electronics

MDPI

Article

LLGF-Net: Learning Local and Global Feature Fusion for 3D Point Cloud Semantic Segmentation

Jiazhe Zhang †, Xingwei Li *,†, Xianfa Zhao and Zheng Zhang

College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; zhangjiazhe@nudt.edu.cn (J.Z.); zhaoxianfa16@nudt.edu.cn (X.Z.); zhangzheng0624@nudt.edu.cn (Z.Z.)

* Correspondence: lxw_mmoml@nudt.edu.cn

† These authors contributed equally to this work.

Abstract: Three-dimensional (3D) point cloud semantic segmentation is fundamental in complex scene perception. Currently, although various efficient 3D semantic segmentation networks have been proposed, the overall effect has a certain gap to 2D image segmentation. Recently, some transformer-based methods have opened a new stage in computer vision, which also has accelerated the effective development of methods in 3D point cloud segmentation. In this paper, we propose a novel semantic segmentation network named LLGF-Net that can aggregate features from both local and global levels of point clouds, effectively improving the ability to extract feature information from point clouds. Specifically, we adopt the multi-head attention mechanism in the original Transformer model to obtain the local features of point clouds and then use the position-distance information of point clouds in 3D space to obtain the global features. Finally, the local features and global features are fused and embedded into the encoder–decoder network to generate our method. Our extensive experimental results on the 3D point cloud dataset demonstrate the effectiveness and superiority of our method.

Keywords: local and global; 3D point cloud; semantic segmentation; multi-head attention

Citation: Zhang, J.; Li, X.; Zhao, X.; Zhang, Z. LLGF-Net: Learning Local and Global Feature Fusion for 3D Point Cloud Semantic Segmentation. *Electronics* **2022**, *11*, 2191. https://doi.org/10.3390/electronics11142191

Academic Editor: Byung Cheol Song

Received: 8 June 2022
Accepted: 11 July 2022
Published: 13 July 2022

1. Introduction

Point clouds are a common form of data representation in 3D vision. 3D point cloud processing is an important technology in the development of 3D vision and plays a significant role in the development of autonomous driving, intelligent robots, and other fields. Because point clouds have high-dimensional information in 3D space compared to 2D images, 3D point cloud processing can obtain richer spatial structure information and complex geometric shape information, which has incomparable advantages for accurate navigation and perception of environment information [1]. In this paper, we focus on 3D point cloud semantic segmentation, which is one of the subtasks of 3D point cloud processing. Similar to the task of 2D image semantic segmentation, the purpose of 3D point cloud semantic segmentation is to classify point clouds with the same semantic information in space into the same category, which lays an important foundation for scene understanding and information perception in complex environments.

Due to the irregularity and disorder of point clouds in 3D space, the methods in the field of 2D images cannot be directly applied to point clouds. There is still a certain gap in the performance of semantic segmentation between point clouds and 2D images. 3D point cloud semantic segmentation methods include projection-based methods [2–5] and voxel-based methods [6–10]. Since the proposal of the PointNet [11] network, 3D point cloud processing has evolved into point-based methods. In recent years, more methods for 3D point cloud semantic segmentation have been proposed, and the performance of 3D point cloud semantic segmentation is improving year by year [12–15]. Due to the rapid development of methods based on Transformer [16] in 2D image vision in recent

years [17,18], the point-based method in particular has also opened a new stage in the development of 3D point cloud semantic segmentation. Recently, various transformer-based methods have showed strong performance in 3D point cloud segmentation, including PT [19], PCT [20], Voxel Transformer [21], etc. These methods not only show the feasibility of transformer-based methods in 3D point cloud semantic segmentation, but also show that 3D point cloud semantic segmentation methods based on Transformer still have rich potential research value.

In this work, our semantic segmentation network is based on Transformer. Since most networks are currently insufficient and incomplete in the feature extraction process, or only focus on the design of local feature extraction, we consider extracting and merging features from both local and global levels to improve the receptive field and feature aggregation capability of point clouds. Specifically, in the local feature layer, we adopt the multi-head attention mechanism in Transformer to aggregate the surrounding neighbor point features from different feature spaces. At the global feature level, we use the ratio of the distance between the local surrounding points and the boundary points in the global scope as the global features. Finally, we fuse the local and global features and embed them on the conventional convolution-based segmentation network to enhance the effect of semantic segmentation.

In summary, our main contributions include the following.

- We adopt the multi-head self-attention mechanism to extract local features of point clouds and use the position-distance information of point clouds in 3D space to extract global features.
- We propose a novel semantic segmentation network named LLGF-Net that can fuse effective features from the local and global feature levels of point clouds, which has superior semantic segmentation performance to most existing methods.
- We conduct quantitative experiments and various ablation studies on the challenging Stanford Large-Scale 3D Indoor Spaces (S3DIS) dataset [22] with the method proposed in this paper. The experimental results demonstrate the rationality and effectiveness of our method.

2. Related Works

In this section, we introduce the methods for 3D point cloud semantic segmentation that have been used in recent years. Different from 2D image segmentation methods, 3D point cloud semantic segmentation methods can be roughly divided into the following three types: projection-based, voxel-based and point-based.

Projection-based methods. 3D point cloud semantic segmentation can be regarded as a task extension of 2D image semantic segmentation. Therefore, the main idea in projection-based methods is to project point clouds to 2D images through methods such as multi-view, and then apply existing 2D image methods for semantic segmentation. For example, Lawin et al. [3] use multiple virtual views to project point clouds to 2D images; they then reproject and fuse the views from different locations to obtain the semantic labels of point clouds. However, the disadvantage of this method is that it is easily affected by factors such as viewing-angle selection and projection occlusion during the projection conversion process, and some geometric information may be lost during the projection process, which can affect the segmentation accuracy of point clouds.

