machines

MDPI

Article

Semantic Segmentation for Point Clouds via Semantic-Based Local Aggregation and Multi-Scale Global Pyramid

Shipeng Cao [1,2,3,4], Huaici Zhao [1,2,3,4,*] and Pengfei Liu [1,2,3,4]

[1] Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
[2] Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China
[3] Key Laboratory of Opto-Electronic Information Process, Shenyang 110016, China
[4] The Key Laboratory of Image Understanding and Computer Vision, Shenyang 110016, China
* Correspondence: hczhao@sia.cn

Abstract: Recently, point-based networks have begun to prevail because they retain more original geometric information from point clouds than other deep learning-based methods. However, we observe that: (1) the set abstraction design for local aggregation in point-based networks neglects that the points in a local region may belong to different semantic categories, and (2) most works focus on single-scale local features while ignoring the importance of multi-scale global features. To tackle the above issues, we propose two novel strategies named semantic-based local aggregation (SLA) and multi-scale global pyramid (MGP). The key idea of SLA is to augment local features based on the semantic similarity of neighboring points in the local region. Additionally, we propose a hierarchical global aggregation (HGA) module to extend local feature aggregation to global feature aggregation. Based on HGA, we introduce MGP to obtain discriminative multi-scale global features from multi-resolution point cloud scenes. Extensive experiments on two prevailing benchmarks, S3DIS and Semantic3D, demonstrate the effectiveness of our method.

Keywords: point cloud; deep learning; semantic segmentation; feature aggregation

Citation: Cao, S.; Zhao, H.; Liu, P. Semantic Segmentation for Point Clouds via Semantic-Based Local Aggregation and Multi-Scale Global Pyramid. *Machines* **2023**, *11*, 11. https://doi.org/10.3390/machines11010011

Academic Editors: Praneel Chand and Antonios Gasteratos

Received: 29 November 2022
Revised: 17 December 2022
Accepted: 19 December 2022
Published: 22 December 2022

1. Introduction

With the development of augmented reality, autonomous driving and other technologies that interact with the real world, 3D scene understanding [1–4] is becoming increasingly important. As one of the methods for fine-grained 3D scene understanding, semantic segmentation of 3D point clouds has attracted more and more attention recently. Furthermore, deep learning methods have achieved impressive success in image processing. Therefore, many recent works have attempted to process point clouds with deep learning networks. Compared with regular images, point clouds contain more geometric details. However, point clouds are typically irregularly sampled, unstructured and unordered. In order to process point clouds with networks, it is necessary to solve a series of problems caused by these properties. Recent neural network-based point cloud processing methods mainly include projection-based, voxel-based and point-based methods. We focus on point-based methods because they preserve more geometrical details than others.

PointNet [2] is the pioneering work of point-based networks. Furthermore, the set abstract (SA) layer proposed in PointNet++ [3] is usually exploited as a basic structure to extract local features of point clouds. In the SA layer, a sampling algorithm is introduced to select a subset of input as center points in this layer and simultaneously as the input for the next layer. The hierarchical learning is constructed by gradually reducing the resolution of point clouds through the sampling algorithm. Then, the SA layer uses a grouping algorithm to find neighboring points for each sampled center point to construct a local region. Finally, a permutation-invariant operation is applied to aggregate local representation from neighboring point-wise features. However, most works [4–6] based on the SA layer have two drawbacks: (1) the SA layer neglects to consider whether the

points in the local region belong to the same category. Furthermore, the local representation aggregated from point-wise features of different categories cannot represent any class. Assuming that the local region is inside the boundary of an object, the local representation aggregated naturally represents the class to which the object belongs. Nevertheless, in the junction region with multi-objects of different classes, the local representation may not represent any class; (2) most works focus on how to extract local features efficiently while ignoring global features. Moreover, due to the use of a fixed searching field to construct local regions, most works with the SA layer have only a unicity receptive field, which limits the expressive ability of features. However, global features contain more implicit information about large-scale objects than local features, and multi-scale features enable the network to identify objects with similar spatial structures but different scales. Some works [7–9] use multi-scale receptive fields simultaneously to extract multi-scale features, but this leads to increased computational consumption and redundant information.

To overcome the shortcomings mentioned above, we propose two simple but effective strategies, semantic-based local aggregation (SLA) and multi-scale global pyramid (MGP), that can plug in the encoders with the SA layer to improve the semantic segmentation ability of the network. Specifically, for SLA, we embed a semantic augmentation (SeA) module in each encoding layer to augment local contexts by emphasizing points with semantic similarity. Semantically augmented local features are then aggregated from local contexts. Furthermore, we propose MGP to introduce multi-scale global features. MGP contains multiple hierarchical global aggregation (HGA) modules, and each HGA module outputs a global feature at a specific scale. Finally, these global features at different scales are fused to generate a new multi-scale global feature. Through MGP, we obtain discriminative multi-scale global features without adding multi-scale receptive fields additionally. Therefore, we avoid redundant information, and the increase in computational consumption is limited. The main idea of HGA is to divide the global scene into multiple 'super local regions', then extend the local feature aggregation to hierarchical global aggregation to obtain global features.

In summary, our contributions are as follows:

- We discuss the defect in the local feature aggregation of the SA layer and propose a novel strategy to handle it by exploiting the semantic similarity of neighboring points in the local region.
- We redesign the local feature aggregation to extend it to aggregate global features. Without adding additionally receptive fields, we obtain multi-scale global features with a limited increase in computational consumption and avoid redundant information.
- We conduct extensive experiments on prevailing benchmarks, and the results demonstrate the effectiveness of our method.

2. Related Work

Firstly, we briefly overview three existing deep learning-based methods for point clouds: projection-based, voxel-based and point-based methods. Then, we discuss the expression capacity of various point-based features.

