Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 9, No. 3, 2024

3D Point Cloud Semantic Segmentation based on Multi-scale Dense Nested Networks

Zishuo Wang [1], Tianxiang Lai [1], Yufeng Wang [1], Xingquan Gao [2]

[1] School of Information and Control Engineering, Jilin Institute of Chemical Technology, Jilin, Jilin 132022, China
[2] School of Information Engineering, Jilin Vocational College of Industry and Technology, Jilin, Jilin 132022, China

Abstract: Aiming at the problem that the relationship between geometric features and semantic features is ignored in the point cloud data downsampling process, which leads to inaccurate segmentation of object boundaries and structural details, this paper proposes a 3D point cloud semantic segmentation network based on multi-scale dense nested type. Firstly, a dense nested network architecture is constructed by nesting multiple multi-scale feature fusion modules to fuse multi-scale features of different directions between encoder-decoder paths, so as to effectively propagate local ge-ometric context information and enhance the ability of cross-scale information interaction. Secondly, a local feature aggregation unit is constructed in the multi-scale feature fusion module, which strengthens the structural awareness within the local point set based on graph convolution and attention mechanism, and promotes the complementarity of local geometric features and abstract semantic information. Then, the cross-layer multi-loss supervision module is combined to further optimize the multi-scale feature propagation, which makes the network training more stable and improves the point cloud segmentation accuracy. Finally, this paper verified the proposed network on the S3DIS dataset. The experimental results show that the proposed network has the mean intersection over Union of 71.2% and the overall accuracy of 88.7%, which is 1.2 and 0.7 percentage points higher than RandLA-Net, respectively, which proves that the proposed network can effectively improve the accuracy of 3D point cloud semantic segmentation.

Keywords: Point Cloud Semantic Segmentation; Nested Networks; Multi-scale Feature Fusion; Dense Connection.

1. Introduction

Point cloud data can provide rich geometric shape and depth information when describing three-dimensional scenes, and is widely used in artificial intelligence fields such as automatic driving, robotics, and three-dimensional reconstruction[1]. The purpose of point cloud semantic segmentation is to identify the semantic labels of each point in the point cloud for semantic understanding and environment perception of scene objects. With the development of 3D sensing technology and the generalization of data annotation[2],Deep learning methods have achieved good results on tasks such as image denoising[3], Super resolution reconstruction[4], object classification[5] and semantic segmentation[6].

At present, many novels or enhanced semantic segmentation methods have been proposed. Su et al.[7] proposed an image-based multi-view convolutional neural network (MVCNN), which used CNN to extract multi-view two-dimensional image features to improve segmentation accuracy. Boulch et al.[8] marked RGB and depth composite views pixel by pixel and proposed SnapNet network to improve the segmentation ability in the face of large-scale point cloud scenes. Riegler et al.[9] built the OctNet network by dividing space by octree layers, which significantly reduced the computing and memory requirements. Qi et al.[10] proposed PointNet network, which realized end-to-end semantic segmentation for the first time. However, it ignores the rich local geometric features, resulting in low semantic segmentation accuracy for complex scenes. In order to capture local geometric shapes and details from the point cloud and enhance context dependence, Qi et al.[11] adopted multi-scale sampling and group fusion mechanism to improve the PointNet network, and then proposed the PointNet++ network, and extracted finer granularity semantic features by stacking multiple feature abstraction layers. Hu et al.[12] proposed that RandLA-Net network can effectively capture and retain local geometric features in point clouds by introducing local feature aggregation module to model the spatial relationship between points. Li et al.[13] developed the X conversion operator PointCNN and used it to learn the characteristics of the input point cloud in order to obtain high-level semantics. Wu et al.[14] extended the dynamic filter to a new convolution operation called PointConv, which can be used to compute the features of a set of points in three-dimensional space. In addition, Du Jing et al. [15] proposed a point cloud segmentation network based on multi-feature fusion and residual optimization, and optimized the network training by adding residual blocks into the feature aggregation module. Liu et al.[16] used a deep network PosPool and a simple local aggregation operator to analyze point clouds. Xu et al.[17] proposed a position-adaptive convolution operator PAConv with dynamic kernel components to make full use of the characteristics of neighborhood point clouds. Hao Wen et al. [18] built a multi-feature fusion dynamic graph convolutional neural network by mapping low-dimensional geometric features to high-dimensional space, extracting and fusing geometric shape features and high-level semantic features, and improved the ability to capture network feature distribution.

To sum up, the existing 3D point cloud semantic segmentation methods mainly include multi-view-based methods, voxel-based methods, and point-based methods.