Voxel-based methods. The main idea in voxel-based methods is to discretize point clouds in 3D space. Specifically, point clouds are converted into 3D grids that can be arranged regularly, as in the pixels of 2D images, and then segmented using a 3D convolutional network [23,24]. For example, Tchapmi et al. [9] preprocess point clouds by voxelization; they then use a 3D, fully convolutional neural network for prediction, introduce trilinear interpolation to map the results back to the original point cloud and use a fully connected conditional random field (CRF) to predict the semantic labels. Although existing sparse convolution methods can solve the problem of space occupation and re-

source consumption in high-resolution voxels [6,25], there is still the possibility of losing local details in voxel construction.

Point-based methods. The early representative work of point-based methods is PointNet [11], which directly uses original point clouds as inputs without complex preprocessing and postprocessing and uses shared multi-layer perceptions (MLPs) and max pooling to extract features. The success of PointNet and its improved version, PointNet++ [26], shows that feature extraction based on the original point clouds can achieve better results than previous methods based on projection and voxels. This also led to point-based methods gradually becoming the mainstream algorithm direction of 3D point cloud semantic segmentation. Based on this research method, various efficient point-based aggregation networks have emerged to extract features from point clouds, including the use of CNN [27–31], RNN [32–34], Graph [35–37], and various custom-feature extraction modules. In recent years, the network using the Transformer mechanism has outperformed CNN in 2D image processing and achieved excellent performance. The core component of Transformer is self-attention, which can obtain attention scores by calculating the similarity between the query and the key generated by the input data and use the scores to weight the value generated by the input data to generate new features [16]. It is with the help of self-attention that Transformer can extract feature associations in the input sequence, which seems to be very effective for the feature learning of point clouds with location attributes. Recently, various Transformer-based networks have been proposed. These methods continuously improve the state-of-the-art performance of 3D point cloud segmentation. For example, Point Transformer [19] achieves excellent performance by using Transformer to construct local feature extraction modules to obtain the contextual information of point clouds, and its success also demonstrates the effectiveness of Transformer in 3D point cloud segmentation. Guo et al. [20] propose the PCT network, which is based on Transformer, to construct offset attention with offset invariance. Lai et al. [14] propose Stratified Transformer, which overcomes the problem of the limited effectiveness of the receptive field. It makes full use of Transformer to obtain the contextual features of remote points, and improves the state-of-the-art performance.

3. Our Methods

In this section, we introduce the self-attention mechanism and the multi-head attention module in local feature extraction. Furthermore, we introduce the use of distance information from point clouds in 3D space to extract global features. Next, we introduce how to fuse local features and global features. Finally, we introduce the overall network architecture of learning local and global feature fusion for 3D point cloud semantic segmentation.

3.1. Local Feature Extraction

The self-attention mechanism is the core component of the Transformer network structure. The calculation form is shown in Equation (1). It effectively implements the learning of associated features between input sequences. In this paper, we apply this mechanism to point clouds for feature extraction of neighbor points on the local scale.

$$\begin{gathered} \text{Attention}(Q, K, V) = \text{Softmax}(\frac{Q \bullet K^T}{\sqrt{d_k}})V \\ Q = W^q X, K = W^k X, V = W^v X \end{gathered} \tag{1}$$

where X is the feature vectors of input points. Q (query), K (key) and V (value) are learned through the learnable weight W. d_k is expressed as the feature dimension of key.

The self-attention mechanism can be regarded as the aggregation of information similar to its own characteristics in each part of the input sequence, in order to effectively obtain the long-distance contextual information of the input sequence. In this paper, the self-attention module used in the feature extraction process of point clouds is shown in Figure 1. It is worth noting that in Figure 1, we take a feature vector as the input in order to clearly illustrate the whole process of feature extraction using the self-attention (SA)

module. First, each input point feature vector obtains query, key, and value through a fully-connected layer (FC). Next, each key and value can obtain the key and value corresponding to the nearest neighbor points in their local space range through K-Nearest Neighbor (KNN). Next, we make a slight change to Equation (1) with reference to PT [19], using subtraction of query and key to replace the dot product. We use the spatial distance between the surrounding neighbor points and the input point as the positional encoding. Finally, the weighted summation of attention scores and values is obtained to generate a new point-feature vector.

