PDF: A Probability-Driven Framework for Open World 3D Point Cloud Semantic Segmentation

Jinfeng Xu, Siyuan Yang, Xianzhi Li *, Yuan Tang, Yixue Hao, Long Hu, Min Chen
Huazhong University of Science and Technology
{jinfengx, reedyoung, xzli, yuan_tang, yixuehao, hulong, minchen2012}@hust.edu.cn

Abstract

Existing point cloud semantic segmentation networks cannot identify unknown classes and update their knowledge, due to a closed-set and static perspective of the real world, which would induce the intelligent agent to make bad decisions. To address this problem, we propose a ***Probability-Driven Framework (PDF)***[1] *for open world semantic segmentation that includes (i) a lightweight U-decoder branch to identify unknown classes by estimating the uncertainties, (ii) a flexible pseudo-labeling scheme to supply geometry features along with probability distribution features of unknown classes by generating pseudo labels, and (iii) an incremental knowledge distillation strategy to incorporate novel classes into the existing knowledge base gradually. Our framework enables the model to behave like human beings, which could recognize unknown objects and incrementally learn them with the corresponding knowledge. Experimental results on the S3DIS and ScanNetv2 datasets demonstrate that the proposed PDF outperforms other methods by a large margin in both important tasks of open world semantic segmentation.*

1. Introduction

In recent years, deep learning for 3D point clouds has attracted increasing interest due to its great potential in various applications, such as virtual/augmented reality, robotics, autonomous driving, *etc*. Taking advantage of the emergence of high-quality datasets [2, 3, 5, 9] and advances in point cloud networks [21, 33, 34, 41, 44], the semantic segmentation task for point clouds has achieved promising performance in several important metrics. Most existing methods work under the strong assumption that the world is *closed-set* and *static*, which supposes that all the object categories remain consistent in both the training and inference stages. However, the assumption is not valid for many dynamic real-world scenarios in which *unknown* object classes will be encountered outside the learning process. Intelligent agents could make wrong decisions under a closed-set design due to incorrect recognition of unknown classes. Moreover, the agent cannot update its knowledge base under a static perception of the world, while humans can continuously extend their learned knowledge without forgetting. These problems limit the closed-set and static methods to particular scenarios.

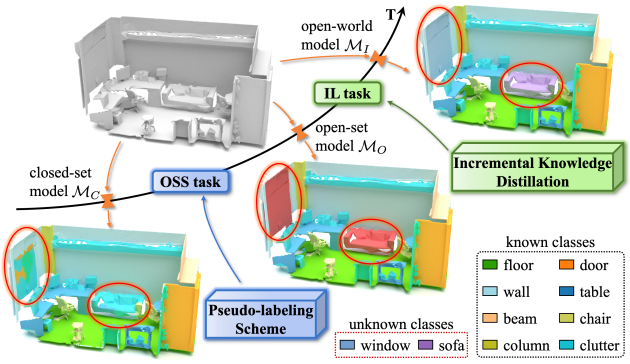

Figure 1. The closed-set model $\mathcal{M}_C$ continuously improves its open-world capabilities by successively finetuning to open-set model $\mathcal{M}_O$ and open-world $\mathcal{M}_I$ with the help of open-set semantic segmentation (OSS) task and incremental learning (IL) task, where the proposed pseudo-labeling scheme and incremental knowledge distillation strategy are employed, respectively.

The open world semantic segmentation (OWSS) addresses the above issues by introducing two tasks: 1) open-set semantic segmentation (OSS) to recognize the known objects and identify unknown objects simultaneously; 2) incremental learning (IL) to update knowledge of the model without retraining from scratch when information about the identified unknown classes would be accessible. Notably, the OSS focuses on identifying unknown classes not present during training, while OWSS addresses both unknown identification and continuous learning when novel classes with ground-truth labels are provided in IL task. Thus, OWSS can be considered as an extension of OSS. Fig. 1 shows the

*Corresponding author
[1]Code available at: https://github.com/JinfengX/PointCloudPDF.

OWSS pipeline for point clouds with an example, where unknown objects are marked with red circles.

Early works on the OSS task mainly focus on the 2D image domain. Uncertainty estimation-based methods [12, 14, 15] and generative network-based methods [20, 30, 45] are two main trends for the 2D OSS task. However, these methods suffer from performance loss when applied in 3D scenes. Although the OWSS problem has gained more attention for relaxing the assumption of realistic application, few studies have investigated it in 3D scenes. Cen *et al*. [7] first proposed a REAL framework to solve both OSS and IL tasks in a general architecture, making redundancy classifiers an important part of the framework. Li *et al*. [27] proposed an Adversarial Prototype Framework (APF) to estimate the distribution of unknown classes with the help of learned prototypes. Although REAL and APF have made satisfactory progress in the OSS task, they have drawbacks when applied to indoor 3D scenes, as discussed in Sec. 2.3.

The present insufficient research situation motivates us to propose a novel OWSS framework, named the Probability-Driven Framework (PDF) for point clouds, which leverages the probability output (the last-layer output of the backbone) to solve both the OSS task and the IL task. As shown in Fig. 1, for the OSS task, we identify the unknown objects by learning the geometry features and the implicit probability distribution features of the unknown classes with our proposed pseudo-labeling scheme. After that, the model incrementally expands its knowledge base through our proposed incremental knowledge distillation strategy for the IL task. This process transforms previously acquired knowledge along with the information from novel classes in a distilled manner. The quantitative and qualitative results show the superiority and effectiveness of our framework compared to the state of the arts. Overall, our contributions can be summarized as follows:

- We propose a novel probability-driven framework (PDF) for open world semantic segmentation of point clouds. It requires less strict conditions in real-world applications.
- We propose a novel pseudo-labeling scheme designed for the OSS task to capture features of unknown classes by leveraging probability outputs. This approach enhances the model's ability to recognize unknown objects.
- We propose a general incremental knowledge distillation strategy for the IL task to incorporate novel semantic classes into learned knowledge incrementally.

2. Related Work

2.1. Closed-set 3D semantic segmentation

Recent 3D semantic segmentation methods can be broadly categorized into projection-based methods, voxel-based methods and point-based methods.