2.1. Projection-Based Method and Voxel-Based Methods

The inherent properties of point clouds, including that they are unordered and unstructured, make it impossible to extract features through convolutional neural networks directly. Some works [10–12] project 3D point clouds onto multi-view 2D images to leverage successful 2D CNNs. However, the projection leads to the loss of geometric information, which is the most significant advantage of point clouds over other data formats. Some works [13–15] voxelized point clouds into 3D grids to utilize 3D CNNs. The primary limitation of voxel-based networks is their heavy memory consumption, and voxelization leads to loss of fine-grained geometry information. Moreover, the size of the 3D grids will affect the performance of the network. There is no efficient way to automatically set the size, relying on experience.

2.2. Point-Based Methods

The point-based networks [16–18] process the raw point clouds directly. The input to each encoding layer of the pioneer work PointNet [2] are geometric embeddings. Then, a symmetry function is applied to solve the unordered property of point clouds. PointNet++ [3] introduces sampling and grouping layers to construct hierarchical point set feature learning. Inspired by them, many recent works explore how to extract local features effectively. Although these networks have shown promising performance, only a few works consider distinguishing the categories of points within a local region. SASA [19] and IA-SSD [20] introduce classification operations to retain more important foreground points in detection during the downsampling stage. CGA-Net [21] inserts a new branch to identify points before prediction. The new branch re-searches neighboring points to construct new local regions. Then, it uses a classifier to partition the neighboring points into two soft collections to obtain augmented local features. However, the new branch introduces additional computational consumption. We directly utilize the local regions constructed in encoding layers to avoid excessive computational consumption.

2.3. Point-Based Feature Expression

Local Feature Expression. PointNet++ aggregates local features through a symmetry function, which makes the network permutation-invariant. Symmetry functions are simple but effective, achieving strong performance in different backbones. Then, several local aggregation strategies [5,22] have been introduced to capture geometric structures of point cloud local regions based on symmetric functions. On the other hand, due to the inherent permutation-invariance, the attention mechanism is well suited for point clouds. Attention-based local aggregation methods [4,6] consider neighboring points with different importance and avoid information loss.

Global Feature Expression. Global features model the entire scene in the environment understanding task and are helpful in recognizing medium- and large-scale objects. Recently, many efforts [23–25] have attempted to design global descriptors for point clouds. However, they did not consider that it is feasible to extend local aggregation to global aggregation without completely redesigning a new global aggregation scheme. Therefore, we propose the HGA module to implement this idea.

Multi-scale Feature Expression. Many works construct local regions based on their pre-defined fixed receptive fields, resulting in only single-scale features being extracted. However, single-scale features confuse the network when recognizing objects with similar spatial structures but different scales. PointNet++ [3] uses multi-scale grouping (MSG) to extract multi-scale local features. The MSG method applies grouping layers with different scales to capture multi-scale patterns simultaneously. However, extracting point-wise features with multi-scale receptive fields leads to a huge increase in computational consumption. Furthermore, redundant information is introduced in the overlapping regions of different-scale receptive fields. We propose the MGP strategy to obtain multi-scale global features while avoiding the above problems. Specifically, we use multiple HGA modules on the multi-resolution point cloud scenes generated from the encoding layers to obtain multiple global features with different-level receptive fields. Then, multi-level receptive field information is fused to obtain the final multi-scale global features.

3. Method

The overall structure of our method is shown in Figure 1. We utilized [4] with five encoding layers as the baseline. Firstly, we embedded one SLA into each encoding layer to augment the local features through semantic similarity. Then, the augmented local features were fed into the next encoding layer and the HGA module. In MGP, HGAs aggregate multiple global features at different scales, and these global features are fused to form a new multi-scale global feature.