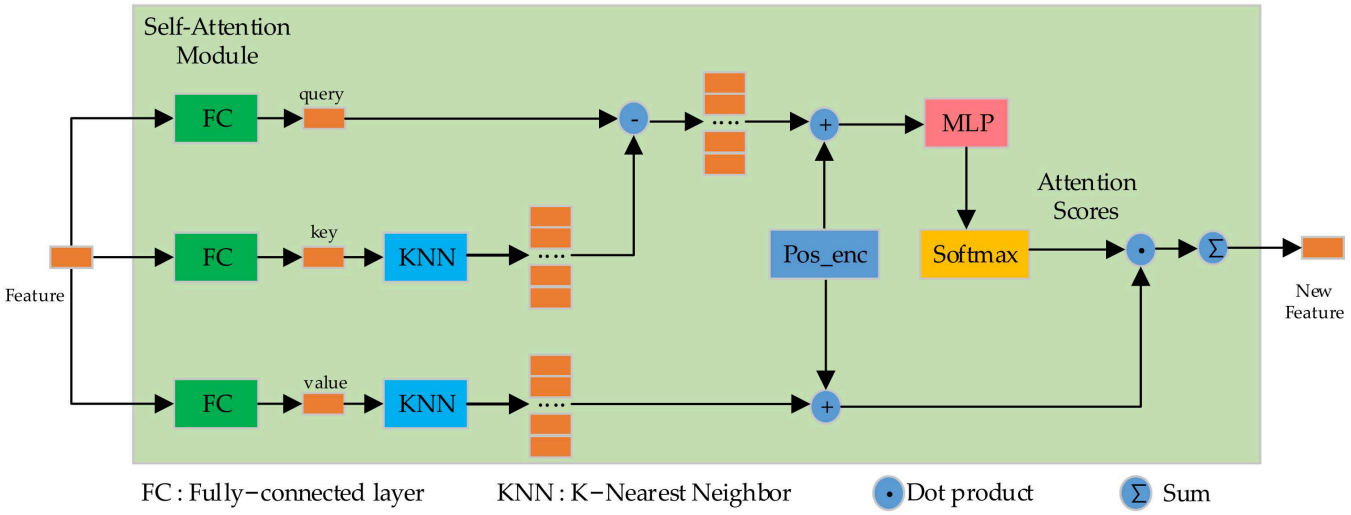

Figure 1. Self-attention (SA) module.

Based on the SA module, we adopt the self-attention mechanism to learn the features of point clouds at different levels, and then splice and combine these features to obtain the more effective information of point clouds. Therefore, we use the multi-head attention mechanism in Transformer, in which each self-attention module is called a head. Its calculation form is shown in Equation (2).

$$\begin{aligned} x_1 \oplus x_2 \cdots \oplus x_n &= X_{Input} \\ h_i &= \mathrm{F}(W_i^{(q)} x_i, W_i^{(k)} x_i, W_i^{(v)} x_i) \\ X_{output} &= W_o(h_1 \oplus h_2 \cdots \oplus h_n) \end{aligned} \tag{2}$$

where F represents the function of self-attention, X represents the input and output feature vectors, h represents the output feature of each self-attention module, n represents the number of multi-heads, $\oplus$ represents the feature concatenation operation, and x_i represents each sub-feature vector entering the self-attention mechanism. All uses of W are learnable parameters for generating queries, keys, values, and output features.

According to the Equation (2), we construct the module structure of the multi-head attention mechanism in our network, as shown in Figure 2. In general, a feature vector of the input point is split into sub-vector features. Next, they are sent to the self-attention mechanism module for feature extraction. Finally, the new sub-features obtained from each part are concatenated to obtain more abundant local aggregated features.

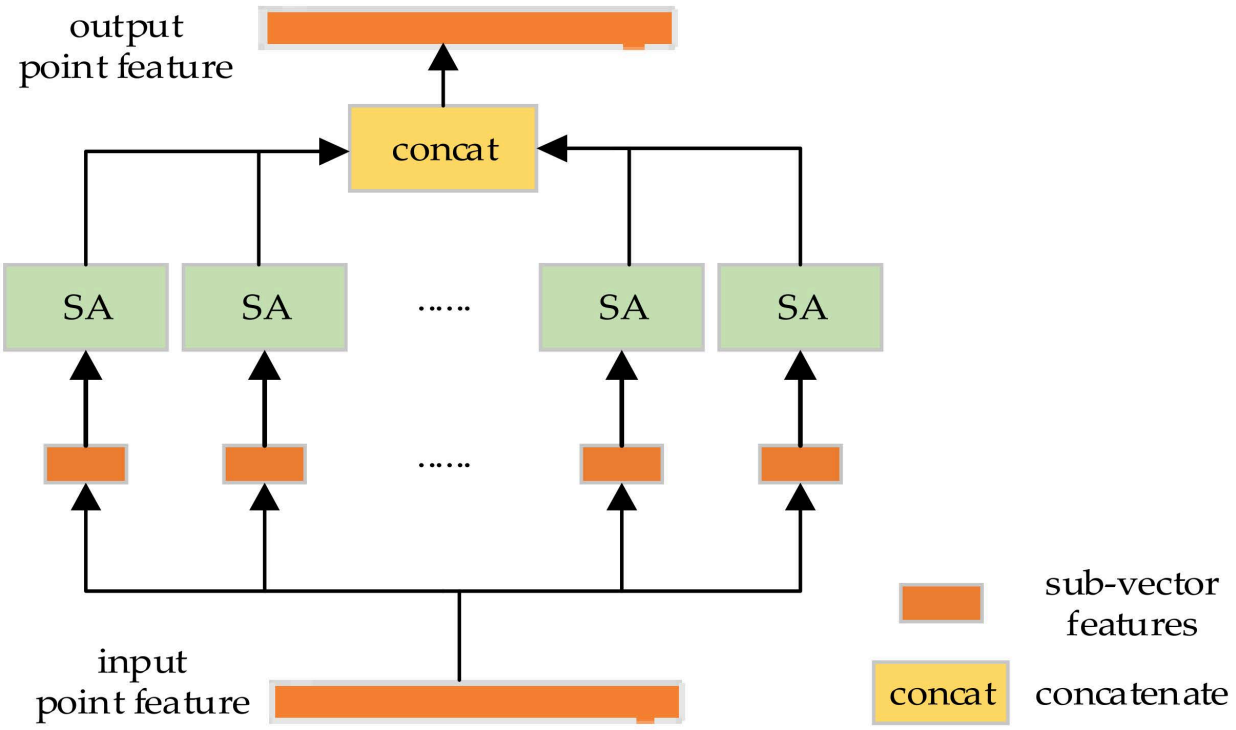

Figure 2. Local feature extraction module based on multi-head attention.