Projection-based methods [1, 38, 46] use 2D CNNs to extract features from projected 3D scenes or shapes and then aggregate these features for label regression. Voxel-based [8, 10, 13, 31, 36, 37] methods employ 3D convolutions to predict semantic occupancy within a voxelized representation of the 3D scenes or shapes. Point-based methods [17, 18, 24–26, 28, 35, 40, 41, 43, 47, 48] handle the segmentation task with efficient point cloud representation. Pioneering point-based methods [33, 34] employed Multi-Layer Perceptrons (MLPs) with symmetric functions to capture geometric features. With the popularity of Transformers, recent methods [21, 22, 32, 39, 44, 49] have used similar mechanisms to improve segmentation accuracy.

In particular, the aforementioned methods focused primarily on the closed-set and static settings, However, the real world presents a dynamic environment in which novel objects of unknown classes may be encountered. Closed-set approaches have high confidence in recognizing known classes, but struggle to identify features of unknown classes. Additionally, these approaches cannot update their models for novel classes. Thus, these issues motivate us to design PDF to train an open-world model with the capability of identifying unknown objects and continuous learning.

2.2. Open-set 2D semantic segmentation

Early research of 2D open-set semantic segmentation (OSS) established robust baselines to estimate model uncertainties using maximum softmax probabilities (MSP) [14] or maximum logit (MaxLogit) [15]. MC-Dropout [12] and Ensembles [23] improved performance by approximating Bayesian inference, adopting a probabilistic perspective. Wang *et al*. [42] and Zhou *et al*. [50] introduced redundancy classifiers to widen the gap between known and unknown classes. Instead of quantifying uncertainties of unknown classes, the generative methods [20, 30, 45] locate unknown objects by comparing the original inputs against the reconstructed outputs of the generative models. Hwang *et al*. [19] proposed an exemplar-based approach for identifying novel classes by clustering. Cen *et al*. [6] proposed to learn class prototypes using contrast clustering and then calculated the similarities of the features within the metric space.

Recent years have witnessed the rapid development of 2D OSS, and these methods have shown robustness, making them applicable to 3D OSS. However, these methods could not handle the 3D OSS task well enough due to neglect of the special contextual information and an unbalanced semantic distribution in 3D scenes.

2.3. Open world 3D semantic segmentation

The open-world problem was first formulated in [4], which combines open-set recognition with incremental learning technology to extend the existing deep learning network to real-world settings. As far as we know, rarely works have investigated the 3D open world semantic seg-

mentation (OWSS). Cen *et al*. [7] first proposed a novel framework called REAL to address the OWSS task for Lidar point clouds. REAL adds several redundancy classifiers (RCs) to the backbone to predict the probability of unknown classes. The RCs are trained with a calibration loss to uniformly force the probability distribution to the unknown classes. Meanwhile, REAL uses the instance of known classes to synthesize unknown objects by randomly resizing, which supplies geometric features for RCs. Then the model in the OSS task is saved to supply previously learned knowledge for the IL task. Very recently, Li *et al*. [27] proposed an adversarial prototype framework (APF) that designs a feature adversarial module and a prototypical constraint module to aggregate the features of known and unknown classes. The APF achieved better performance on the 3D OSS task, while no further research has been conducted on the IL task.

Both REAL and APF try to implicitly rearrange the distribution of features to enlarge the gap between the known classes and the unknown classes. However, the geometry structures of the unknown objects, as well as the associated probability distribution of the unknown classes, are ignored.

3. Open World Semantic Segmentation

In this section, we formalize the definition and give the working pipeline of open world semantic segmentation (OWSS) in 3D point clouds. At any time t, we assume that the set of known object classes $\mathcal{K}^t = \{1, 2, \cdots, C\} \subset \mathbb{N}^+$ is labeled in the training datasets. In addition, there is a set of unknown classes $\mathcal{U}^t = \{C+1, \cdots\}$ that may be encountered in the inference stage. Each sample in the dataset is paired with a point cloud $\mathcal{P}_i = \{\mathbf{p}_1, \cdots, \mathbf{p}_N\}$ and its associated label $\mathbf{Y}_i = \{y_1, \cdots, y_N\}$, where $y_i \in \mathcal{K}^t$ is class label for point $\mathbf{p}_i$ and N is the number of points. Here, each point $\mathbf{p}_i$ is composed of a point coordinate (x_i, y_i, z_i) and c channels features $(f_{i1}, \cdots, f_{ic})$.

As discussed in Sec. 1, a closed-set model $\mathcal{M}_C$ trained with known classes $\mathcal{K}^t$ fails to recognize unknown object classes $\mathcal{U}^t$ under the OWSS condition. However, the closed-set model $\mathcal{M}_C$ can be adapted to open-world applications by introducing the open-set semantic segmentation (OSS) task and incremental learning (IL) task. During the OSS task, the model $\mathcal{M}_C$ will be finetuned to open-set model $\mathcal{M}_O$, which predicts the label of the points of the known classes $\mathcal{K}^t$ and identifies the points belonging to any of the unknown classes $\mathcal{U}^t$. For the IL task, the points of identified unknown objects are annotated with novel class labels $\mathcal{K}_n = \{C+1, \cdots, C+n\} \subset \mathcal{U}$, where n is the number of novel classes. Then the model $\mathcal{M}_O$ is incrementally finetuned to the model $\mathcal{M}_I$ with $\mathcal{K}_n$ so that the knowledge base is enlarged to $\mathcal{K}^{t+1} = \mathcal{K}^t + \mathcal{K}_n$. Considering privacy protection and computation limitation in the open-world setting, the model $\mathcal{M}_I$ can only access $\mathcal{K}_n$ during its training process. Note that the model $\mathcal{M}_I$ maintains the open-set semantic segmentation ability without forgetting all the previously learned knowledge. This pipeline of OSS and IL task cycles in life-long time to continuously update the open-world semantic segmentation model.

4. Probability-Driven Framework

Figure 2 shows the high-level overview of our proposed probability-driven framework (PDF) for open world semantic segmentation (OWSS) of point clouds. Our framework, which is composed of two stages, addresses the open set semantic segmentation (OSS) task in the first stage (highlighted in blue color) and the incremental learning (IL) task in the second stage (highlighted in green color). The encoder-decoder structure is adopted as our backbone in both the OSS and IL tasks.