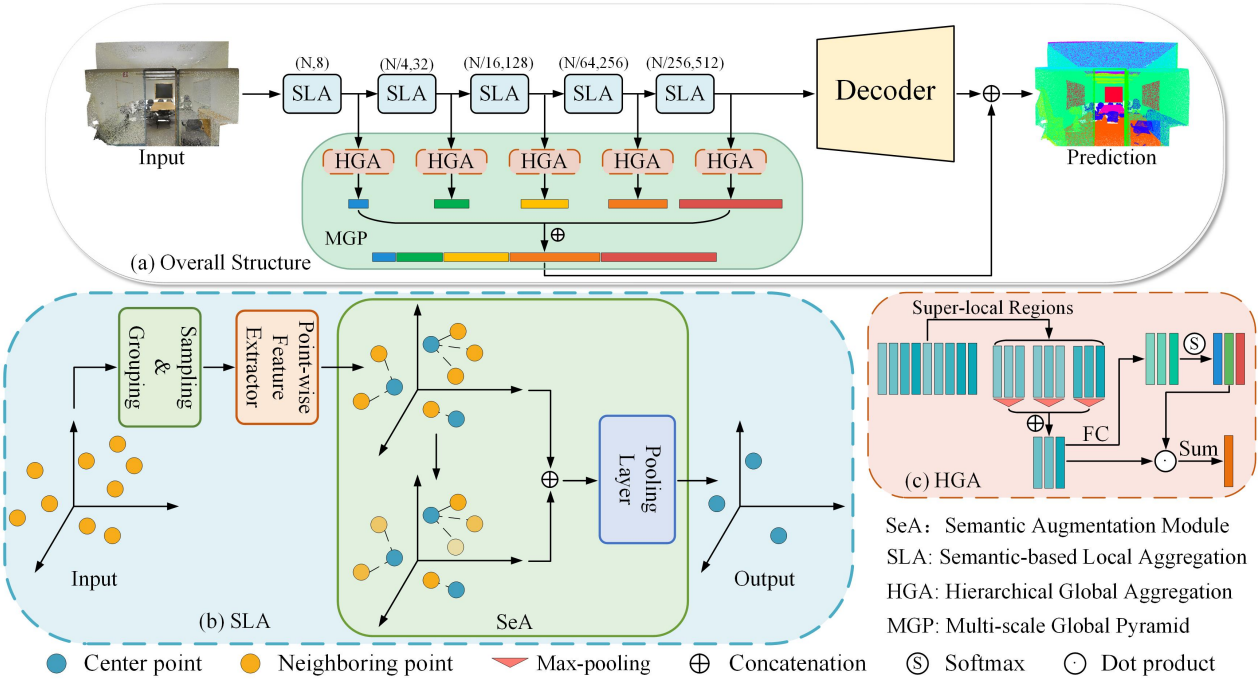


Figure 1. Flowchart of our proposed method. (**a**) is the overall framework. (**b**) is the SLA strategy; we embed a SeA module between the point-wise extractor and the pooling layer to augment local context. (Different transparencies indicate different levels of semantic similarity). (**c**) is the HGA module, and MGP in (**a**) contains five HGAs.

3.1. Selected-Based Local Aggregation

Given a point cloud P containing N points, the SA layer samples and groups input points first. Then, it aggregates the local representation through a pooling layer. However, existing point-based methods hardly consider that not all points in a local region belong to the same class. The points belonging to different classes may be located in a local region where multiple objects adhere to each other. A straightforward idea to tackle the problem is that we need to distinguish which neighboring points belong to the same class as the center point in feature aggregation. Motivated by this idea, we propose a simple but effective strategy named semantic-based local aggregation (SLA). It is mainly composed of two parts: a point-wise feature extractor and a semantic augmentation (SeA) module. First, point-wise features of input point clouds are extracted through the point-wise feature extractor. Then, a SeA module is applied to obtain local representations, which are augmented by the semantic similarity between neighboring points and the center point.

3.1.1. Point-Wise Feature Extractor

Most existing point-based networks follow the structure of the SA layer for hierarchical feature learning. In the baseline [4], the SA layers sample the input point cloud into subsets $\{P_1, P_2, P_3, P_4, P_5\}$ and then use a fixed-number KNN to find K neighbors for each center point. Given the input points with coordinates X and per-point raw features F (e.g., color, normal), we feed them to the encoding layers with the point-wise feature extractor of the baseline to extract point-wise features before the local aggregation. We use both the 3D positions and RGB values of each point for training.

3.1.2. Semantic Augmentation Module

As shown in Figure 1, we employ a semantic augmentation (SeA) module to obtain semantically augmented local representations after extracting the point-wise features of neighboring points. The SeA module consists of two blocks: a semantic score block and a semantic augmentation block.

Semantic Score Block. We design a score function in this block to calculate semantic scores. The score function consist of a 1 × 1 convolution layer followed by *softmax*. Specifically, denoting $\{f_1^{l_c} \cdots f_k^{l_c} \cdots f_K^{l_c}\}$ as the c-dimension neighboring point-wise features in the local region from the l-th encoding layer, the semantic score $p_j^l \in [0, 1]$ for each neighboring point is calculated as:

$$p_j^l = \sigma(Conv(f_j^{l_c})), j = 1, 2, \cdots, K \tag{1}$$

where *Conv* denotes the convolution layer, σ is a *softmax* function, and K is the number of neighboring points. The score function maps the input point-wise features to semantic scores in a learning-based way.

Semantic Augmentation Block. Different semantic scores reflect the different levels of semantic similarity possessed by neighboring points. In other words, the mutual semantic features of neighboring points are suitable to represent the local region. Given neighboring point-wise features $\{f_1^{l_c} \cdots f_k^{l_c} \cdots f_K^{l_c}\}$ from the point-wise feature extractor and their corresponding semantic scores $\{p_1^l \cdots p_k^l \cdots p_K^l\}$, firstly, we use the semantic scores as a soft mask to augment the neighboring features:

$$f_{j_{Score}}^l = p_j^l \odot f_j^l \tag{2}$$

where $f_{j_{Score}}^l$ is the neighboring features augmented by semantic scores, and $\odot$ means element-wise multiplication, which makes the point-wise and channel-wise of neighboring features weighted by the semantic score simultaneously. However, the fitting ability of the score function is limited, which makes it impossible for the classification of neighboring points to be completely accurate. Therefore, score mapping errors are inevitable. For example, some features with semantic similarity may be assigned small scores. As a result, these features are suppressed after semantic augmentation, and the feature values become smaller. They may even be values of zero in extreme cases, and information loss occurs at this moment. Aiming at this problem, we concatenate the original neighboring features to supplement the features suppressed by the error semantic scores. By concatenating the two kinds of features together, it not only alleviates the information loss but also ensures the network knows the implicit semantic level. Theoretically, it follows:

$$\hat{f}_j^l = f_{j_{Score}}^l \oplus f_j^l \tag{3}$$

where $\oplus$ is the concatenation operation. Finally, the pooling layer of the baseline is applied to obtain the augmented local context representation set $\hat{f}^l$ of the l-th encoding layer. The progress mentioned above is formulated as:

$$\hat{f}^l = pool(\hat{f}_j^l) \tag{4}$$

where *pool* denotes the pooling layer.

Losses. The mapping of semantic scores is supervised by the cross-entropy loss, which is formulated as follows:

$$L_{cls} = -\sum_{c=1}^{C} (s_i \log(\widehat{s}_j) + (1 - s_j) \log(1 - \widehat{s}_j) \tag{5}$$

where $\widehat{s}_j$ is the predicted logits, and s_j is the ground-truth segmentation label of the j-th point in the l-th encoding layer. (A classification prediction of 1 means that the label of the neighbor is the same as the center point, and 0 means it is not the same.)