3.2. Global Feature Extraction

The same type of object in different scenes has different styles, but its geometric feature structure is often invariant [13]. Therefore, in order to better aggregate the effective feature information of point clouds, we use the global feature extraction module in the entire spatial range of point clouds to obtain the global feature information. Its calculation is defined as Equation (3):

$$F_{global} = \Phi(f_{xyz} \oplus \mathrm{R}(D_{L\_max}, D_{G\_max})) \tag{3}$$

where F_{global} represents the information of global features, f_{xyx} is the XYZ coordinate features of the input point, D_{L_max} is the position distance of the farthest point in the local surrounding, D_{G_max} is the position distance of the farthest input point in the global space, Φ is a one-hidden-layer MLP with two linear layers, one batch-normalization layer, and one Relu activation layer, R represents the defined ratio function, and $\oplus$ represents the concatenation operation.

The structure of the global feature extraction module is shown in Figure 3. As can be seen from the figure, the XYZ coordinate information of point clouds is used as the input of the module, and the nearest neighbors of each point in the local neighborhood are obtained through KNN. We can obtain the ratio of the distance between the farthest point in local neighborhood and the farthest point in global space. Next, we concatenate the input XYZ features and the distance-feature ratios of point clouds and send them to the one-hidden-layer MLP to obtain the final global features.

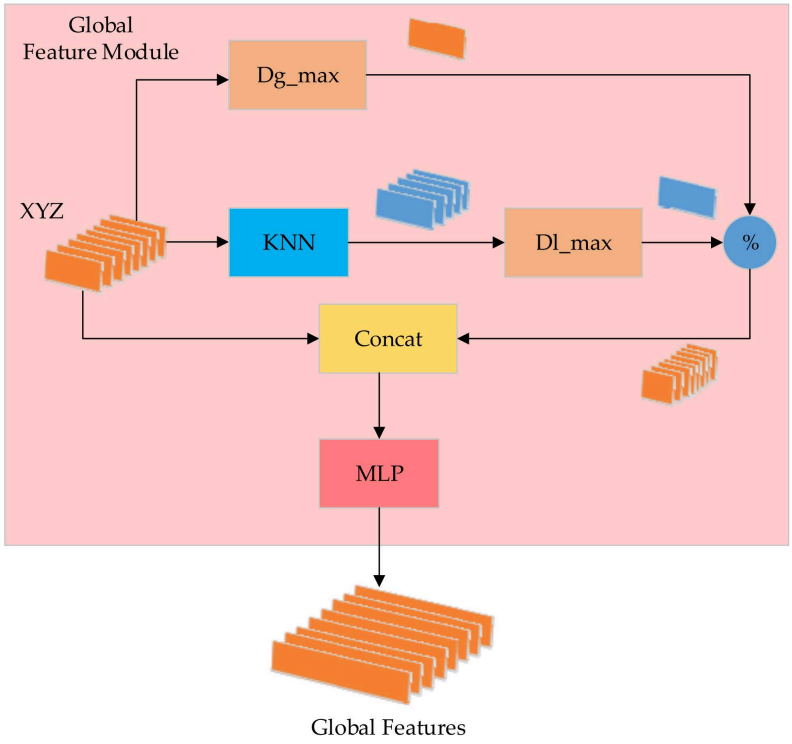

Figure 3. Global feature extraction module based on the position-distance information.

3.3. Local and Global Feature Fusion

After obtaining local features and global features, we construct the local and global feature fusion (LGF) module to aggregate the two features to obtain more effective point cloud features. The architecture of the LGF module is shown in Figure 4. The inputs of the module are the coordinates and features of point clouds. The input coordinates of point clouds enter the global feature extraction module in Section 3.1, and the point cloud features enter the local feature extraction module in Section 3.2 via a one-layer MLP with one linear layer, one batch normalization layer, and one Relu activation layer. We send the local and global features output by two modules to the one-layer MLP through an activation function (Leaky Relu) for feature fusion. Next, we add the original input points and aggregated point features by using the residual connection. Finally, we obtain the final new features through an activation function (Relu), which effectively combine local and global feature information of point clouds.

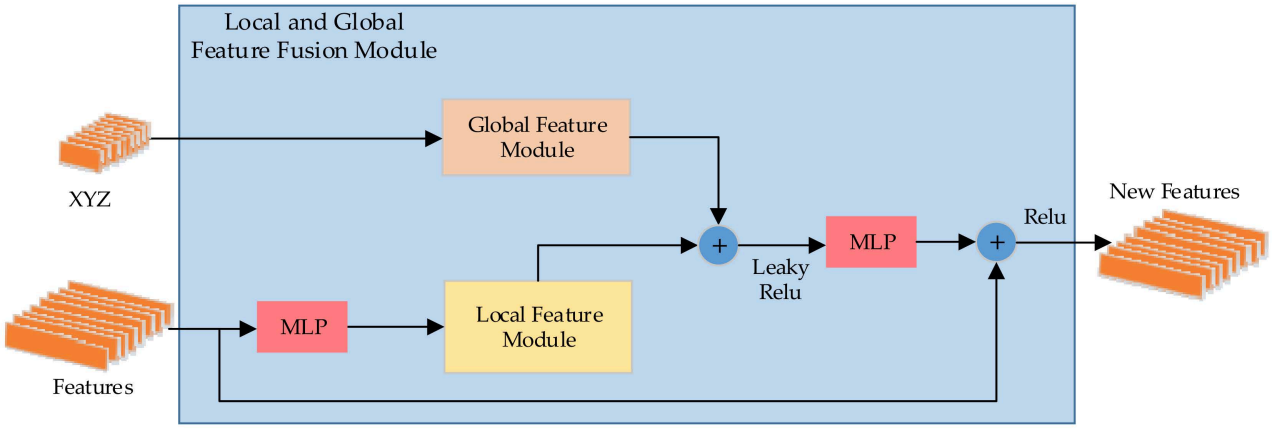

Figure 4. Local and global feature fusion (LGF) module.

