf{P}_1$ over $\mathbf{P}_2$ when employed independently. This indicates that the pseudo-label supervision derived from the $\mathbf{P}_1$ branch brings more favorable optimization gains for the segmentation model, despite the two branches sharing the same augmentation mechanism. However, enabling both $\mathbf{P}_1$ and $\mathbf{P}_2$ simultaneously reduces the accuracy to 67.6%. One possible explanation is that the two augmented views produce inconsistent prediction distributions, which may introduce conflicting supervisory signals during cross-entropy optimization and affect the quality of pseudo-label training. However, we do not provide direct evidence for this hypothesis and leave a detailed analysis as future work.

In conclusion, two augmentation branches with identical augmentation strategies cannot be synchronously adopted for pseudo-label cross-entropy constraint, and we select the $\mathbf{P}_1$ branch alone for the student model to achieve optimal performance.

5.7. Ablation Study on Training Epoch and Framework Effects

To separate the contribution of our modules from additional training and the teacher-student framework itself, we conducted two controlled comparisons. As listed in Table 11. All methods share the same backbone, annotation setting (0.1% class-wise even sampling), data augmentation, optimizer, and batch size.Crucially, the plain teacher-student baseline uses the sameteacher as MPG (the Stage-I model), and its student is re-initialized in the same way. The only difference is that the teacher's predictions are used directly as pseudo-labels without PPF or PLF.

Table 11. Comparison of different training budgets and frameworks under 0.1% supervision on S3DIS. Training time is reported in aggregate GPU-hours, calculated as the number of GPUs (2 × NVIDIA 3090) multiplied by the elapsed training time. Results are reported as mean ± standard deviation over 3 independent runs with different random seeds.

Method	Total Epochs	GPU-Hours	mIoU (%)
CPCM (original)	180	4.55	66.53 ± 0.25
CPCM (extended)	600	15.17	66.73 ± 0.23
Teacher model (MPG Stage I)	300	7.75	67.70 ± 0.36
Plain T-S PL (student)	600 (300 + 300)	15.33	67.07 ± 0.21
MPG (ours)	600 (300 + 300)	16.83	69.17 ± 0.31

First, extending CPCM training from 180 to 600 epochs yields only a very slight improvement (66.53% → 66.73%, +0.20%), indicating that extending the training epochs alone does not lead to a meaningful performance enhancement.

Second, the plain teacher-student baseline achieves 67.07% mIoU, which is below its teacher (67.70%). This indicates that naively propagating unfiltered pseudo-labels to a freshly initialized student degrades performance: the noise in the teacher's predictions is transferred to the student, outweighing any benefit of the additional training stage.

In contrast, MPG achieves 69.17% under the same total budget, a 2.10% improvement over the plain teacher-student baseline and a 1.47% improvement over its own teacher. Since the two share the same teacher, the same student initialization, and the same training budget, this gain is directly attributable to the proposed PPF and PLF modules.

5.8. Quantitative Analysis of Pseudo-Label Quality

To provide direct evidence for the mechanism of PPF and PLF, we measure the accuracy of PPF and PLF pseudo-labels, sampling once every 50 epochs. The pseudo-label accuracy is computed as the proportion of pseudo-labeled points whose predicted class matches the ground-truth class, which is available for evaluation purposes but not used during training.

Table 12 reports the results. For PPF, the accuracy of PPF pseudo-labels increases steadily during training, from 35.75% at epoch 1 to 94.19% at epoch 300. The retention rate follows a different pattern: it rises from 0.5% at epoch 1 to a peak of 10.5% at epoch 50, then declines to 0.7% by epoch 200 and remains stable thereafter. This reflects the evolution of the prototype-based filter: in the early stage, prototypes estimated from sparse annotations are unreliable, so only a small fraction of pseudo-labels pass the similarity threshold. As training proceeds, prototypes become more stable, allowing more pseudo-labels to be retained. In the later stage, the filter becomes more selective, retaining fewer but more accurate pseudo-labels.

Table 12. Pseudo-label accuracy (%) and retention rate (%) at different training epochs on S3DIS (0.1%).