Our method can be trained end-to-end. The total loss comprises the baseline loss L_{ori} and the L_{cls} classification loss as follows:

$$L_{total} = L_{ori} + L_{cls} \tag{6}$$

3.2. Multi-Scale Global Pyramid

The operations mentioned above focused on local features and ignored global features. Furthermore, limited by the fixed receptive field, the baseline only extracts single-scale features. To tackle the above shortcomings, we propose multi-scale global pyramid (MGP) to introduce multi-scale global features. The architecture of MGP is shown in Figure 1.

3.2.1. Implicit Multi-Scale Information

The downsampling operations in the encoding layers naturally generate multi-resolution point cloud scenes. To introduce multi-scale features, a straightforward idea is to consider how to employ existing local representations $\{\hat{f}^1 \cdots \hat{f}^l \cdots \hat{f}^M\}$ from different encoding layers. These local representations are aggregated from the scenes at different resolutions with $\{N_1 \cdots N_l \cdots N_M\}$ points. Although the baseline uses a fixed number KNN to search neighboring points to construct local regions, the receptive field of K points in each encoding layer naturally changes according to the changes in the scene resolution. The reason for this is that the spaces covered by the K nearest neighbors are diverse when KNN is applied for scenes with different resolutions. Therefore, the receptive field in each encoding layer is variational. Technically, multi-scale information is implicit in the local features from different resolution scenes. However, the local representations contain different numbers of points; they cannot directly be utilized together to obtain multi-scale information. Therefore, we aggregate global features, which are equal point-wise, from the local representations to exploit the multi-scale information explicitly.

3.2.2. Hierarchical Global Aggregation

The most significant distinction between the local region and the global scene is their coverage, i.e., the number of points they contain. An experiment about permutation invariance in PointNet [2] shows that max-pooling performs better than attention-summing when dealing with all global points at one time. The conclusion is the same as our experiments on how to aggregate global features effectively. However, recent works [4,6] have utilized attention-based pooling methods to aggregate local features and achieved better performance than max-pooling. Based on the observation that "max-pooling is suitable for processing massive global points, attention is suitable for processing local points", and considering global scenes as "super-local regions", we redesign the local feature aggregation method and extend it to global feature aggregation. Furthermore, we propose a novel hierarchical global aggregation method. Given the local representation $\hat{f}^l = \{\hat{f}^l_1 \cdots \hat{f}^l_n \cdots \hat{f}^l_{N_l}\}$ with N_l points, and that $\hat{f}^l$ is from the l-th encoding layer, firstly, we divide N_l points into T super-local regions. Each super-local region contains N^T points. Then, the max-pooling operations are used on them to obtain super-local features. The formula for calculating the t-th super-local feature is:

$$\tilde{f}^l_t = \max\{\hat{f}^l_1 \cdots \hat{f}^l_{n^T} \cdots \hat{f}^l_{N^T}\} \tag{7}$$

where $\max\{\}$ means max-pooling. We repeat the above operation to obtain the super-local feature set $\{\tilde{f}^l_1 \cdots \tilde{f}^l_t \cdots \tilde{f}^l_T\}$.

After obtaining the super-local feature set, the attention mechanism is used to aggregate a global feature from it. We design a function $\alpha()$ that consists of a multilayer perceptron (MLP) followed by *softmax* to regress the attention parameters $\{\gamma_1 \cdots \gamma_t \cdots \gamma_T\}$ corresponding to $\{\tilde{f}^l_1 \cdots \tilde{f}^l_t \cdots \tilde{f}^l_T\}$ based on learning. It is formally defined as follows:

$$\gamma^l_t = \alpha(\tilde{f}^l_t, W) \tag{8}$$

We exploit the attention parameters in the adaptive fusion of the super-local feature set and finally obtain a comprehensive global feature map $\tilde{r}_l$. This is formulated as:

$$\tilde{r}_l = \sum_{t=1}^{T} (\gamma_t^l \cdot \tilde{f}_t^l) \tag{9}$$

By repeating the above operations on the point cloud scenes with M different resolutions, we obtain M global features with different receptive fields, denoted as $\{\tilde{r}_1 \cdots \tilde{r}_l \cdots \tilde{r}_M\}$.

3.2.3. Multi-Scale Global Feature Representation

Aiming to form a new global feature including multi-scale information with minimal computational consumption, we concatenate $\{\tilde{r}_1 \cdots \tilde{r}_l \cdots \tilde{r}_M\}$ together simply. This is formulated as:

$$\tilde{r} = Concat(\tilde{r}_1 \cdots \tilde{r}_l \cdots \tilde{r}_M) \tag{10}$$

where $Concat()$ is the concatenation operation. Finally, we fuse the comprehensive global feature map $\tilde{r}$ with each local representation $\widehat{f}$ to make predictions. Theoretically, it follows:

$$F_{pre} = \widehat{f} \oplus \tilde{r} \tag{11}$$

4. Experiments

We conducted experiments and evaluated our method on two prevailing benchmarks, S3DIS [26] and Semantic3D [27]. We used the complete network of [4] as the baseline, including all its encoding and decoding layers. For the encoding layers, we only inserted our two strategies into the basic local feature aggregation modules and kept the other structures unchanged. For the decoding layers, we did not modify any settings. In addition, the random sampling algorithm was still used in the downsampling process. The official implementation of it was conducted by TensorFlow; we remained on the same platform. All experiments were implemented in Tensorflow 1.14.0 on a single NVIDIA RTX2080Ti GPU with CUDA 10.0 and cuDNN v7. The experimental results demonstrate the baseline is improved by our method.