3.4. Overview of Network Architecture

We embed the LGF module into the encoder–decoder network for 3D point cloud segmentation, and we can obtain the overall structure of our LLGF-Net, as shown in Figure 5. As can be seen from Figure 5, the encoder stage of the network corresponds to the process of downsampling and feature extraction, and the decoder stage corresponds to the process of upsampling and feature recovery. The inputs to the network are original point clouds, which contain XYZ and RGB; their feature size is N × 6. After all input points pass through the one-layer MLP, they enter into downsampling and LGF module. The downsampling consists of three steps, including farthest point sampling (FPS), max-pooling, and the one-layer MLP. Next, the point features after downsampling at each layer are extracted by the LGF module. During the entire stage of the encoder, the number of point clouds is reduced from N to N/256 layer by layer, and the feature dimension is increased from 6 to 512. The output in the downsampling stage passes through the one-layer MLP and then enters the upsampling stage. The upsampling adopts the method of trilinear interpolation to restore the number of point clouds. We aggregate the point cloud features from the encoder stage with the sampled points by using the skip connection, after which the features are fed into the LGF module. The number of point clouds in the entire decoder stage is restored from N/256 to N, and the size of point cloud features is reduced from 512 to 32. Finally, we can obtain the category label of each point cloud through the prediction network constructed by the fully connected network.

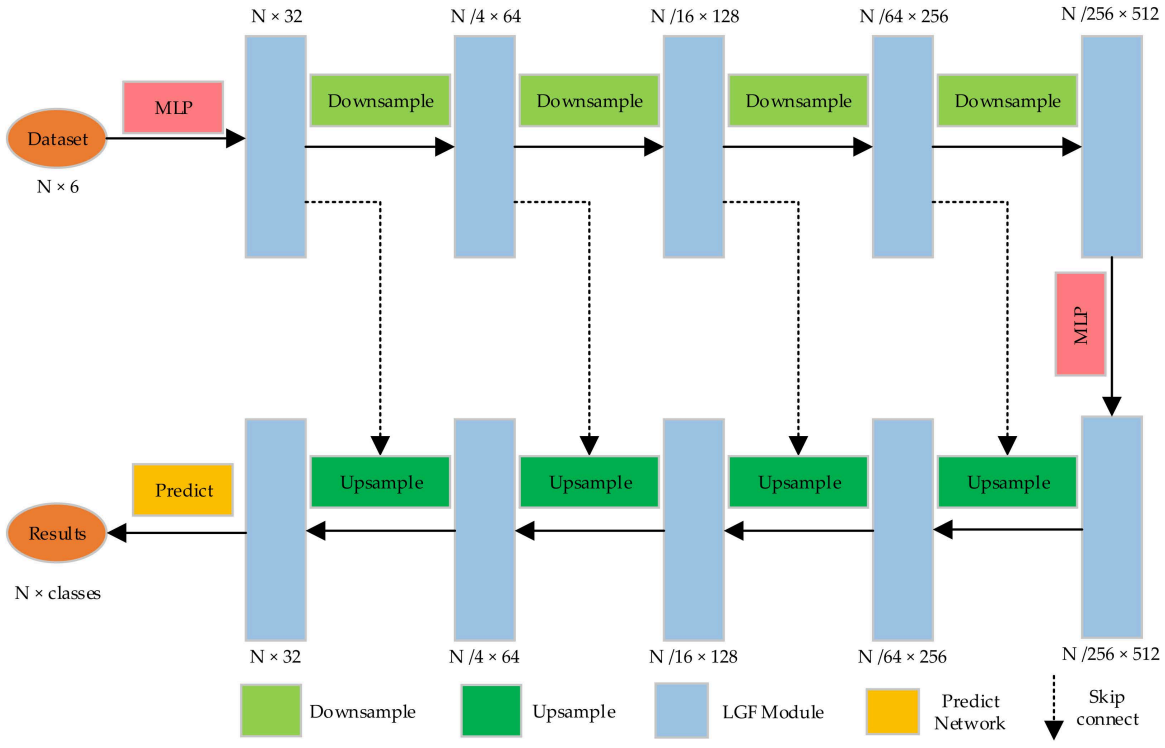

Figure 5. Architecture of the LLGF-Net.

4. Experiments

In this section, we first introduce the 3D point cloud dataset (S3DIS) used for experiments and performance evaluation metrics for semantic segmentation. Next, we evaluate our LLGF-Net based on quantitative experiments and 6-fold cross-validation experiments on the dataset S3DIS and compare the experimental results with some mainstream methods. Finally, we report our ablation experiments to demonstrate the rationality and effectiveness of each component of our network.

4.1. Dataset and Evaluation Metrics

Dataset. In the semantic segmentation experiments, we use the S3DIS dataset for network training and testing. S3DIS is a large scene indoor 3D point cloud dataset. It contains five indoor areas in Stanford University, including a total of 271 rooms, and is divided into six areas for training and testing. The dataset includes 11 scenes, such as offices and conference rooms, and 13 categories, such as ceilings, floors, and walls. The features of each point in the dataset include XYZ, RGB color, and normalized coordinate values in the Cartesian coordinate system.

Evaluation metrics. We use common semantic segmentation evaluation metrics, namely mean intersection-over-union (mIoU), mean class accuracy (mAcc), and overall accuracy (OA), to evaluate the experimental results. The OA is the proportion of correctly predicted point clouds in the total number of point clouds, and Equation (4) is its calculation form. The mAcc is calculated as Equation (5), which is used to calculate the proportion of correctly predicted point clouds in each category and then calculate the average of all the categories. The mIoU is the most important metric, which represents the average of the intersection-over-union ratio for each category in the dataset. Its calculation form is shown in Equation (6).

$$OA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}} \tag{4}$$

$$mAcc = \frac{1}{k+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij}} \tag{5}$$

$$mIoU = \frac{1}{(k+1)} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \tag{6}$$

where represents the prediction of category i as j, p_{ii} represents the correct prediction for category i and k is the number of categories in the dataset.