Epoch	PPF		PLF		w/o PLF	
	Acc	Ret.	Acc	Ret.	Acc	Ret.
1	35.75	0.5	99.71	0.3	94.55	100
50	82.60	10.5	99.69	0.3	94.45	100
100	86.59	9.6	99.71	0.3	94.56	100
150	88.47	5.2	99.78	0.3	94.59	100
200	94.24	0.7	99.69	0.3	94.63	100
250	94.92	0.7	99.69	0.3	94.55	100
300	94.19	0.7	99.76	0.3	94.68	100

For PLF, the pseudo-label accuracy remains consistently above 99.6% throughout training, with only minor fluctuations (from 99.69% to 99.78%). This is expected because PLF operates on the pseudo-labels generated by the frozen teacher model, which has already been fully trained in Stage I. The high accuracy reflects the reliability of the teacher's predictions. The small fluctuations are attributable to the data augmentation applied to different training batches: since each batch is augmented differently, the pseudo-labels generated for a given point may vary slightly across batches, while the teacher's parameters remain frozen and unchanged.

However, the retention rate of PLF is only 0.3% throughout training. This indicates that the fusion strategy is highly conservative: it retains only pseudo-labels on which the multiple sources strongly agree, resulting in near-perfect accuracy at the cost of very few pseudo-labels. The stability of the retention rate (0.3% at all epochs) is consistent with the frozen teacher—since the teacher's parameters do not change, the set of pseudo-labels it generates remains stable, and the agreement between sources does not vary substantially across training.

To isolate the effect of PLF, we also report the pseudo-label accuracy without fusion (w/o PLF), i.e., using the teacher's predictions directly. Without fusion, the accuracy remains around 94.5%, compared with above 99.6% with PLF. This 5.1% gap indicates that PLF improves pseudo-label quality by applying strict dual-gate screening: it retains only pseudo-labels that pass both gates, trading a smaller number of pseudo-labels for higher quality. The accuracy of the retained pseudo-labels is therefore higher than that of the unfused teacher predictions, at the cost of a reduced retention rate.

The combination of high accuracy and low retention suggests that PLF alone would provide insufficient supervision for training. Its role is therefore complementary to PPF: PPF provides a larger set of pseudo-labels with moderate accuracy, while PLF provides a smaller set with very high accuracy. Together, they allow the student to learn from both a broad and a precise source of supervision.

6. Discussion

6.1. Elation to Prior Weakly Supervised Methods

As reviewed in Section 2, weakly supervised point cloud segmentation methods can be grouped into four routes: 2D label-based methods, self-training methods, perturbation consistency-based methods, prototype-guided and teacher-student pseudo-label learning methods. Our method differs from these in the following respects.

Compared with 2D label-based methods, which generate pseudo-labels by projecting 2D image annotations onto 3D point clouds, MPG does not rely on external 2D supervision. Projection-based schemes lose part of the point cloud information and are subject to viewpoint restrictions, as noted in Section 2. MPG avoids these limitations by generating pseudo-labels directly in 3D space from a frozen teacher.

Compared with self-training methods, which iteratively refine pseudo-labels from the model's own predictions, MPG does not reuse its own predictions as the sole source of pseudo-labels. Self-training is prone to error accumulation, as the model's systematic errors are repeatedly reinforced. In MPG, pseudo-labels are generated by a frozen teacher and filtered through class-level prototype statistics, which reduces this accumulation.

Compared with perturbation consistency-based methods, which enforce consistency between augmented views as a training objective, MPG uses consistency as a criterion for filtering pseudo-labels. Perturbation consistency methods do not generate explicit pseudo-labels; they directly constrain the model's predictions to be invariant across perturbations. In contrast, MPG generates pseudo-labels and retains only those that pass the consistency check, which are then used as supervision for the student. This distinguishes MPG from consistency-based methods in that the consistency constraint operates at the label level rather than at the prediction level.

Compared with prototype-guided methods, which use class prototypes as regularization for feature learning—pulling point features toward their class prototype and away from others, aligning prototypes of the same class across scenes, or using multiple prototypes per class as classifier weights—MPG uses the class prototype as a semantic anchor to generate and filter pseudo-labels. This shifts the role of prototypes from feature-level regularization to label-level refinement, which is more directly aligned with the goal of improving pseudo-label quality under extremely sparse supervision.