4.1. Datasets and Training Settings

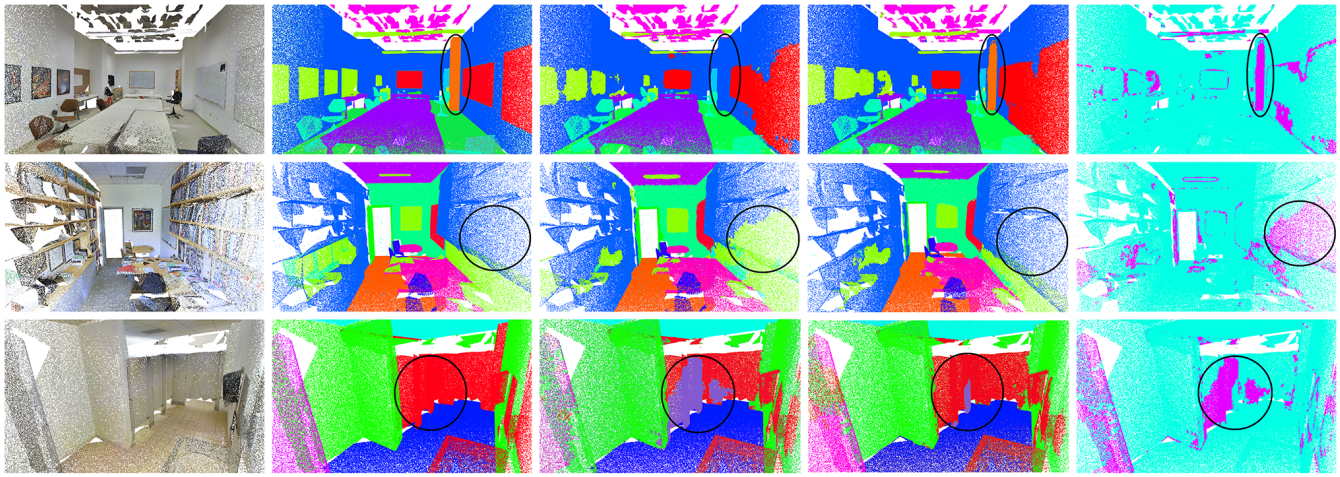
S3DIS. The Stanford 3D Large-Scale Indoor Spaces (S3DIS) dataset contains 6 subareas with 271 rooms, covering about 6020 square meters. Each area has different functional attributes, architectural structures and interior decoration details, mainly including office areas, walkways and restrooms. The point clouds that make up each room range from 0.5 million to 2.5 million, and each point labelled as one of 13 semantic categories. All points are provided with 3D coordinates and color information.

Semantic3D. The Semantic3D dataset is a large-scale 3D point cloud dataset of outdoor scenes consisting of more than 4 billion points covering a variety of natural landscapes and artificial buildings in rural and urban settings: churches, streets, villages, castles, etc. The coverage area reaches 160 × 240 × 30 cubic meters. All points are labeled into 8 classes, and each point contains 3D coordinates, RGB information, and intensity value information. We used the 3D position and the colors of points for training and testing.

Training Settings. We trained for 100 epochs with the Adam optimizer to minimize the overall loss. The initial learning rate was 10^2, and it was set to decay with a rate of 5% after every epoch. The batch size was set as 4 or 2 when training with S3DIS or Semantic3D. The number of K in the KNN algorithm was set to be 16 for both datasets. The amount of input points was 40×2^{10} and 64×2^{10} for S3DIS and Semantic3D, respectively.

4.2. Evaluation on S3DIS

We evaluated our method on Area5 of S3DIS and compared it with recent works on the semantic segmentation task. The mean intersection over union (mIoU) is utilized as a standard metric. Table 1 presents the quantitative results of different approaches on S3DIS, and Figure 2 shows the visual comparison of the segmentation results of our method and the baseline in three typical indoor scenes.



(a) Input (b) Ground truth (c) Baseline (d) Ours (e) Different areas

Figure 2. Visualization results of *S3DIS*. From the left to the right: RGB-colored input point clouds, ground truths, baseline, ours, and the different areas between the baseline and ours.

As seen from Table 1, our method achieves the best performance on three classes, including walls, boards, and columns. The segmentation accuracy of the door class is also greatly improved. The common property of these objects is that they have similar simple spatial structures but different scales. This structural property confuses the network with only single-scale features when recognizing objects. Nevertheless, our method distinguishes the objects well, demonstrating the effectiveness of the introduced multi-scale global features. The visualization of results shown in Figure 2 also demonstrates the superiority of our method. A major difficulty in indoor scene segmentation is that some objects, such as columns close to white walls, are difficult to distinguish. However, as seen in Figure 2, our method improves the areas where the baseline segment is wrong.

4.3. Evaluation on Semantic3D

Table 2 presents a comparison of quantitative results of various methods on the dataset, leveraging the mIoU and the overall accuracy (OA) as the standard metric. Compared to the baseline, our method achieves improvements in high vegetation and low vegetation, two classes with similar structures but different scales. There is also a certain improvement in the classes of buildings and hardscapes. Figure 3 shows the visualization of the segmentation results.

Table 1. Quantitative results of different approaches on Area5 of *S3DIS* dataset.