For the OSS task, we first feed the raw point cloud $\mathcal{P}_{in} \in \mathbb{R}^{N\times(3+c)}$ with N points and c channel features into the backbone for the semantic segmentation output $\mathcal{O}_S \in \mathbb{R}^{N\times C}$, where $C \in \mathcal{K}^t$ is the number of known semantic classes. Note that we treat $\mathcal{O}_S$ as the **probability output** whose vectors could represent the semantic probabilities of the corresponding classes. Meanwhile, we send the backbone features to our designed lightweight U-decoder for the estimated uncertainties $\mathcal{O}_U \in \mathbb{R}^{N\times 1}$ of $\mathcal{O}_S$. Then, we obtain pseudo masks through a pseudo-labeling scheme which will be detailed in Sec. 4.2 The pseudo masks are used to generate the pseudo ground truth by masking the corresponding labels of known classes with $\max \mathcal{K}^t + 1$. Finally, the semantic output $\mathcal{O}_S$ is supervised by closed-set ground truth that only includes known classes. For better performance, the uncertainty output $\mathcal{O}_U$ is concatenated with $\mathcal{O}_S$, then jointly supervised by the pseudo ground truth. The trained model $\mathcal{M}_O$ will be saved for the following IL task.

During the IL task, only the labels of n novel classes $\mathcal{K}_n$ are accessible. The number of backbone's head is correspondingly extended by n to recognize the newly introduced classes. The input point clouds are then fed into the saved open-set model $\mathcal{M}_O$ and open-world model $\mathcal{M}_I$ for output $\mathcal{O}_I$ and O_S^{t+1}, respectively. The output of the open-world model O_S^{t+1} is supervised by distilled ground truth, which is generated by the proposed incremental knowledge distillation strategy (detailed in Sec. 4.3) from $\mathcal{O}_I$ and novel class labels. Note that the open-set task is still in an active state to maintain the open-set capability when training the open-world model.

At the inference stage, the model prediction $\hat{\mathbf{Y}}$ is given based on the semantic segmentation output $\mathcal{O}_S$ and the estimated uncertainties $\mathcal{O}_U$:

$$\hat{\mathbf{Y}} = \begin{cases} \operatorname{argmax} \mathcal{O}_S^i, & \mathcal{O}_U^i < \lambda \\ \max \mathcal{K}^t + 1, & \mathcal{O}_U^i > \lambda, \end{cases} \tag{1}$$

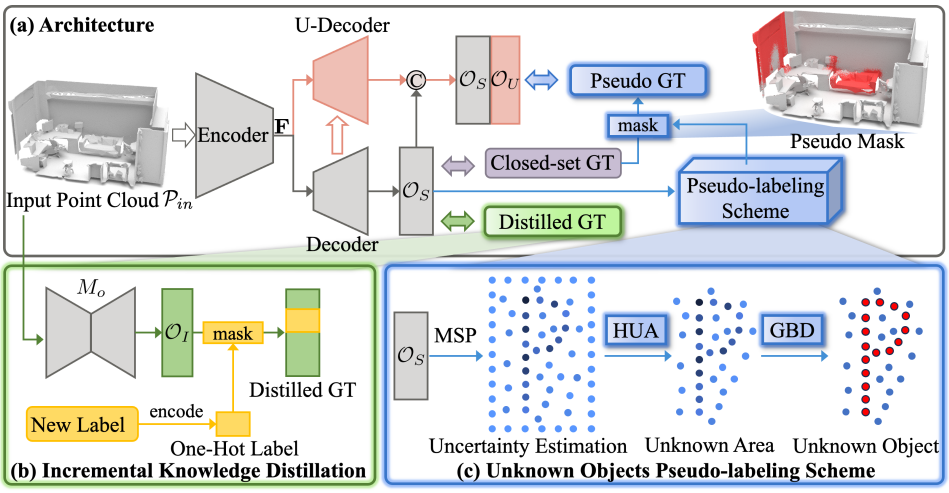

Figure 2. (a) **Architecture of the probability-driven framework.** Given an input point cloud $\mathcal{P}_{in}$, the architecture produces semantic results $\mathcal{O}_S$ and uncertainty results $\mathcal{O}_U$, which are respectively supervised by closed-set GT and pseudo GT in the open-set semantic segmentation task (marked in blue). During incremental learning task (marked in green), $\mathcal{O}_S$ is further supervised by distilled GT. (b) Pipeline of incremental knowledge distillation. (c) Pipeline of pseudo-labeling scheme, which consumes $\mathcal{O}_S$ and outputs pseudo mask for unknown classes.

where λ is the threshold to determine whether the i-th point belongs to unknown classes. Next, we present each component of the proposed PDF in detail.

4.1. Open-set 3D semantic segmentation (OSS)

Given a point cloud $\mathcal{P}_{in}$, the OSS task aims to train an open-set model $\mathcal{M}_O$ to predict per-point semantic labels from the probability output $\mathcal{O}_S$, as well as detect unknown objects by estimating the uncertainties $\mathcal{O}_U$ of the segmentation results $\mathcal{O}_S$. Figure 2 (a) illustrates the pipeline of the OSS task, $\mathcal{P}_{in}$ is first fed into an encoder to extract sparse features $\mathbf{F}$ with downsampling and embedding layers. Then the features $\mathbf{F}$ are upsampled with a decoder to regress the probability output $\mathcal{O}_S$. Closed-set labels $\mathbf{Y}_C$ are used to calculate closed-set semantic segmentation loss $\mathcal{L}_C$:

$$\mathcal{L}_C = CE(\mathcal{O}_S, \mathbf{Y}_C), \tag{2}$$

where $CE(\cdot)$ is the cross entropy loss function.