Compared with teacher-student pseudo-label learning methods, which obtain pseudo-labels from a single source—typically an exponential-moving-average teacher with confidence-based or uncertainty-based filtering—MPG obtains pseudo-labels from multiple sources (PPF, PLF). Single-source pseudo-labeling is prone to information bias and error accumulation, as the teacher's systematic errors are repeatedly reinforced. Multi-source fusion mitigates this by combining complementary predictions, which we attribute to the more stable gains of MPG under extremely sparse.

6.2. Why MPG Performs Differently

The improvement of MPG over the backbone and over single-source pseudo-labeling methods stems from two design choices: Prototype-guided Pseudo-label Filter and multi-source pseudo-label fusion.

Prototype-guided Pseudo-label Filter. Class prototypes are computed from the labeled points and serve as semantic anchors for pseudo-label generation. For each unlabeled point, MPG computes its feature similarity to each class prototype, and pseudo-labels are assigned only when the similarity exceeds a threshold. Since the prototypes are derived from the labeled points—the only source of ground-truth supervision under extremely sparse annotation—they carry reliable class-level semantic information. By using class prototypes as anchors, PPF leverages this reliable semantic information to identify unlabeled points that can be confidently assigned to a class.

Multi-source pseudo-label fusion. Single-source pseudo-labeling methods, such as DBCL, CASC, and EviUD, obtain pseudo-labels from one teacher and are prone to information bias: systematic errors in the teacher are repeatedly reinforced during training. MPG instead generates pseudo-labels from two different sources that rely on different mechanisms. The first source is generated by the student model using PPF, which assigns pseudo-labels based on the feature similarity between unlabeled points and class prototypes. The second source is generated by the teacher model through PLF, which fuses predictions from multiple augmented views. Since the two sources rely on different criteria, their predictions are complementary, and errors from one source are less likely to appear in the other. The two sources are then fused to produce the final pseudo-labels for training. As shown in Table 11, MPG outperforms the plain teacher-student baseline by 2.6% under the same teacher and training budget, confirming that the gain is attributable to the fusion of complementary sources rather than to the teacher itself.

6.3. Computational Efficiency

Table 13 compares the computational cost of MPG with the CPCM backbone on S3DIS. The results show that MPG improves segmentation accuracy without increasing model size or inference cost.

Table 13. Computational efficiency comparison between CPCM and MPG on S3DIS. Inference metrics are measured on a single NVIDIA 3090 GPU with batch size 2.

Metric	CPCM	MPG
Parameters (M)	37.85	37.85
Inference speed (FPS)	5.79	5.79
GPU utilization (inference)	44%	44%

Model parameters. The parameter count of MPG is identical to that of CPCM (37.85M), because the PPF and PLF modules do not introduce additional learnable parameters. PPF operates on class prototypes computed from feature embeddings, and PLF is a pseudo-label fusion strategy applied during training; neither adds parameters to the network.

Inference efficiency. At inference time, MPG achieves the same speed as CPCM (5.79 FPS) and the same GPU utilization (44%). This is because both modules operate only during training: PPF filters the pseudo-labels generated by the teacher, and PLF fuses pseudo-labels for the student. Neither is used in the forward pass at inference. The inference pipeline of MPG is therefore identical to that of the backbone, and the improvement in segmentation accuracy comes at no additional inference cost.

Training cost. The additional cost of MPG is confined to training. The two-stage architecture requires 300 epochs per stage (600 in total), compared with 180 epochs for CPCM. As shown in Table 11, this increase stems from the two-stage training pipeline rather than from the PPF and PLF modules, which are lightweight operations on pseudo-labels.

Engineering practicability. From a deployment perspective, these results indicate that MPG can be applied to large-scale building scene segmentation without increasing model

size, inference latency, or GPU utilization. The improvement in segmentation accuracy under extremely sparse supervision is achieved purely by improving the training process, which makes MPG suitable for practical deployment where inference efficiency is critical.

6.4. Limitation

MPG does not specifically address class imbalance or semantically ambiguous classes such as clutter; its benefit depends on the reliability of prototype estimation, which is limited when annotated points are scarce. Additionally, The two-stage training introduces additional computational cost compared with single-stage methods.