Method	mIoU (%)	Ceil.	Floor	Wall	Beam	Col.	Wind.	Door	Chair	Table	Book.	Sofa	Board	Clut.
PointNet [2]	41.1	88.0	97.3	69.8	0.1	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
PointCNN [18]	57.3	92.3	98.2	79.4	0.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	**56.7**
PCCN [28]	58.3	92.3	96.2	75.9	**0.3**	6.0	**69.5**	63.5	66.9	65.6	47.3	68.9	59.1	46.2
PointWeb [22]	60.3	92.0	98.5	79.4	0.0	21.1	59.7	34.8	76.3	88.3	46.9	69.3	64.9	52.5
HPEIN [29]	61.9	91.5	98.2	81.4	0.0	23.3	65.3	40.0	75.5	87.7	58.5	67.8	65.6	49.4
PointASNL [30]	62.6	**94.3**	98.4	79.1	0.0	26.7	55.2	66.2	**83.3**	86.8	47.6	68.3	56.4	52.1
Seg-GCN [31]	63.6	93.7	**98.6**	80.6	0.0	28.5	42.6	**74.5**	80.9	**88.7**	**69.0**	71.3	44.4	54.3
SCF-Net [6]	63.4	90.9	97.0	80.8	0.0	18.6	60.3	44.8	79.2	87.8	73.8	**71.4**	68.6	50.5
Baseline [4]	62.4	91.1	95.6	80.2	0.0	25.0	61.9	47.3	75.9	83.4	60.7	70.7	65.4	53.8
Ours	**65.2**	91.8	97.2	**81.5**	0.0	**39.4**	63.4	50.4	78.3	86.8	65.1	70.6	**70.3**	52.9

Table 2. Quantitative results of different approaches on the reduced 8 split of *Semantic3D* dataset.

Method	mIoU (%)	OA (%)	Manmade	Natural	High Veg.	Low Veg.	Buildings	Hardscape	Scanning Art	Cars
SnapNet_ [32]	59.1	88.6	82.0	77.3	79.7	22.9	91.1	18.4	37.3	64.4
SEGCloud [33]	61.3	88.1	83.9	66.0	86.0	40.5	91.1	30.9	27.5	64.3
SPG [34]	73.2	94.0	**97.4**	92.6	87.9	44.0	93.2	31.0	63.5	76.2
RF_MSSF [35]	62.7	90.3	87.6	80.3	81.8	36.4	92.2	24.1	42.6	56.6
KPConv [36]	74.6	92.9	90.9	82.2	84.2	47.9	94.9	40.0	**77.3**	79.7
GACNet [37]	70.8	91.9	86.4	77.7	**88.5**	60.6	94.2	37.3	43.5	77.8
BAAF-Net [5]	75.3	94.3	96.3	**93.7**	87.7	48.1	94.6	43.8	58.2	79.5
SCF-Net [6]	77.6	94.7	97.1	91.8	86.3	51.2	95.3	50.5	67.9	**80.7**
Baseline [4]	77.4	94.8	95.6	91.4	86.6	51.5	95.7	51.5	69.8	76.8
Ours	**78.0**	**94.9**	96.0	89.9	88.3	**53.7**	**97.1**	**52.6**	70.6	74.0

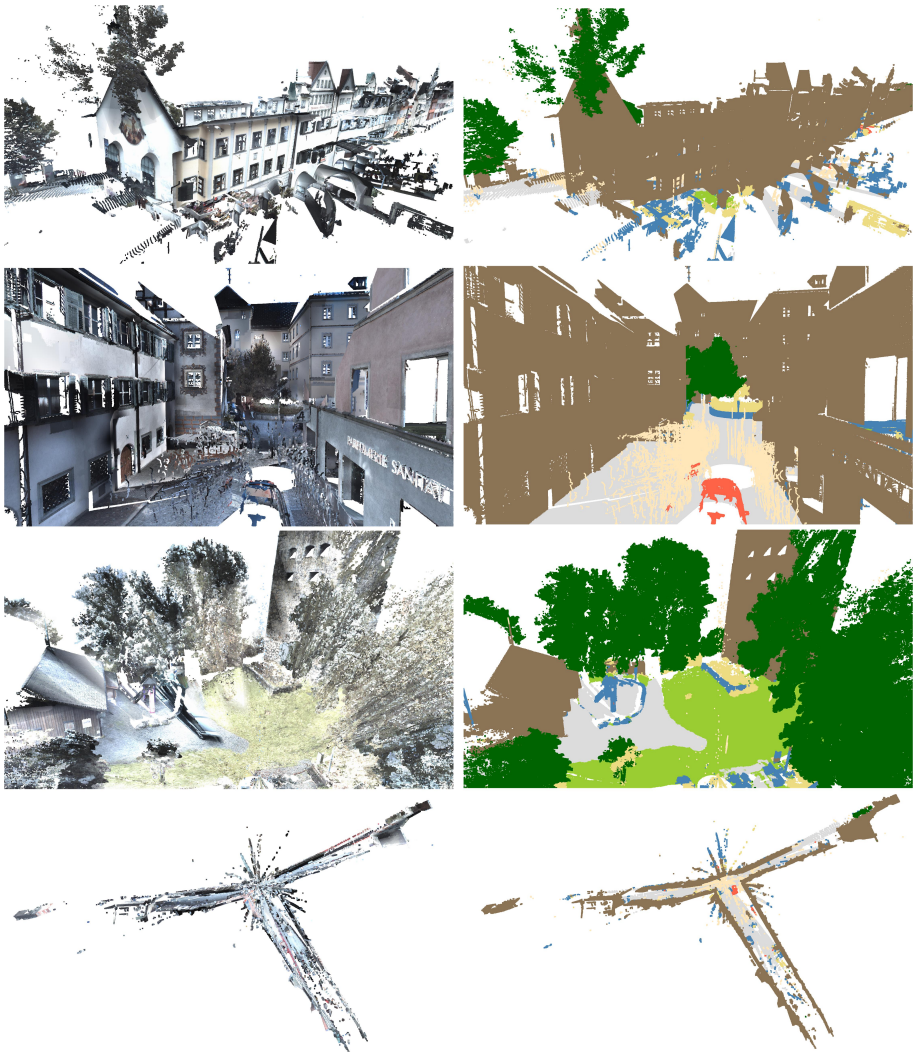


Figure 3. Qualitative results of our method on the reduced 8 split of *Semantic3D*. Left: RGB-colored point clouds; right: predicted semantic results. Note that the ground truth of the test set is not publicly available.