The multi-view based method and the voxel-based method will lose part of the feature information in the process of data form conversion, which also increases the computation cost and memory overhead. The point based method can retain the spatial characteristics of the original point cloud data, and can effectively solve the problem of irregularity and disorder of the point cloud, so it has become the mainstream research

method. However, in the process of point sampling, the learning of local features is limited to the input point cloud, and the edges of some sparse point clouds are still difficult to be accurately segmented. Although the random sampling method can reduce the amount of point cloud processed in the process of network downsampling, it sacrifices the accuracy of the point cloud feature extraction.

To solve the above problems, this paper proposes a multi-scale dense nested network architecture, which can extract rich point feature information at different network depths. The multi-scale feature fusion module utilizes the ability of graph convolution and attention mechanism to capture information in the local feature aggregation unit, so that the geometric structure information from different directions and levels can be integrated more comprehensively. In order to ensure the steady-state of the multi-scale feature fusion process, a cross-layer multi-loss monitoring module is proposed, which can realize the efficient training control of the network by adding sub-losses and adjusting class weights.

2. Network Model

2.1. Dense Nested Network Architecture

The network proposed in this paper is based on multi-scale dense nested network architecture. As shown in Figure 1, the black solid and dotted arrows are direct connection and dense jump connection respectively, the blue and purple arrows are up-sampling and down-sampling processes respectively, and the circle part is the multi-scale feature fusion module (MFFM), which defines the output feature of each MFFM as, where and respectively represent the depth and width of the current network layer. Therefore, the output characteristics of each MFFM in the second layer are shown in formula (1):

$$F_{i,j}=\begin{cases}\psi\left[D\left(F_{i-1,j}\right)\right] & j=0\\ \psi\left[FA\left(F_{i,j-1}\right),D\left(F_{i-1,j}\right),U\left(F_{i+1,j-1}\right)\right] & j=1,2,3\\ \psi\left[FA\left(F_{i,j-1}\right),D\left(F_{i-1,j}\right),U\left(F_{i+1,j-1}\right),F_{i,0}\right] & j=4\end{cases} \tag{1}$$

among $\psi[,]$ is the feature splice, $FA(\ )$ is the aggregation of features in the block, $D(\ )$ and $U(\ )$ are downsampled and upsampled respectively.

$F_{0,0}\rightarrow F_{5,0}$ indicates the encoder path direction, $F_{5,0}\rightarrow F_{0,5}$ is the decoder path direction. MFFM is densely nested in coding-decoder paths of different depths to accurately extract point-by-point local geometric details, thus propagating multi-scale features more effectively.