7. Conclusions

In this work, we propose a novel Multi-stage Pseudo-label Guided (MPG) framework for extremely sparse weakly supervised point cloud semantic segmentation. To address the challenge of unreliable pseudo-labels, MPG progressively mines and refines semantic information from unlabeled points through two stages. The proposed Prototype-guided Pseudo-label Filter (PPF) and Pseudo-Label Fusion (PLF) strategy effectively improve pseudo-label quality and enhance segmentation performance. Extensive experiments on indoor and outdoor benchmarks demonstrate that MPG achieves state-of-the-art performance under extremely sparse annotation settings and achieves competitive performance compared with some fully supervised methods. In future work, we will explore multi-modal feature fusion to further exploit complementary semantic information for more accurate point cloud understanding.

Author Contributions: Conceptualization, Y.S.; methodology, Y.S.; software, Y.S., Y.W. and T.M.; validation, Y.S. and Y.W.; formal analysis, Y.S. and J.P.; investigation, Y.S. and Y.W.; resources, Y.S.; data curation, Y.S.; writing—original draft preparation, Y.S. and Y.W.; writing—review and editing, Y.S., T.M. and J.P.; visualization, Y.S. and Y.W.; supervision, J.P.; project administration, J.P.; funding acquisition, J.P. All authors have read and agreed to the published version of the manuscript.

Funding: Thiswork was supported by the Natural Science Foundation of Xiamen, China (3502Z202472024), Fujian Provincial Natural Science Foundation of China (2025J011280) and National Natural Science Foundation of China under Grant (42501513).

Data Availability Statement: The original data presented in this study are openly available from S3DIS, ScanNet-V2 and XIAMEN-Buildings at http://buildingparser.stanford.edu/ (accessed on 28 September 2025), https://kaldir.vc.in.tum.de/scannet_benchmark/ (accessed on 20 December 2025) and https://drive.google.com/drive/folders/1cZEUnyF3jn0UnQNrhlZCkVjb54XzRTBd?hl=zh-CN (accessed on 17 March 2026).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MPG	Multi-stage Pseudo-label Guided training framework
PPF	Prototype-guided Pseudo-label Filter
PLF	Pseudo-label Fusion
SOTA	state-of-the-art
GT	Ground Truth
CE	cross-entropy loss
IoU	Intersection over Union
mIoU	mean Intersection over Union