4.4. Ablation Study

We conducted ablation studies to prove the effectiveness of our method. All experiments were conducted on Area5 of S3DIS under the same baseline configurations, except for the controlled conditions.

4.4.1. Semantic-Based Local Aggregation

The goal of SLA is to augment the local representation through the semantic similarity between neighboring points and the center point. A vanilla idea is to divide the neighbors in a local region into two sets according to whether they belong to the same class as the center point. Then, these two sets are utilized to augment local context. Based on this idea, we employed *softmax* to model a binary classifier. Specifically, we used a linear layer to map the input features to a 2-dimensional feature space, and the two dimensions represented the probability of whether the neighbors and the center point belonged to the same class. The experiment results are shown in Table 3. A1 was the baseline model, and its local representation was aggregated from the baseline local context. BLC, FSS and FIS indicate different local contexts. From Table 3, we observe that: (1) the performance of A2 is lower than A1 because the information of points with low-level semantic similarity is discarded, and the local information is incomplete; (2) comparing model A2 with A3, the neighboring points with irrelevant semantics improve their performance when the

augmented local context is not concatenated with the baseline local context because FIS is beneficial for modeling the relationship between the neighbors belonging to specific semantic categories around the center point; (3) from A3&A4, concatenating with a baseline local context is better than two types of augmented local context concatenation. This may be because SeA augments the local context according to the calculated semantic scores. However, error calculations inevitably occur due to insufficient modeling ability, especially in shallow networks. Using two augmented local contexts simultaneously results in the superposition of errors. The baseline local context only provides the original information without any predictive operation that may be incorrect. Therefore, A4 is better than A3; (4) the performance of model A5 is lower than model A4, which shows that more information does not necessarily lead to better results but valuable information.

Table 3. Ablation studies for the semantic augmentation module on Area5 of the *S3DIS* dataset. BLC indicates baseline local context; FSS indicates augmented local context by neighboring features with semantic similarity; FIS indicates augmented local context by neighboring features with irrelevant semantics.

Model	BLC	FSS	FIS	mIoU (%)
A1	✓			62.38
A2		✓		62.09
A3		✓	✓	63.24
A4	✓	✓		64.32
A5	✓	✓	✓	62.76

4.4.2. The Number and Location of SeA Modules

There are two aggregation operations in the local feature aggregation module of [4], so we validated how many times and where the SeA module would result in better performance when used. The results are shown in Table 4. Intuitively, two stacked SeA modules in each encoding layer with two semantic classifications should distinguish the points with semantic similarity better. Nevertheless, more SeA modules lead to lower mIoU than baseline (B1&B2). The reason may be that the network does not have enough model capacity to classify the neighboring points correctly in the first aggregation operation, leading to poor performance. To verify this conjecture, we designed a comparison experiment (B3&B4). The result is that B4 performs poorer than B3, proving our conjecture.

Table 4. Ablation studies for the number and location of SeA module on Area5 of the *S3DIS* dataset. B1 is the baseline; B2 means plugging two SeA modules in the first and second feature aggregation operations; B3 means plugging one SeA in the first feature aggregation operation; B4 means plugging one SeA in the second feature aggregation operation.

Model	B1	B2	B3	B4
mIoU (%)	62.38	62.12	62.26	64.32

4.4.3. Multi-Scale Global Pyramid

In Table 5, we study the structure of MGP by investigating the components individually. C1 is the baseline model with SLA. From models C2, C3 and C4, we observe that the max-pooling operation achieves the best performance when processing global points. Comparing C4 with C5, the concatenation operation is better than the summation operation. Comparing models C6, C7 and C8, the suitable number of super-local region on S3DIS is shown to be 4.

Table 5. Ablation studies on the MGP strategy tested on Area5 of the *S3DIS* dataset. *max* indicates max-pooling; *mean* indicates mean-pooling; *attention-sum* indicates attention-based weighted sum; *T* indicates the number of super-local regions; and *concat* and *sum*, respectively, represent concatenating and summing multiple global features to form a new multi-scale global feature.

Model	Aggregation Manner	*T*	Fusion Method	mIoU (%)
C1	none	none	none	64.32
C2	mean	none	concat	64.43
C3	attention-sum	none	concat	64.29
C4	max	none	concat	64.81
C5	max	none	sum	63.76
C6	max	2	concat	62.75
C7	max	4	concat	65.21
C8	max	8	concat	61.89

4.4.4. Network Complexity

Table 6 shows the changes in network complexity caused by the two strategies of SLA and MGP. It can be seen that as these two strategies are embedded in the baseline, the parameters and FLOPs of the network gradually increase. However, we obtain more effective and accurate semantic segmentation results.

Table 6. Complexity analysis of different semantic segmentation networks on Area5 of *S3DIS*.

Model	Parameters (Millions)	FLOPs (M)	mIoU (%)
Baseline	4.99	31.35	62.38
Baseline + SLA	6.39	35.90	64.32
Baseline + SLA + MGP	7.85	46.12	65.21

5. Conclusions

In this paper, we propose two novel strategies for point-based 3D semantic segmentation. Firstly, we exploit the semantic similarity between neighboring points and the center point in the local region to augment the local context. Then, the semantic augmented local representation is aggregated. Additionally, we divide global scenes into super-local regions, extending local aggregation to aggregate global features in a hierarchical aggregation manner. Finally, we leverage the implicit multi-scale information in local features to obtain multi-scale global features. Experiments on prevailing benchmarks illustrate that our method effectively improves the baseline performance, and we analyze our method by conducting related ablation studies. In the future, how to model the relationship between local regions will be our next goal.