With $\mathcal{O}_S$ in hand, some methods [6, 14, 15, 27] calculate uncertainty scores for unknown classes with hand-crafted discrimination functions. In contrast, we estimate the uncertainties of $\mathcal{O}_S$ using a lightweight U-decoder, which has a structure similar to the backbone decoder. The U-decoder consumes the encoder features $\mathbf{F}$ as input and fuses the hidden layer output of the backbone decoder layers by skip connections, which can fully leverage the extracted geometric and scene-wise features of the backbone. The output of the U-decoder $\mathcal{O}_U$ describes how confident the model is with its segmentation results $\mathcal{O}_S$. Please refer to our supplementary material for detailed network architecture. To ensure the promising performance of the U-decoder branch, we generate pseudo labels with our proposed pseudo-labeling scheme (Sec. 4.2) to indicate the features of unknown classes. However, these pseudo labels are not used directly for supervision due to the potential risk that leads to an unbalanced distribution on $\mathcal{O}_U$. We thus combined the pseudo labels with closed-set labels by mask operation to obtain the pseudo ground truth $\mathbf{Y}_P$. Then we employ the cross entropy loss on the concatenated U-decoder branch output $\mathcal{O}_U$ and the semantic output $\mathcal{O}_S$:

$$\mathcal{L}_P = CE(\mathcal{O}_S \oplus \mathcal{O}_U, \mathbf{Y}_P), \tag{3}$$

where $\oplus$ is the concatenation operation. The overall open-set semantic segmentation loss $\mathcal{L}_O$ consists of both $\mathcal{L}_C$ and $\mathcal{L}_P$, which are balanced with parameter α:

$$\mathcal{L}_P = \mathcal{L}_C + \alpha\mathcal{L}_P. \tag{4}$$

4.2. Pseudo-labeling scheme

The goal of the pseudo-labeling scheme is to generate pseudo labels of unknown classes by leveraging probability output $\mathcal{O}_S$ to supervise the uncertainty output $\mathcal{O}_U$ of U-decoder. As shown in Fig. 2 (c), we first obtain the uncertainties of points to measure the uncertainties of objects. Hendrycks *et al.* [14] found that known examples tend to have a higher maximum softmax probability (MSP) than unknown examples. Thus, we use the MSP to calculate the uncertainty score $\mathcal{S}$ of the i-th point $\mathbf{p}_i$:

$$\mathcal{S}(\mathbf{p}_i) = \max_k(\exp \mathcal{O}_S^{ik} / \sum\nolimits_j \exp \mathcal{O}_S^{ij}), \tag{5}$$

where $\mathcal{O}_S^{ij}$ is the probability of $\mathbf{p}_i$ belonging to the j-th semantic class and $\max_k(\cdot)$ calculates the max value of the dimension where the index k is located. In particular, maximum logit (MaxLogit) [15] can also be used to calculate uncertainty scores. Then, the HUA algorithm locates the unknown areas based on the per-point uncertainty scores. The unknown areas usually cover both unknown objects and their surrounding points that belong to the background or other objects. Thus, we further separate unknown objects from the other points by designing the GBD algorithm.

Heuristic unknown-aware algorithm. The HUA algorithm takes $\mathcal{O}_S$ as input and outputs pseudo labels that indicate the unknown area. The key idea of HUA is that the uncertainty scores of known objects are probably higher than the uncertainty scores of unknown objects. Therefore, unknown objects tend to be located in the area with lower uncertainty scores. In contrast to cropping the unknown areas with a fixed parameter, we start the unknown area search process with a set of seeds and then heuristically find the unknown area until the stop condition.

Specifically, assume that we already have the uncertainty scores $\mathcal{S}(\mathcal{O}_S) \in \mathbb{R}^{N\times 1}$ of all N points from the probability output $\mathcal{O}_S$ by Eq. (5). We sort $\mathcal{S}(\mathcal{O}_S)$ in ascending order and select m ($m \ll N$) seed points among the top p percent of the sorted $\mathcal{S}(\mathcal{O}_S)$. The selected seeds $\mathcal{P}^0 = \{\mathbf{p}_1^0, \cdots, \mathbf{p}_m^0\}$ got a promising low uncertainty score, which establishes the start conditions of the HUA algorithm. Then, we iteratively merge more points into $\mathcal{P}^0$ to construct a point set with a low average uncertainty score, ensuring the incorporation of the majority of points associated with unknown classes into this set. Let us start the iteration with $\mathcal{P}^0$. We search for the k nearest neighbors of $\mathcal{P}^0$ and measure the similarity between the seeds $\mathcal{P}^0$ and their neighbors $NN(\mathcal{P}^0) = \{nn(\mathbf{p}_1^0), nn(\mathbf{p}_2^0), \cdots, nn(\mathbf{p}_m^0)\}$ with the distance value and the uncertainty score. Here, $nn(\mathbf{p}_i^0) = \{\mathbf{p}_{i1}^0, \mathbf{p}_{i2}^0, \cdots, \mathbf{p}_{ik}^0\} \in \mathbb{R}^{k\times 3}$ are the k neighbor points of $\mathbf{p}_i^0$. The distance similarity matrix Sim_{dis} of $\mathcal{P}_s$ and $NN(\mathcal{P}^0)$ are formulated as:

$$Sim_D = \left(\frac{\|\mathbf{p}_i - nn(\mathbf{p}_i)\|^2}{\max \|\mathbf{p}_i - nn(\mathbf{p}_i)\|^2}\right) \in \mathbb{R}^{m\times k}, \quad (6)$$

where $\mathbf{p}_i \in \mathcal{P}^0$. To accurately represent the similarity of uncertainty scores, we utilize a negative exponent function to calculate the uncertainty score similarity matrix Sim_U:

$$Sim_U = \exp\left(-|\mathcal{S}(\mathbf{p}_i) - \mathcal{S}(nn(\mathbf{p}_i))|\right) \in \mathbb{R}^{m\times k}. \quad (7)$$

The overall similarity matrix is given as:

$$Sim\big(\mathcal{P}^0, NN(\mathcal{P}^0)\big) = Sim_D + Sim_U. \quad (8)$$

The seed points $\mathcal{P}^0$ preferentially pick neighbors with high similarity from the matrix for updating. Empirically, we enlarge $\mathcal{P}^0$ to $\mathcal{P}^1 \in \mathbb{R}^{m^1\times 3}$ ($m^1 > m$) with the points whose similarity values are in the top 50% of $Sim\big(\mathcal{P}^0, NN(\mathcal{P}^0)\big)$. Recall that the HUA algorithm starts from a small set of seed points $\mathcal{P}^0$ with almost the lowest uncertainty scores. Thus, the average value of the uncertainty score of $\mathcal{P}^0$ will increase due to the heuristic search process. Generally, we expect the average uncertainty score of the s-th iteration output $\mathcal{P}^s$ to be below the average uncertainty score of the input points $\mathcal{P}_{in}$. Based on the Eq. (5), the stop condition of the s-th iteration is given as:

$$\overline{\sum \mathcal{S}(\mathbf{p}_i)} < \overline{\sum \mathcal{S}(\mathbf{p}_j)} - \lambda \cdot \sigma\Big(\sum \mathcal{S}(\mathbf{p}_j)\Big), \quad (9)$$

where $\mathbf{p}_i \in \mathcal{P}^s$, $\mathbf{p}_j \in \mathcal{P}_{in}$, $\sigma(\cdot)$ is the standard deviation function and λ is the hyperparameter to adjust the stop condition. Naturally, when λ increases, the HUA has a more strict stop condition, *i.e.*, the HUA output has lower uncertainty scores, which reduces the range of unknown areas.