References

1. Lyu, M.; Li, Y.; Chen, X. KL-CLIP: A K-means learning model for zero-shot anomaly segmentation. *Multimed. Syst.* **2025**, *31*, 294. [CrossRef]
2. Wang, J.; Yu, Y.; Cheng, J.; Li, J.; Tang, J. PillarBAPI: Enhancing pillar-based 3D object detection through attentive pseudo-image feature extraction: J. Wang et al. *Multimed. Syst.* **2025**, *31*, 263.
3. Zhuang, M.; Zhong, X.; Gu, D.; Feng, L.; Zhong, X.; Hu, H. LRDNet: A lightweight and efficient network with refined dual attention decorder for real-time semantic segmentation. *Neurocomputing* **2021**, *459*, 349–360. [CrossRef]
4. Zhang, M.; Li, W.; Zhao, Y.; Si, X.; Zhang, J. A Robust 3D Mesh Segmentation Algorithm With Anisotropic Sparse Embedding. *Comput. Animat. Virtual Worlds* **2025**, *36*, e70042. [CrossRef]
5. Wang, D.H.; Li, J.; Zhu, S. Few-labeled visual recognition for self-driving using multi-view visual-semantic representation. *Neurocomputing* **2021**, *428*, 361–367. [CrossRef]
6. Xing, Z.; Meng, Z.; Zheng, G.; Yang, L.; Guo, X.; Tan, L.; Jiang, Y. Human-computer interactive rehabilitation: A 3D graph deep learning method for non-contact gesture recognition in post-epidemic and aging societies. *Measurement* **2025**, *257*, 118794. [CrossRef]
7. Ren, Z.; Misra, I.; Schwing, A.G.; Girdhar, R. 3d spatial recognition without spatially labeled 3d. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 13204–13213.
8. Cheng, R.; Razani, R.; Taghavi, E.; Li, E.; Liu, B. 2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12547–12556.
9. Graham, B.; Engelcke, M.; Van Der Maaten, L. 3d semantic segmentation with submanifold sparse convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 9224–9232.
10. Zhang, M.; Zhou, J.; Miao, T.; Zhao, Y.; Si, X.; Zhang, J. Joint-Learning: A Robust Segmentation Method for 3D Point Clouds Under Label Noise. *Comput. Animat. Virtual Worlds* **2025**, *36*, e70038. [CrossRef]
11. Lian, J.; Du, X.; Liu, J.; Hui, L.; Yang, J. Cross-Modal Driven Object Restoration for 3D Point Cloud Backdoor Defense. *IEEE Trans. Inf. Forensics Secur.* **2025**, *20*, 11006–11018. [CrossRef]
12. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. Learning semantic segmentation of large-scale point clouds with random sampling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 8338–8354. [CrossRef]
13. Le, T.; Duan, Y. Pointgrid: A deep network for 3d shape understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9204–9214.
14. Xu, J.; Zhang, R.; Dou, J.; Zhu, Y.; Sun, J.; Pu, S. Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 16024–16033.
15. Zhuang, Z.; Li, R.; Jia, K.; Wang, Q.; Li, Y.; Tan, M. Perception-aware multi-sensor fusion for 3d lidar semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 16280–16290.
16. Wei, J.; Lin, G.; Yap, K.H.; Hung, T.Y.; Xie, L. Multi-path region mining for weakly supervised 3D semantic segmentation on point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 4384–4393.
17. Xu, X.; Lee, G.H. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13706–13715.
18. Weng, W.; Chen, Y.N.; Chen, C.L.; Wu, S.X.; Liu, J.H. Non-sparse label specific features selection for multi-label classification. *Neurocomputing* **2020**, *377*, 85–94. [CrossRef]
19. Liu, L.; Cao, J.; Liu, M.; Guo, Y.; Chen, Q.; Tan, M. Dynamic extension nets for few-shot semantic segmentation. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 1441–1449.
20. Liu, M.; Zhou, Y.; Qi, C.R.; Gong, B.; Su, H.; Anguelov, D. Less: Label-efficient semantic segmentation for lidar point clouds. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 70–89.
21. Liu, Z.; Qi, X.; Fu, C.W. One thing one click: A self-training approach for weakly supervised 3d semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 1726–1736.
22. Yang, C.K.; Wu, J.J.; Chen, K.S.; Chuang, Y.Y.; Lin, Y.Y. An mil-derived transformer for weakly supervised point cloud segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11830–11839.