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

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. U2013210).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Park, C.; Jeong, Y.; Cho, M.; Park, J. Fast Point Transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 16949–16958.
2. 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 (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
3. 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 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
4. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. Randla-net: Efficient semantic segmentation of large-scale point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11108–11117.
5. Qiu, S.; Anwar, S.; Barnes, N. Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 1757–1767.
6. 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.
7. Xu, S.; Wan, R.; Ye, M.; Zou, X.; Cao, T. Sparse cross-scale attention network for efficient lidar panoptic segmentation. *arXiv* **2022**, arXiv:2201.05972.
8. Mao, Y.; Sun, X.; Diao, W.; Chen, K.; Guo, Z.; Lu, X.; Fu, K. Semantic Segmentation for Point Cloud Scenes via Dilated Graph Feature Aggregation and Pyramid Decoders. *arXiv* **2022**, arXiv:2204.04944.
9. Guan, T.; Wang, J.; Lan, S.; Chandra, R.; Wu, Z.; Davis, L.; Manocha, D. M3detr: Multi-representation, multi-scale, mutual-relation 3d object detection with transformers. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3-08 January 2022; pp. 772–782.
10. Su, H.; Maji, S.; Kalogerakis, E.; Learned-Miller, E. Multi-view convolutional neural networks for 3d shape recognition. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 945–953.
11. Wu, B.; Wan, A.; Yue, X.; Keutzer, K. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 1887–1893.
12. Milioto, A.; Vizzo, I.; Behley, J.; Stachniss, C. Rangenet++: Fast and accurate lidar semantic segmentation. In Proceedings of the 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS), Macau, China, 3–8 November 2019; pp. 4213–4220.
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. Meng, H.Y.; Gao, L.; Lai, Y.K.; Manocha, D. Vv-net: Voxel vae net with group convolutions for point cloud segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8500–8508.
15. Zhu, X.; Zhou, H.; Wang, T.; Hong, F.; Ma, Y.; Li, W.; Li, H.; Lin, D. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9939–9948.
16. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph. (Tog)* **2019**, *38*, 1–12. [CrossRef]
17. Wu, W.; Qi, Z.; Fuxin, L. Pointconv: Deep convolutional networks on 3d point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9621–9630.
18. Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; Chen, B. Pointcnn: Convolution on x-transformed points. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, ON, Canada, 3–8 December 2018.
19. Chen, C.; Chen, Z.; Zhang, J.; Tao, D. Sasa: Semantics-augmented set abstraction for point-based 3d object detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 1.
20. Zhang, Y.; Hu, Q.; Xu, G.; Ma, Y.; Wan, J.; Guo, Y. Not all points are equal: Learning highly efficient point-based detectors for 3d lidar point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 18953–18962.
21. Lu, T.; Wang, L.; Wu, G. Cga-net: Category guided aggregation for point cloud semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 11693–11702.
22. Zhao, H.; Jiang, L.; Fu, C.W.; Jia, J. Pointweb: Enhancing local neighborhood features for point cloud processing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5565–5573.
23. Cop, K.P.; Borges, P.V.; Dubé, R. Delight: An efficient descriptor for global localisation using lidar intensities. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 3653–3660.
24. Du, J.; Wang, R.; Cremers, D. Dh3d: Deep hierarchical 3d descriptors for robust large-scale 6dof relocalization. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 744–762.

25. Hui, L.; Yang, H.; Cheng, M.; Xie, J.; Yang, J. Pyramid point cloud transformer for large-scale place recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 6098–6107.
26. 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.
27. Hackel, T.; Savinov, N.; Ladicky, L.; Wegner, J.; Schindler, K.; Pollefeys, M. Semantic3D.net: A New Large-Scale Point Cloud Classification Benchmark. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *41*, 91–98. [CrossRef]
28. Wang, S.; Suo, S.; Ma, W.C.; Pokrovsky, A.; Urtasun, R. Deep parametric continuous convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2589–2597.
29. Jiang, L.; Zhao, H.; Liu, S.; Shen, X.; Fu, C.W.; Jia, J. Hierarchical point-edge interaction network for point cloud semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 10433–10441.
30. Yan, X.; Zheng, C.; Li, Z.; Wang, S.; Cui, S. Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5589–5598.
31. Lei, H.; Akhtar, N.; Mian, A. Seggcn: Efficient 3d point cloud segmentation with fuzzy spherical kernel. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11611–11620.
32. Boulch, A.; Le Saux, B.; Audebert, N. Unstructured point cloud semantic labeling using deep segmentation networks. In Proceedings of the 3Dor '17: Workshop on 3D Object Retrieval, Lyon, France, 23–24 April 2017.
33. Tchapmi, L.; Choy, C.; Armeni, I.; Gwak, J.; Savarese, S. Segcloud: Semantic segmentation of 3d point clouds. In Proceedings of the 2017 International Conference on 3D Vision (3DV), Qingdao, China, 10–12 October 2017; pp. 537–547.
34. Landrieu, L.; Simonovsky, M. Large-scale point cloud semantic segmentation with superpoint graphs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4558–4567.
35. Thomas, H.; Goulette, F.; Deschaud, J.E.; Marcotegui, B.; LeGall, Y. Semantic classification of 3D point clouds with multiscale spherical neighborhoods. In Proceedings of the 2018 International Conference on 3D Vision (3DV), Verona, Italy, 5–8 September 2018; pp. 390–398.
36. 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.
37. Wang, L.; Huang, Y.; Hou, Y.; Zhang, S.; Shan, J. Graph attention convolution for point cloud semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 10296–10305.