4.2. Results on S3DIS

Implementation details. Our training and testing are performed on TITAN RTX GPU. We implement the entire LLGF network in Pytorch. We set the initial learning rate to 0.5 and decay the learning rate by 10% at a fixed epoch. The network optimizer adopts SGD and sets the momentum and weight decay to 0.9 and 0.001.

Quantitative results. In order to clearly compare with the results of mainstream algorithm networks, we test on area 5 of the dataset S3DIS and train on the remaining areas, in accordance the experimental strategy adopted by most methods on S3DIS. The quantitative experimental results of our method and other comparative methods are shown in Table 1. As can be seen from the table, our method achieves a mIoU metric of 68.0% and a mAcc metric of 74.4%, which is superior to other methods. In general, the semantic segmentation of indoor scenes is very difficult, especially in some categories that are difficult to distinguish, such as boards. In the S3DIS dataset, since the backgrounds of whiteboards are white walls, they are easily confused if the feature extraction is not sufficient. The quantitative results in Table 1 show that our method achieves a mIoU metric of 79.8% on boards, which is also the only method that can be used to achieve more than 70% of the results compared with other methods. In addition, the best segmentation results are obtained on categories such as ceilings, walls, columns, and tables.

6-fold cross-validation. Next, we conduct a 6-fold cross-validation experiment on the S3DIS dataset to better evaluate the performance of our network, and the results are shown in Table 2. As can be seen from the table, our network attains a mIoU of 71.4% and an OA of 89.6%. Our method outperforms MLP-based methods, such as PointNet [11] and PointNet++ [26], graph-based methods, such as DGCNN [35], CNN-based methods, such as PointCNN [28], and other methods. In short, the results of the 6-fold cross-validation

show that our method based on local and global feature fusion is superior to other methods and is an effective method for 3D point cloud semantic segmentation.

Table 1. Quantitative results of S3DIS Area 5 dataset.

Methods	mIoU (%)	mAcc (%)	Ceiling	Floor	Wall	Beam	Column	Window	Door	Table	Chair	Sofa	Bookcase	Board	Clutter
PointNet [11]	41.1	49.0	88.8	97.3	69.8	0.1	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
SegCloud [9]	48.9	57.4	90.1	96.1	69.9	0.0	18.4	38.4	23.1	70.4	75.9	40.9	58.4	13.0	41.6
PointCNN [28]	57.3	63.9	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 [29]	58.3	67.0	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 [38]	60.3	66.6	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 [39]	61.9	68.3	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
SegGCN [40]	63.6	70.4	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
KP-Conv [41]	67.1	72.8	92.8	97.3	82.4	0.0	23.9	58.0	69.0	81.5	91.0	75.4	75.3	66.7	58.9
Ours	68.0	74.4	94.1	98.2	85.1	0.0	30.6	60.6	73.5	89.5	79.6	72.3	63.1	79.8	57.6

Table 2. Quantitative results of S3DIS with 6-fold cross-validation.

Methods	OA (%)	mAcc (%)	mIoU (%)
PointNet [11]	78.6	66.2	47.6
PointNet++ [26]	81.0	67.1	54.5
DGCNN [35]	84.1	–	56.1
RsNet [32]	–	66.5	56.5
PointCNN [28]	88.1	75.6	65.4
PointWeb [38]	87.3	76.2	66.7
ShellNet [42]	87.1	–	66.8
RandLA-Net [12]	88.0	82.0	70.0
Ours	89.6	80.6	71.4

Visualization. Finally, in order to more clearly show the semantic segmentation results of the LLGF network on 3D point clouds, we visualize the typical conference room and office scenes in area 5, as shown in Figure 6. Figure 6 shows the original point clouds, ground truth, and baseline with PointNet++ [26], as well as our results. As can be seen from the figure, we obtain excellent results of 3D point cloud semantic segmentation, which are exceedingly close to the ground truth. In particular, the circled parts in the first and third rows show that our method can correctly segment slender objects, such as columns and doors, compared to the baseline. Furthermore, the circled parts in the second and fourth rows show that our method can still obtain excellent results when segmenting the complex and indistinguishable categories of boards and bookcases.

4.3. Ablation Studies

The experimental results on S3DIS demonstrate the effectiveness of our method. To further evaluate the rationality of the design of each core component in our approach, we conduct extensive ablation studies. In order to compare the results with other methods, we still conduct the ablation studies on area 5 of the S3DIS dataset.

Ablation study on LLGF-Net. To explore the effectiveness of each module in the overall network, we further conduct an ablation study on each module. The experimental results are shown in Table 3. As can be seen from the table, we remove the global feature extraction module; the final mIoU result is 65.3%. The performance is reduced by 2.7% compared to the result of the original network, which shows that the global feature extraction module can provide effective feature information for 3D point cloud segmentation. We then remove the local feature-extraction module. The final result of the mIoU is reduced by 12.7%, which shows that the local feature extraction module plays a significant role in the overall feature extraction and is the key to the understanding of the point cloud scene features.