3D graph boundary detection algorithm. The unknown areas obtained by HUA include not only unknown objects but also other points of the background or known objects, which leads to pollution on the features of the unknown classes. To address this issue, we design a 3D graph boundary detection (GBD) algorithm to separate the unknown objects from other surrounding points by detecting the boundaries of the unknown objects. Unlike 2D images, where pixels are tightly and neatly arranged, point clouds consist of irregularly positioned points with complex geometry structures, rendering traditional boundary detection algorithms ineffective. However, we could construct a 3D "image" based on the graph structure by embedding points into an undirected graph and perform 3D boundary detection on this "image" by applying the idea of 2D boundary detection algorithm, which finds boundaries using the gradient produced from the difference between adjacent pixels.

Specifically, given the output of HUA algorithm $\mathcal{P}^s$, we first obtain the similarity value between $\mathcal{P}^s$ and their neighbors $NN(\mathcal{P}^s)$ by Eq. (6), Eq. (7) and Eq. (8). Then we embed each point of $\mathcal{P}^s$ in an undirected graph $\mathcal{G}$ with the similarity matrix $Sim\left(\mathcal{P}^s, NN(\mathcal{P}^s)\right)$:

$$\mathcal{G} = \left\{\mathcal{P}^s, Sim\left(\mathcal{P}^s, NN(\mathcal{P}^s)\right)\right\}, \quad (10)$$

where $\mathcal{P}^s$ and $Sim\left(\mathcal{P}^s, NN(\mathcal{P}^s)\right)$ are the nodes and the edge weights of $\mathcal{G}$ respectively. Note that, directly dealing with $\mathcal{G}$ is inefficient, thus we crop the redundancy edges by searching the minimum spanning tree (MST) $\mathcal{T}$ of $\mathcal{G}$. The MST connects all the nodes of $\mathcal{G}$ with minimum sum weights of existing edges, *i.e.*, the nodes tend to connect with their least similar neighbors.

As aforementioned in Sec. 4.2, the unknown classes tend to have lower uncertainty scores than known classes. Furthermore, we find that the uncertainty scores of the unknown classes are distributed in a wider region with lower values compared to the known classes. Thus, nodes of known classes probably get larger edge weights than nodes of unknown classes according to Eq. (6), Eq. (7) and Eq. (8), as demonstrated in Fig. 3. When the edges with high weights are cut off, most of the nodes belonging to known classes are isolated. On the contrary, nodes belonging to unknown classes still preserve most of the edges, which are easy to distinguish from isolated nodes. To achieve this goal, we approximately fit the overall distribution of the edge weights with a Gaussian Mixed Model (GMM):

$$p(x) = \sum_{k=1}^{2} \pi_k \mathcal{N}(x|\mu_k, \sigma_k), \quad (11)$$

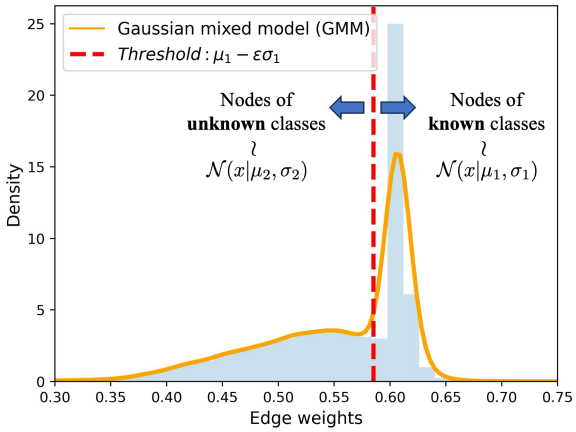

Figure 3. **Edge weights distribution.** The edges' weights between nodes of known classes are distinct from unknown classes. The distribution is approximately fitted with a Gaussian mixed model, and divided by the threshold $\mu_1 - \epsilon\sigma_1$ derived from the 3σ criteria.

where π_k, μ_k and σ_k are the weight, mean and variance of the k-th Gaussian component respectively. Assuming that the distributions of the known classes and unknown classes are $\mathcal{N}(x|\mu_1, \sigma_1)$ and $\mathcal{N}(x|\mu_2, \sigma_2)$ respectively, we have $\mu_1 > \mu_2$ and $\sigma_1 < \sigma_2$. Then, the edges whose weights are larger than $\mu_1 - \epsilon\sigma_1$ are cut off to split $\mathcal{T}$ into a group of subgraphs. Here, $\mu_1 - \epsilon\sigma_1$ are the 3σ outlier detection criteria. Finally, we obtain the unknown objects by merging the subgraphs whose number of nodes is not rejected by the outlier detection algorithm.

4.3. Incremental learning (IL)

IL task aims to finetune the open-set model $\mathcal{M}_O$ to the open-world model $\mathcal{M}_I$ by learning introduced novel class $\mathcal{K}_n$ without losing the open-set semantic segmentation ability. Importantly, the training data only contains unknown classes at this stage to prevent retraining the model from scratch. Finetuning the model with only the labels of $\mathcal{K}_n$ will lead to an erroneous prediction tendency for the novel classes, which is called catastrophic forgetting [11]. To alleviate knowledge forgetting, the open-set model $\mathcal{M}_O$ in open-set semantic segmentation (OSS) task is used as a teacher model to supply the knowledge of previously learned C known classes for the IL task. Inspired by knowledge distillation [16], we transform knowledge from $\mathcal{M}_O$ to $\mathcal{M}_I$ by minimizing the distribution of probability output.