23. Wang, H.; Rong, X.; Yang, L.; Wang, S.; Tian, Y. Towards Weakly Supervised Semantic Segmentation in 3D Graph-Structured Point Clouds of Wild Scenes. In Proceedings of the BMVC, Cardiff, UK, 9–12 September 2019; Volume 284.
24. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
25. Kweon, H.; Yoon, K.J. Joint learning of 2d-3d weakly supervised semantic segmentation. In *Advances in Neural Information Processing Systems 35*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2022; pp. 30499–30511.
26. Hong, C.; Yu, J.; You, J.; Chen, X.; Tao, D. Multi-view ensemble manifold regularization for 3D object recognition. *Inf. Sci.* **2015**, *320*, 395–405. [CrossRef]
27. Jiang, R.; Yan, Y.; Xue, J.H.; Chen, S.; Wang, N.; Wang, H. Knowledge distillation meets label noise learning: Ambiguity-guided mutual label refinery. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *36*, 939–952. [CrossRef]
28. Liu, L.; Zhuang, Z.; Huang, S.; Xiao, X.; Xiang, T.; Chen, C.; Wang, J.; Tan, M. Cpcm: Contextual point cloud modeling for weakly-supervised point cloud semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 18413–18422.
29. Li, M.; Xie, Y.; Shen, Y.; Ke, B.; Qiao, R.; Ren, B.; Lin, S.; Ma, L. Hybridcr: Weakly-supervised 3d point cloud semantic segmentation via hybrid contrastive regularization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 14930–14939.
30. Zhang, Y.; Lan, Y.; Xie, Y.; Li, C.; Qu, Y. Cross-cloud consistency for weakly supervised point cloud semantic segmentation. *IEEE Trans. Neural Netw. Learn. Syst.* **2025**, *36*, 14452–14463. [CrossRef]
31. Su, Y.; Xu, X.; Jia, K. Weakly supervised 3D point cloud segmentation via multi-prototype learning. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 7723–7736. [CrossRef]
32. Zhang, X.; Hu, Y.; Li, Y.; Nan, Z.; Yu, Q. Dual-Branch Point Cloud Semantic Segmentation: An EMA-Based Teacher–Student Collaborative Learning Framework. *Sensors* **2026**, *26*, 450. [CrossRef]
33. Wu, J.; Sun, M.; Xu, H.; Jiang, C.; Ma, W.; Zhang, Q. Class agnostic and specific consistency learning for weakly-supervised point cloud semantic segmentation. *Pattern Recognit.* **2025**, *158*, 111067. [CrossRef]
34. Wang, Q.; Wang, Y.; Zhang, J.; Wang, Y.; Kang, S. Evidentially Driven Uncertainty Decomposition for Weakly Supervised Point Cloud Semantic Segmentation. *ISPRS Int. J. Geo-Inf.* **2026**, *15*, 167. [CrossRef]
35. Armeni, I.; Sener, O.; Zamir, A.R.; Jiang, H.; Brilakis, I.; Fischer, M.; Savarese, S. 3d semantic parsing of large-scale indoor spaces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1534–1543.
36. Dai, A.; Chang, A.X.; Savva, M.; Halber, M.; Funkhouser, T.; Nießner, M. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5828–5839.
37. Su, Y.; Cheng, M.; Yuan, Z.; Liu, W.; Zeng, W.; Zhang, Z.; Wang, C. Spatial adaptive fusion consistency contrastive constraint: Weakly supervised building facade point cloud semantic segmentation. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5703214. [CrossRef]
38. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
39. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Proceedings of the Advances in Neural Information Processing Systems 30, Long Beach, CA, USA, 4–9 December 2017; p. 10.
40. Choy, C.; Gwak, J.; Savarese, S. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3075–3084.
41. Liu, Y.; Fan, B.; Meng, G.; Lu, J.; Xiang, S.; Pan, C. Densepoint: Learning densely contextual representation for efficient point cloud processing. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5239–5248.
42. Wang, B.H.; Chao, W.L.; Wang, Y.; Hariharan, B.; Weinberger, K.Q.; Campbell, M. Ldls: 3-d object segmentation through label diffusion from 2-d images. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2902–2909. [CrossRef]
43. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
44. Wang, H.; Rong, X.; Yang, L.; Feng, J.; Xiao, J.; Tian, Y. Weakly supervised semantic segmentation in 3d graph-structured point clouds of wild scenes. *arXiv* **2020**, arXiv:2004.12498.
45. Kweon, H.; Kim, J.; Yoon, K.J. Weakly supervised point cloud semantic segmentation via artificial oracle. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 3721–3731.
46. Cheng, M.; Hui, L.; Xie, J.; Yang, J. Sspc-net: Semi-supervised semantic 3d point cloud segmentation network. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 2–9 February 2021; pp. 1140–1147.