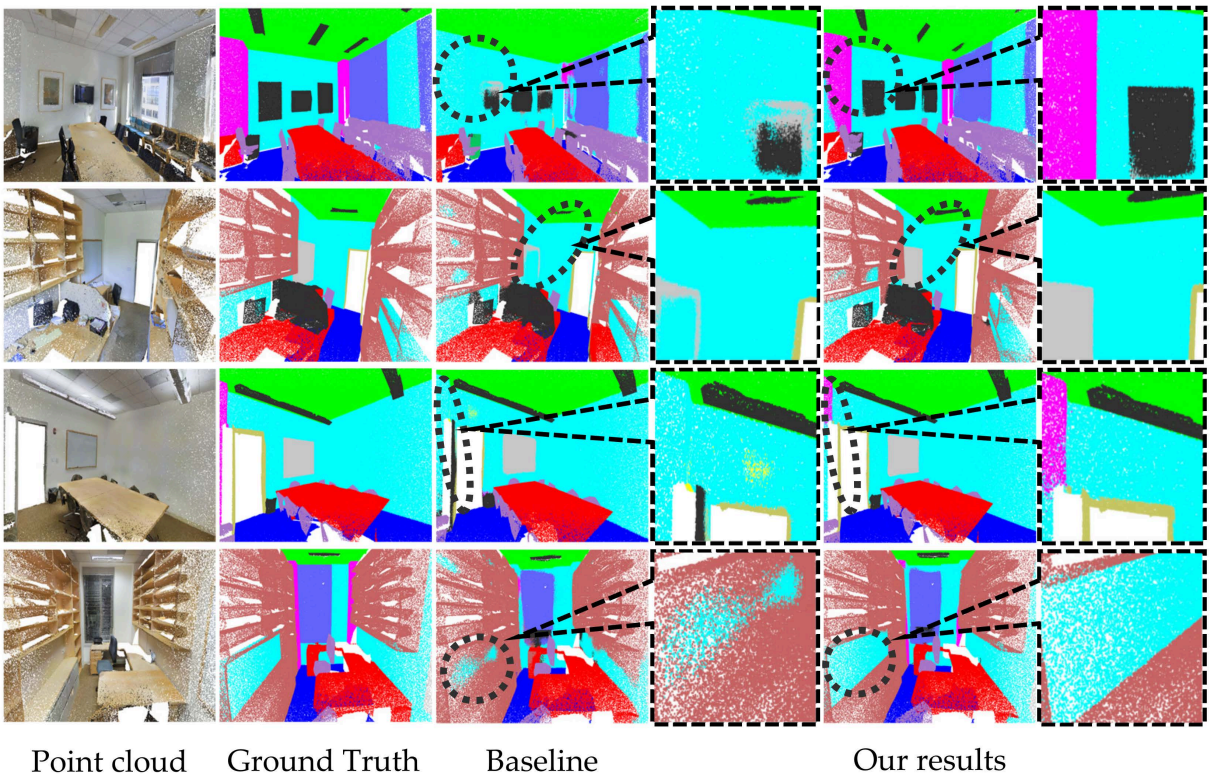

Figure 6. Visualization of semantic segmentation results on the S3DIS dataset.

Table 3. Ablation study: effectiveness of each module.

	mIoU (%)	↓ mIoU (%)
Original network	68.0	0
Removing globa Feature-extraction module	65.3	2.7
Removing local Feature-extraction module	55.3	12.7

Ablation study on local feature-extraction module. In order to further explore the details of the parameter setting of the local feature extraction module, we change the parameter values of the module in turn, and the experimental results are shown in Table 4. In the local feature extraction module of our original network, the number of heads is set to 16. As can be seen from the table, if the number of heads is reduced to 8, the final mIoU decreases by 0.6%; if the number of heads is reduced to 4, the final mIoU decreases by 1.0%. Therefore, it can be seen that when the multi head parameter is set to 16, our network achieves the best performance.

Table 4. Ablation study: Multi-head parameter setting.

	Parameter	mIoU (%)	↓ mIoU (%)
Number of multi-heads	16	68.0	0
	8	67.4	0.6
	4	67.0	1.0

Ablation study on feature fusion. After obtaining the local and global features of point clouds, in order to obtain more representative features, we design a feature fusion process, which is shown in Figure 4. We compare two fusion methods, including addition and concatenation, in our ablation experiment. As can be seen from the results in Table 5, if we fuse features by using concatenation, the final result of the mIoU is 67.5%, which is 0.5% lower. Therefore, we fuse the local, global, and input features by using addition, which can provide more effective feature information for 3D point cloud semantic segmentation.

Table 5. Ablation study: feature fusion.

	Method	mIoU (%)	↓ mIoU (%)
Feature fusion	Addition	68.0	0
	Concatenation	67.5	0.5

5. Conclusions

In this paper, we propose a novel network named LLGF-Net for 3D point cloud semantic segmentation based on the fusion of local features and global features, which achieves superior performance to most current methods. In terms of local feature extraction, we adopt multi-head attention to aggregate the contextual features of neighboring point clouds in local space. In terms of global feature extraction, we use the local and global position-distance information of point clouds in 3D space to extract the global spatial features. Next, we fuse the two features to provide more representative point cloud features for 3D point cloud semantic segmentation. Finally, we conduct quantitative experiments and various ablation studies on the S3DIS dataset, which demonstrate the rationality of each component in our network and the effectiveness of our network in 3D point cloud semantic segmentation.

Author Contributions: Conceptualization, J.Z. and X.L.; methodology, J.Z. and X.L.; software, J.Z.; validation, J.Z., X.L., X.Z. and Z.Z.; investigation, J.Z., X.L., X.Z. and Z.Z.; data curation, X.Z. and Z.Z.; writing—original draft preparation, J.Z. and X.L.; writing—review and editing, J.Z., X.L., X.Z. and Z.Z.; visualization, J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The S3DIS dataset presented in this study is openly available on the website. Available online: http://buildingparser.stanford.edu/dataset.html (accessed on 7 June 2022).