As shown in Fig. 2 (b), the input point cloud $\mathcal{P}_{in}$ is sent into $\mathcal{M}_O$ and $\mathcal{M}_I$ for the probability output $\mathcal{O}_I$ and $\mathcal{O}_S^{t+1}$, respectively. The output $\mathcal{O}_S$ is used to generate pseudo probability labels $\mathcal{Y}_I = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_N\}$ for each input point, where N is the number of points. Each pseudo probability label $\mathbf{y}_i$ is obtained by distilling the corresponding soft target $\mathbf{o}_I^i \in \mathcal{O}_I$:

$$\mathbf{y}_i = \mathcal{D}(\mathbf{o}_I^i, T) = \left[\frac{\exp(\mathbf{o}_I^{ic}/T)}{\sum_c \exp(\mathbf{o}_I^{ic}/T)}\right] \in \mathbb{R}^{1\times C}, \quad (12)$$

where $\mathbf{o}_I^{ic}$ is the probability of $\mathbf{o}_I^i$ belonging to the c-th semantic class and T is the distillation temperature. Then, we transform the labels of novel classes to one-hot format labels $\mathbf{E}_I$ for incorporating with $\mathcal{Y}_I$. By masking $\mathcal{Y}_I$ with $\mathbf{E}_I$, we get the distilled ground truth $\mathcal{Y}_D$:

$$\mathcal{Y}_D = \begin{cases} \mathcal{Y}_I, & \text{if } \mathcal{Y}_{\mathbf{p}_i} \notin \mathcal{K}_n \\ \mathbf{E}_I, & \text{if } \mathcal{Y}_{\mathbf{p}_i} \in \mathcal{K}_n, \end{cases} \quad (13)$$

where $\mathcal{Y}_{\mathbf{p}_i}$ is the label of point $\mathbf{p}_i$. In this way, we obtain the previously learned knowledge and the new knowledge by combining the labels of novel classes with generated pseudo probability labels. The semantic segmentation loss of the IL task $\mathcal{L}_I$ is calculated by KL-divergence, which minimizes the probability distribution of $\mathcal{O}_S^{t+1}$ and $\mathcal{Y}_D$:

$$L_I = \sum_{\mathbf{o}_i^{t+1}, \mathbf{y}_D^i} \mathcal{D}(\mathbf{o}_i^{t+1}) \log\left(\frac{\mathcal{D}(\mathbf{o}_i^{t+1})}{\mathbf{y}_D^i}\right), \quad (14)$$

where $\mathbf{o}_i^{t+1} \in \mathcal{O}_S^{t+1}$ and $\mathbf{y}_D^i \in \mathcal{Y}_D$.

5. Experiments

5.1. Datasets

We conduct experiments on the S3DIS [2] and ScanNetv2 [9] datasets for both the open-set semantic segmentation (OSS) task and the incremental learning (IL) task of the open world semantic segmentation (OWSS) problem. S3DIS comprises 271 point cloud scenes, which are annotated with 13 semantic classes. ScanNetv2 includes 1613 indoor scans annotated with point-wise semantic labels in 20 categories. For the OSS task, we follow the APF [27] to select {*window*, *sofa*} as the unknown classes in S3DIS. In addition, more classes including {*chair*, *door*, *refrigerator*, *toilet*} are labeled as unknown classes in ScanNetv2. All unknown classes are annotated with the label "-1". For the IL task, the labels of the unknown classes are introduced with newly assigned labels to update the open-set model.

5.2. Evaluation metrics

The OSS is composed of closed-set segmentation task and unknown class identification task as discussed in Sec. 4.1. We employ the mean class IoU (mIoU) for the former task. The latter task adopts the area under the ROC curve (AUROC) and area under the precision-recall curve (AUPR) as metrics. To evaluate the performance of IL, we employ mIoU for both previously learned known classes and newly introduced classes, which are denoted as mIoU_{old} and $\text{mIoU}_{\text{novel}}$ respectively.

Table 1. Open-set semantic segmentation results of 3D point clouds on S3DIS and ScanNetv2. We use the results in APF and mark these results with "*". The unavailable results are marked with "-". The best results are in bold in each metric.

Methods	S3DIS						ScanNetv2					
	PointTransformer			StratifiedTransformer			PointTransformer			StratifiedTransformer		
	AUPR	AUROC	mIoU	AUPR	AUROC	mIoU	AUPR	AUROC	mIoU	AUPR	AUROC	mIoU
MSP	15.2*	70.3*	**69.8***	16.7	72.4	**70.5**	35.6	75.0	**64.5**	36.8	77.8	**64.9**
MaxLogit	17.5*	74.3*	**69.8***	30.1	81.0	**70.5**	44.2	78.1	**64.5**	49.5	82.7	**64.9**
MC-Dropout	18.2*	75.9*	**69.8***	16.4	68.6	69.0	17.8	57.0	60.4	40.9	79.2	61.3
DMLNet	20.5*	80.7*	67.2*	17.4	66.3	70.4	36.6	76.8	63.5	30.3	76.1	61.6
REAL	25.4*	87.6*	69.7*	50.9	87.1	70.4	42.4	**87.3**	63.9	58.1	90.3	**64.9**
APF	31.6*	90.0*	69.3*	-	-	-	-	-	-	-	-	-
Ours	**73.1**	**96.2**	68.6	**66.4**	**92.7**	70.3	**60.2**	86.0	64.2	**67.8**	**90.9**	64.4

5.3. Implementation details

We apply our proposed probability-driven framework (PDF) on two backbones: PointTransformer [49] and StratifiedTransformer [21]. We implement the networks in PyTorch and trained on 4 Nvidia RTX 3090 GPUs for 3000 (on S3DIS) and 600 (on ScanNetv2) epochs with batch size of 16. During IL task, we further finetune the network for about 200 epochs. The SGD and Adam optimizer are used for PointTransformer and StratifiedTransformer, respectively. The balance parameter α in Eq. (4) is set to 0.001. In S3DIS dataset, we respectively set m, p and λ as 20, 0.02 and 1.0 for the heuristic unknown-aware (HUA) algorithm in Sec. 4.2. In the ScanNetv2 dataset, n, p and λ in HUA increase to 200, 0.15 and 2.0, respectively, due to a higher percentage of points belonging to unknown classes in scenes against S3DIS.

5.4. Open-set semantic segmentation

We first compare our proposed PDF with all existing 3D OSS methods, including REAL [7] and APF [27]. In addition, we further adapt 2D OSS methods, such as MSP [14], MaxLogit [15], MC-Dropout [12] and DMLNet [6], to address 3D OSS tasks. For APF, the code is not released for now, we thus directly use the evaluation values published in their original paper for comparison on S3DIS.