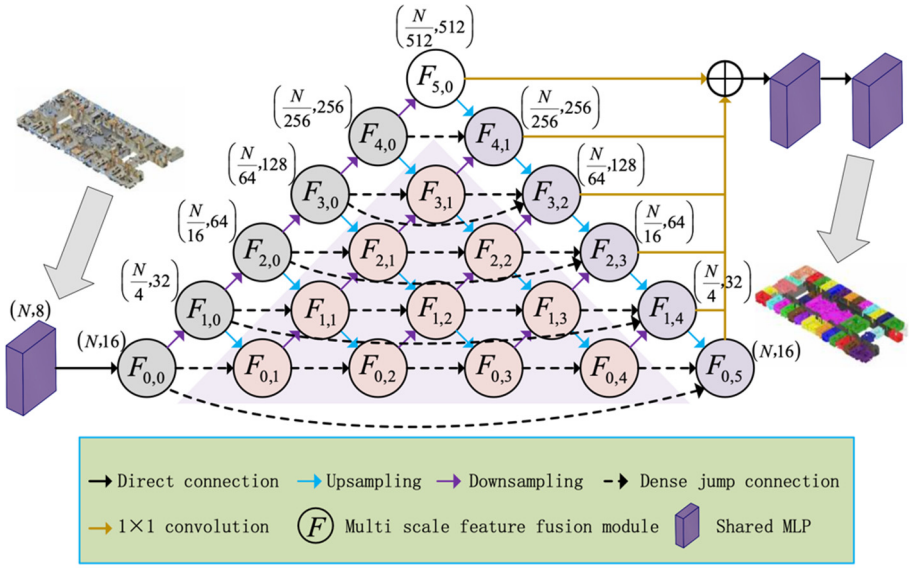


Fig 1. MDNN network architecture

First, the point cloud is input into a shared fully connected layer, extracting the initial features of each point from the original point cloud. Secondly, the learned feature information is input into $F_{0,0}$,Through horizontal and vertical connections, feature information can be propagated between different levels. Compared with methods such as farthest point sampling[19], and inverse density sampling [20], MDNN uses random sampling operations to reduce the spatial resolution of point features, thereby improving computational efficiency and abstraction. Then, along the direction of the encoder path, the network gradually sparse the point cloud (where N is the number of points) in the order of (N→N/4→N/16→N/64→N/256→N/512), each level of MFFM learns the multi-scale of point features at multiple network levels, and gradually expands the receiving domain to obtain higher level of abstract semantic features. At the same time, along the direction of the decoder path, the spatial resolution of the feature of each point is sympathetically restored through the nearest neighbor interpolation operation in the order of (16→32→64→128→256→512), so as to preserve the geometric details. Each level of MFFM receives multi-scale nested features and gradually restores the geometric details of the origin features. Finally, two shared full connection layers are introduced to map point features to semantic prediction tags.

Fundamentally different from the original U-shaped architecture with a single coding-decoder path, MDNN further links information flows in multiple directions and scales, and realizes the propagation and fusion of multi-scale features, thus integrating richer geometric information (such as edge and shape details) and more abstract semantic information in the point cloud. The cross-scale information interaction ability of the network is enhanced, which helps to improve the accuracy of the network model to the point-level object segmentation in the face of complex scenes.

2.2. Multi-scale Feature Fusion Module

In order to alleviate the impact of inaccurate segmentation of point boundaries and structural details caused by the reduction of point cloud resolution in the process of downsampling, multi-scale feature fusion modules are nested in different levels of MDNN. By combining multiple information branches, the module obtains high-level abstract features with rich semantic representation capabilities, thereby improving semantic consistency and realizing multi-scale information interaction and fusion. As shown in Figure 2, the input is a multi-scale feature with different feature resolutions from its previous block.

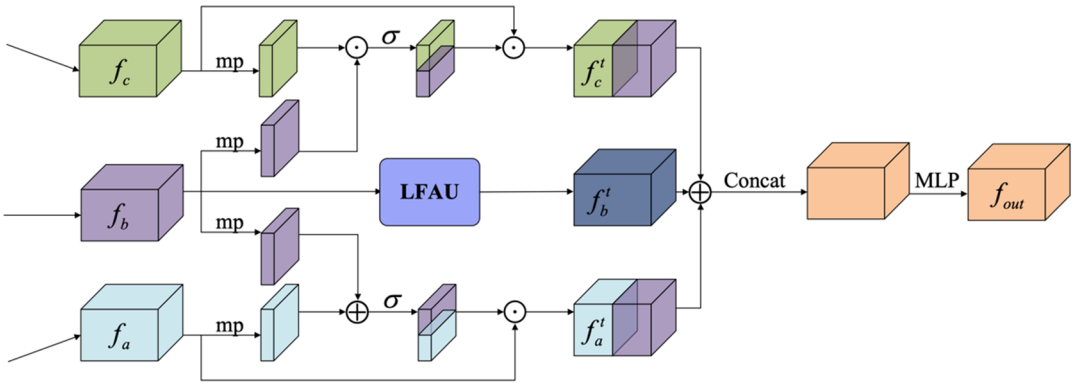


Fig 2. Multi-scale feature fusion module

Take the feature fusion module of $F_{i,j}\left(i>0\right)$ as an example, First, it receives the subsampled data f_a, the same resolution data f_b from horizontal transmission, and the upsampled data f_c,from vertical transmission, which can be expressed as:

$$f_a = D\left(F_{i+1,j-1}\right) \tag{2}$$

$$f_b = F_{i,j-1} \tag{3}$$

$$f_c = U\left(F_{i-1,j}\right) \tag{4}$$

Secondly, the feature channels of the three scales are squeezed into the same dimension, and the features of different layers are reused for fusion transformation to obtain the transform features f_a^t, f_b^t and f_c^t, which can be expressed as:

$$f_a^t = f_a \odot \sigma\left(mp\left(f_a\right) \oplus mp\left(f_b\right)\right) \tag{5}$$

$$f_b^t = LFAU\left(f_b\right) \tag{6}$$

$$f_c^t = f_c \odot \sigma\left(mp\left(f_b\right) \odot mp\left(f_c\right)\right) \tag{7}$$

Where, $\odot$ is dot product, $\oplus$ is addition, σ is sigmoid activation function, mp is maxpooling, $LFAU$ is Local feature aggregation unit(LFAU).

Then, the transformation features are fused and spliced. Finally, MLP is used to reduce the channels of stacked features, so as to output fusion feature f_{out}. The multi-scale feature fusion process can be expressed as:

$$f_{out} = MLP\left(\left[f_a^t, f_b^t, f_c^t\right]\right) \tag{8}$$

Where, MLP is the shared fully connected layer, and $[,]$ is the feature splicing.