47. Lan, Y.; Zhang, Y.; Qu, Y.; Wang, C.; Li, C.; Cai, J.; Xie, Y.; Wu, Z. Weakly supervised 3D segmentation via receptive-driven pseudo label consistency and structural consistency. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; pp. 1222–1230.
48. Zhang, Y.; Li, Z.; Xie, Y.; Qu, Y.; Li, C.; Mei, T. Weakly supervised semantic segmentation for large-scale point cloud. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 19–21 May 2021; pp. 3421–3429.
49. Zhang, Y.; Qu, Y.; Xie, Y.; Li, Z.; Zheng, S.; Li, C. Perturbed self-distillation: Weakly supervised large-scale point cloud semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 15520–15528.
50. Wu, Z.; Wu, Y.; Lin, G.; Cai, J. Reliability-adaptive consistency regularization for weakly-supervised point cloud segmentation. *Int. J. Comput. Vis.* **2024**, *132*, 2276–2289. [CrossRef]
51. Su, Y.; Liu, W.; Yuan, Z.; Cheng, M.; Zhang, Z.; Shen, X.; Wang, C. DLA-Net: Learning dual local attention features for semantic segmentation of large-scale building facade point clouds. *Pattern Recognit.* **2022**, *123*, 108372. [CrossRef]
52. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library In Proceedings of the NeurIPS 2019, Vancouver, BC, USA, 8–14 December 2019; pp. 8024–8035.
53. Hu, Q.; Yang, B.; Fang, G.; Guo, Y.; Leonardis, A.; Trigoni, N.; Markham, A. Sqn: Weakly-supervised semantic segmentation of large-scale 3d point clouds. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 600–619.
54. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
55. Hou, J.; Graham, B.; Nießner, M.; Xie, S. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 9–25 June 2021; pp. 15587–15597.
56. Xie, S.; Gu, J.; Guo, D.; Qi, C.R.; Guibas, L.; Litany, O. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 574–591.
57. Guohao, L.; Müller, M.; Qian, G.; Delgadillo, I.C.; Abualshour, A.; Thabet, A. Deepgcns: Making gcns go as deep as cnns. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021.
58. Qian, G.; Li, Y.; Peng, H.; Mai, J.; Hammoud, H.; Elhoseiny, M.; Ghanem, B. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In *Advances in Neural Information Processing Systems 35*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2022; pp. 23192–23204.
59. Zhao, H.; Jiang, L.; Jia, J.; Torr, P.H.; Koltun, V. Point transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 16259–16268.
60. Wu, X.; Lao, Y.; Jiang, L.; Liu, X.; Zhao, H. Point transformer v2: Grouped vector attention and partition-based pooling. In *Advances in Neural Information Processing Systems 35*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2022; pp. 33330–33342.
61. Wu, Y.; Yan, Z.; Cai, S.; Li, G.; Han, X.; Cui, S. Pointmatch: A consistency training framework for weakly supervised semantic segmentation of 3d point clouds. *Comput. Graph.* **2023**, *116*, 427–436. [CrossRef]
62. Pan, Z.; Zhang, N.; Gao, W.; Liu, S.; Li, G. Point cloud semantic segmentation with sparse and inhomogeneous annotations. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025; pp. 6354–6362.
63. Pan, Z.; Gao, W.; Liu, S.; Li, G. Distribution guidance network for weakly supervised point cloud semantic segmentation. In *Advances in Neural Information Processing Systems 37*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2024; pp. 32400–32420.
64. Yao, B.; Xiao, H.; Zhuang, J.; Peng, C. Weakly supervised learning for point cloud semantic segmentation with dual teacher. *IEEE Robot. Autom. Lett.* **2023**, *8*, 6347–6354. [CrossRef]
65. Tang, L.; Chen, Z.; Zhao, S.; Wang, C.; Tao, D. All points matter: Entropy-regularized distribution alignment for weakly-supervised 3D segmentation. In *Advances in Neural Information Processing Systems 36*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2023; pp. 78657–78673.
66. Lee, M.S.; Yang, S.W.; Han, S.W. Gaia: Graphical information gain based attention network for weakly supervised point cloud semantic segmentation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 2–7 January 2023; pp. 582–591.
67. Han, W.; Wen, C.; Wang, C.; Li, X.; Li, Q. Point2node: Correlation learning of dynamic-node for point cloud feature modeling. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 10925–10932.
68. Thomas, H.; Qi, C.R.; Deschaud, J.E.; Marcotegui, B.; Goulette, F.; Guibas, L.J. Kpconv: Flexible and deformable convolution for point clouds. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6411–6420.

69. Fan, S.; Dong, Q.; Zhu, F.; Lv, Y.; Ye, P.; Wang, F.Y. SCF-Net: Learning spatial contextual features for large-scale point cloud segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14504–14513.
70. Yao, B.; Dong, L.; Qiu, X.; Song, K.; Yan, D.; Peng, C. Uncertainty-guided contrastive learning for weakly supervised point cloud segmentation. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5704913. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.