Table 1 shows the comparison results. MSP and MaxLogit achieve the best performance in the mIoU metric because there is no modification to the networks. Our method obtained significantly better performance on the unknown class identification task across all experiments, with a slight sacrifice to the closed-set semantic segmentation results. Note that we trained PointTransformer in the S3DIS dataset and got 69.2 on the mIoU metric, which is lower than the value supplied in APF. DMLNet and MC-Dropout have slightly lower mIoU than other methods, which may be caused by the negative influence of modifications to the architecture and training process. The results presented above demonstrate that the proposed PDF is capable of enhancing the overall balance between closed-set segmentation and unknown class identification.

Table 2. Incremental learning results on S3DIS. There are 11 previously known semantic classes and 2 novel classes (window, sofa). The mIoU, $\text{mIoU}_{\text{novel}}$ and mIoU_{old} are mean class IoU of all classes, previous known classes and novel classes, respectively.

Methods	mIoU	$\text{mIoU}_{\text{novel}}$	mIoU_{old}
Closed-set	58.6	0	69.2
Upper bound	70.1	72	69.8
Finetune	0.5	3.0	0.0
Feature extraction	57.3	9.5	66.1
LwF	62.3	23.9	69.2
REAL	68.9	61.3	70.3
Ours	**69.4**	**64.3**	**70.3**

Figure 4 further shows the qualitative comparisons on two datasets, where the top scenes are from S3DIS and the bottom scenes are from ScanNetv2. Clearly, the unknown objects identified by our method (g) are closest to the ground truths (a) against others (b-f). More visual results are shown in our supplemental file.

5.5. Incremental learning

Since most OSS methods are not involved in the IL task, we thus carry out experiments on S3DIS with the methods designed for the IL task, *e.g.* LwF [29] and Feature Extraction. We further compare the proposed PDF with REAL, which is designed for OWSS, in terms of the IL task. The result of directly finetuning the open set model $\mathcal{M}_O$ to the open-world model $\mathcal{M}_I$ with only novel classes is reported to illustrate catastrophic forgetting, as discussed in Sec. 4.3. The closed-set result is derived from the model trained with the known classes. Furthermore, the upper bound retrains the model using both known and unknown classes, essentially treating it as an oracle during the forward process. All approaches are implemented on the PointTransformer.

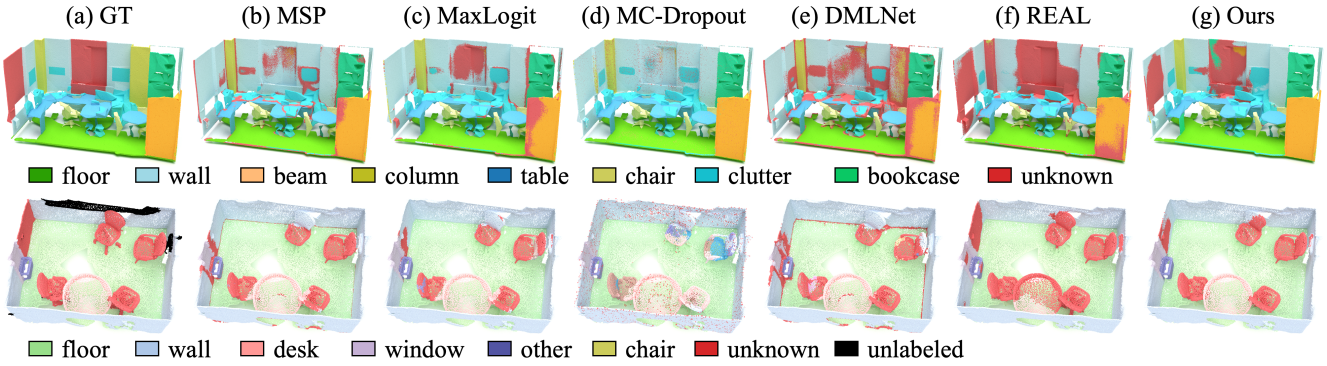

Figure 4. Comparing the open-set semantic segmentation (OSS) results of our method (g) and other OSS methods (b-f).

The results of the IL task are demonstrated in Tab. 2. Directly finetuning the open-set model using novel classes leads to serious bias, which incorrectly assigns the labels of the novel class to all points, *i.e.*, causes catastrophic forgetting. Our method achieves the best performance in both known classes and newly introduced classes compared to the others, showing the effectiveness of the incremental knowledge distillation strategy. Compared with the upper bound, the knowledge of learned known classes is transferred to the open-world model without changing the networks, but experiences a performance degradation when it comes to learning new categories. Further detailed results on the IL task are shown in our supplemental file.

5.6. Ablation study

To evaluate the individual contribution of each component in our framework, we perform ablation studies on the ScanNetv2 dataset, utilizing the StratifiedTransformer as the backbone. We re-trained the network separately for following each experiment and reported the key results in Tab. 3. Please refer to the supplementary material for more details.

- Model A replaces the U-decoder with an MLP layer.
- Model B removes the semantic output $\mathcal{O}_S$ in Eq. (3).
- Model C removes the pseudo-labeling and loss $\mathcal{L}_P$.
- Model D removes the GBD algorithm in Sec. 4.2.

By comparing model C & D *vs*. our full pipeline, we can see that both HUA and GDB in the pseudo-labeling scheme contribute to a better performance for open-set semantic segmentation. In particular, the GDB algorithm can enhance both closed-set and open-set abilities, which can be seen by comparing D with ours. By comparing model A with our full pipeline, we can see that the designed U-decoder can improve task performance. By comparing models A & B, we find that the open-set task has a serious setback when uncertainty output $\mathcal{O}_U$ is not supervised together with probability output $\mathcal{O}_S$, which may be caused by the imbalanced distribution in the supervisory signal.

Table 3. Comparing the contribution of major components in our framework on ScanNetv2 using StratifiedTransformer.