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

References

1. Guo, Y.; Wang, H.; Hu, Q.; Liu, L.; Benmamoun, M. Deep learning for 3D point clouds: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 4338–4364. [CrossRef] [PubMed]
2. Kundu, A.; Yin, X.; Fathi, A.; Ross, D.; Brewington, B.; Funkhouser, T.; Pantofaru, C. Virtual multi-view fusion for 3D semantic segmentation. In Proceedings of the European Conference on Computer Vision, Edinburgh, UK, 23–28 August 2020; pp. 518–535.
3. Lawin, F.J.; Danelljan, M.; Tosteberg, P.; Bhat, G.; Khan, F.S.; Felsberg, M. Deep projective 3D semantic segmentation. In Proceedings of the International Conference on Computer Analysis of Images and Patterns, Ystad, Sweden, 22–24 August 2017; pp. 95–107.
4. Milioto, A.; Vizzo, I.; Behley, J.; Stachniss, C. Rangenet++: Fast and accurate lidar semantic segmentation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Macao, China, 3–8 November 2019; pp. 4213–4330.
5. Boulch, A.; Le Saux, B.; Audebert, N. Unstructured point cloud semantic labeling using deep segmentation networks. *3DOR* **2017**, *3*, 17–24.
6. Graham, B.; Engelcke, M.; Van Der Maaten, L. 3D semantic segmentation with submanifold sparse convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 9224–9232.
7. 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, 18–22 June 2018; pp. 2626–2635.
8. 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, Korea, 27 October–3 November 2019; pp. 8500–8508.
9. 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, Qingdao, China, 10–12 October 2017; pp. 537–547.
10. Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3D shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1912–1920.

11. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3D classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern recognition, Honolulu, HI, USA, 27 January–1 February 2017; pp. 652–660.
12. 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, 14–19 June 2020; pp. 11108–11117.
13. 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, 19–25 June 2021; pp. 14504–14513.
14. Lai, X.; Liu, J.; Jiang, L.; Wang, L.; Zhao, H.; Liu, S.; Qi, X.; Jia, J. Stratified Transformer for 3D Point Cloud Segmentation. *arXiv* **2022**, arXiv:2203.14508.
15. Tang, L.; Zhan, Y.; Chen, Z.; Yu, B.; Tao, D. Contrastive Boundary Learning for Point Cloud Segmentation. *arXiv* **2022**, arXiv:2203.05272.
16. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Lukasz, K.; Polosukhin, I. Attention is all you need. In Proceedings of the Conference and Workshop on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
17. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4 May–8 June 2021.
18. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 10012–10022.
19. Zhao, H.; Jiang, L.; Jia, J.; Torr, P.H.; Koltun, V. Point transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 16259–16268.
20. Guo, M.; Cai, J.; Liu, Z.; Mu, T.; Martin, R.R.; Hu, S. Pct: Point cloud transformer. *Comput. Vis. Media* **2021**, *7*, 187–199. [CrossRef]
21. Mao, J.; Xue, Y.; Niu, M.; Bai, H.; Feng, J.; Liang, X.; Xu, H.; Xu, C. Voxel transformer for 3D object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 3164–3173.
22. 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.
23. Maturana, D.; Scherer, S. Voxnet: A 3D convolutional neural network for real-time object recognition. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Hamburg, Germany, 28 September–2 October 2015; pp. 922–928.
24. Song, S.; Yu, F.; Zeng, A.; Chang, A.X.; Savva, M.; Funkhouser, T. Semantic scene completion from a single depth image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 27 January–1 February 2017; pp. 1746–1754.
25. Choy, C.; Gwak, J.; Savarese, S. 4D spatio-temporal convnets: Minkowski convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 January 2019; pp. 3075–3084.
26. Charles, R.Q.; Li, Y.; Su, H.; Guibas, L.J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In Proceedings of the Thirty-first Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
27. Hua, B.S.; Tran, M.K.; Yeung, S.K. Pointwise 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. 984–993.
28. Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; Chen, B. Pointcnn: Convolution on X-transformed points. In Proceedings of the Thirty-second Conference on Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018.
29. 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.
30. Boulch, A. ConvPoint: Continuous convolutions for point cloud processing. *Comput. Graph.* **2020**, *88*, 24–34. [CrossRef]
31. Engelmann, F.; Kontogianni, T.; Leibe, B. Dilated point convolutions: On the receptive field size of point convolutions on 3D point clouds. In Proceedings of the IEEE International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020; pp. 9463–9469.
32. Huang, Q.; Wang, W.; Neumann, U. Recurrent slice networks for 3D segmentation of point clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2626–2635.
33. Engelmann, F.; Kontogianni, T.; Hermans, A.; Leibe, B. Exploring spatial context for 3D semantic segmentation of point clouds. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 716–724.
34. Ye, X.; Li, J.; Huang, H.; Du, L.; Zhang, X. 3D recurrent neural networks with context fusion for point cloud semantic segmentation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 403–417.
35. 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.* **2019**, *38*, 146. [CrossRef]

36. 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, 15–21 January 2019; pp. 10296–10305.
37. Liang, Z.; Yang, M.; Deng, L.; Wang, C.; Wang, B. Hierarchical depthwise graph convolutional neural network for 3D semantic segmentation of point clouds. In Proceedings of the International Conference on Robotics and Automation, Brisbane Convention & Exhibition Centre, Brisbane, Australia, 21–26 May 2018; pp. 8152–8158.
38. 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, 15–21 January 2019; pp. 5565–5573.
39. 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, Korea, 27 October–3 November 2019; pp. 10433–10441.
40. 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, 14–19 June 2020; pp. 11611–11620.
41. 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, Korea, 27 October–3 November 2019; pp. 6411–6420.
42. Zhang, Z.; Hua, B.S.; Yeung, S.K. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–3 November 2019; pp. 1607–1616.