Experiment	AUPR	AUROC	mIoU
A	67.6	90.6	64.1
B	31.0	76.0	64.2
C	20.0	62.0	**64.9**
D	64.3	85.8	63.0
Ours	**67.8**	**90.9**	64.4

6. Conclusion

In this work, we present a novel Probability-Driven Framework (PDF) for open world semantic segmentation of 3D point clouds. Our framework addresses both open-set semantic segmentation and incremental learning tasks, by designing a lightweight U-decoder for estimating uncertainties of unknown classes, a pseudo-labeling scheme to generate ground truth for unknown classes, and an incremental knowledge distillation strategy for integrating novel classes into the existing knowledge base. Both quantitative and qualitative results demonstrate that our framework significantly outperforms the state-of-the-art. Nevertheless, our method's performance is constrained to capturing the geometry and probability distribution of objects in outdoor scenes, where sparse-occupied and severely incomplete objects are frequently misclassified with high confidence. In the future, we shall explore the possibility of utilizing more network features to improve task performance. We hope that our work can attract more attention to this practically significant open problem.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (NSFC) No.62202182, No.62176101, No.62276109, No.62322205.

References

[1] Angelika Ando, Spyros Gidaris, Andrei Bursuc, Gilles Puy, Alexandre Boulch, and Renaud Marlet. Rangevit: Towards vision transformers for 3d semantic segmentation in autonomous driving. In *CVPR*, pages 5240–5250, 2023. 2

[2] Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *CVPR*, 2016. 1, 6

[3] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *ICCV*, 2019. 1

[4] Abhijit Bendale and Terrance Boult. Towards open world recognition. In *CVPR*, 2015. 2

[5] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 1

[6] Jun Cen, Peng Yun, Junhao Cai, Michael Yu Wang, and Ming Liu. Deep metric learning for open world semantic segmentation. In *ICCV*, pages 15333–15342, 2021. 2, 4, 7

[7] Jun Cen, Peng Yun, Shiwei Zhang, and et al. Open-world semantic segmentation for lidar point clouds. In *ECCV*, pages 318–334, 2022. 2, 3, 7

[8] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *CVPR*, 2019. 2

[9] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 1, 6

[10] Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. In *CVPR*, 2018. 2

[11] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999. 6

[12] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1050–1059, New York, New York, USA, 2016. PMLR. 2, 7

[13] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018. 2

[14] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*, 2017. 2, 4, 7

[15] Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joseph Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *Proceedings of the 39th International Conference on Machine Learning*, pages 8759–8773. PMLR, 2022. 2, 4, 7

[16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 6

[17] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *CVPR*, 2020. 2

[18] Binh-Son Hua, Minh-Khoi Tran, and Sai-Kit Yeung. Pointwise convolutional neural networks. In *CVPR*, 2018. 2

[19] Jaedong Hwang, Seoung Wug Oh, Joon-Young Lee, and Bohyung Han. Exemplar-based open-set panoptic segmentation network. In *CVPR*, pages 1175–1184, 2021. 2

[20] Shu Kong and Deva Ramanan. Opengan: Open-set recognition via open data generation. In *ICCV*, pages 813–822, 2021. 2

[21] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *CVPR*, pages 8500–8509, 2022. 1, 2, 7

[22] Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical transformer for lidar-based 3d recognition. In *CVPR*, pages 17545–17555, 2023. 2

[23] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NeurIPS*. Curran Associates, Inc., 2017. 2

[24] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *CVPR*, 2018. 2

[25] Huan Lei, Naveed Akhtar, and Ajmal Mian. Octree guided cnn with spherical kernels for 3d point clouds. In *CVPR*, 2019.

[26] Guohao Li, Matthias Muller, Ali Thabet, and Bernard Ghanem. Deepgcns: Can gcns go as deep as cnns? In *ICCV*, 2019. 2

[27] Jianan Li and Qiulei Dong. Open-set semantic segmentation for point clouds via adversarial prototype framework. In *CVPR*, pages 9425–9434, 2023. 2, 3, 4, 6, 7

[28] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. In *NeurIPS*. Curran Associates, Inc., 2018. 2

[29] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE TPAMI*, 40(12):2935–2947, 2017. 7

[30] Krzysztof Lis, Krishna Nakka, Pascal Fua, and Mathieu Salzmann. Detecting the unexpected via image resynthesis. In *ICCV*, 2019. 2

[31] Hsien-Yu Meng, Lin Gao, Yu-Kun Lai, and Dinesh Manocha. Vv-net: Voxel vae net with group convolutions for point cloud segmentation. In *ICCV*, 2019. 2

[32] Chunghyun Park, Yoonwoo Jeong, Minsu Cho, and Jaesik Park. Fast point transformer. In *CVPR*, pages 16949–16958, 2022. 2

[33] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 1, 2

[34] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017. 1, 2

[35] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In *NeurIPS*, pages 23192–23204. Curran Associates, Inc., 2022. 2

[36] Dario Rethage, Johanna Wald, Jurgen Sturm, Nassir Navab, and Federico Tombari. Fully-convolutional point networks for large-scale point clouds. In *ECCV*, 2018. 2

[37] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3d representations at high resolutions. In *CVPR*, 2017. 2

[38] Maxim Tatarchenko, Jaesik Park, Vladlen Koltun, and Qian-Yi Zhou. Tangent convolutions for dense prediction in 3d. In *CVPR*, 2018. 2

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*. Curran Associates, Inc., 2017. 2

[40] Shenlong Wang, Simon Suo, Wei-Chiu Ma, Andrei Pokrovsky, and Raquel Urtasun. Deep parametric continuous convolutional neural networks. In *CVPR*, 2018. 2

[41] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics*, 38(5):1–12, 2019. 1, 2

[42] Yezhen Wang, Bo Li, Tong Che, Kaiyang Zhou, Ziwei Liu, and Dongsheng Li. Energy-based open-world uncertainty modeling for confidence calibration. In *ICCV*, pages 9302–9311, 2021. 2

[43] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *CVPR*, 2019. 2

[44] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. In *NeurIPS*, pages 33330–33342. Curran Associates, Inc., 2022. 1, 2

[45] Yingda Xia, Yi Zhang, Fengze Liu, Wei Shen, and Alan L. Yuille. Synthesize then compare: Detecting failures and anomalies for semantic segmentation. In *ECCV*, pages 145–161, Cham, 2020. Springer International Publishing. 2

[46] Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation. In *ECCV*, pages 1–19. Springer, 2020. 2

[47] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *CVPR*, 2020. 2

[48] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *CVPR*, 2019. 2

[49] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H.S. Torr, and Vladlen Koltun. Point transformer. In *ICCV*, pages 16259–16268, 2021. 2, 7

[50] Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Learning placeholders for open-set recognition. In *CVPR*, pages 4401–4410, 2021. 2