In order to capture local features of cross-scale sampling points at different depths, LFAU is used as the basic feature extraction unit in MFFM. As shown in Figure 3, LFAU is mainly composed of two Feature graph convolution operators (FGCO) and two Attention pooling layers (APL). Before each FGCO, the three-dimensional spatial coordinates of the point cloud were entered into the graph convolution operator together with the original position features and the input point features for relative position coding, so as to reuse the overall morphological information and enhance the module's ability to capture local spatial details and semantic information. To more fully capture the detail and complexity of point cloud data, a Dense skip connection (DSC) is used to connect two FGCO and APL after sharing the full connection layer for more efficient feature propagation.

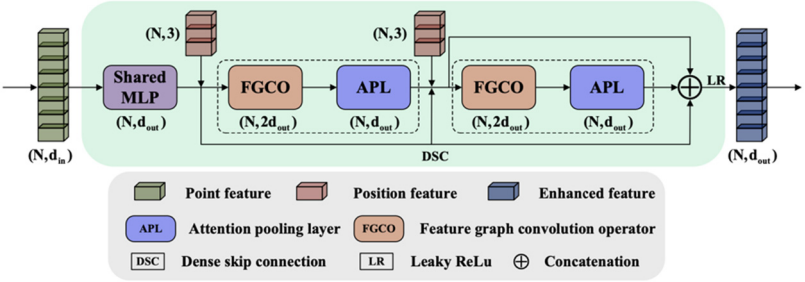


Fig 3. Local feature aggregation unit

The details of FGCO are shown in Figure 4, where the key step is the construction of the local neighborhood map. For an unordered point cluster with N points, it can be defined as: $p = \{p_1, p_2, \ldots, p_N\} \in R^d$,Dimension d is generically expressed as the characteristic dimension of a certain layer. The point characteristics corresponding to the point convergence are: $f_p = \{f_1, f_2, \ldots, f_N\} \in R^{N \times d}$.

First, in the process of constructing the spatial domain of the neighborhood graph, the center point for each input is determined p_i ,FGCO calculates the distance on Euclidean space based on the K-nearest neighbor algorithm, and selects

the k neighbor node $p_k = \left\{ p_{ij} \in R^{N\times 3} \middle| j = 1,2,\ldots,k \right\}$ closest to point p_i and its corresponding neighbor node features $f_k = \left\{ f_{ij} \in R^{N\times d} \middle| j = 1,2,\ldots,k \right\}$.

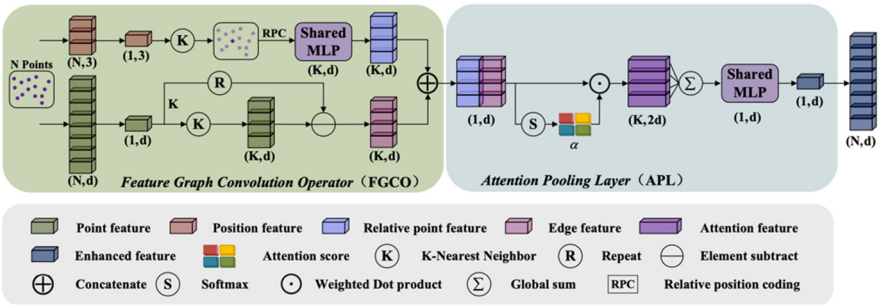


Fig 4. Feature map convolution operator and attention pooling layer details

Secondly, a spatial relationship is established for the distance and direction relationship between each point p_i and its k nearest neighbor node p_k, and a directed graph G_i is used to implicitly represent the local structure of the point cloud, as shown in formula (9) :

$$\begin{cases} G_i = (V_i, E_i) \\ V_i = \left\{ f_{ij} \cup f_i \in R^{N\times d} \middle| j = 1,2,\cdots k \right\} \\ E_i = \left\{ e_{ij} = g_\theta\left(f_{ij} - f_i\right) \in R^{N\times d} \middle| j = 1,2,\cdots k \right\} \end{cases} \tag{9}$$

Where V_i and E_i represent the node feature set and edge feature set respectively in the local graph G_i, $f_{ij} - f_i$ represents the relationship feature between two points, $g_\theta(\cdot)$ is a nonlinear function, θ is a learnable parameter, e_{ij} is the directed feature distance between the reference center point p_i and its JTH adjacent point feature f_{ij} calculated by using the Euclidean distance as the edge feature.

Thirdly, the relative position coding (RPC) followed by the shared multi-layer perceptron is introduced, The local relative feature l_{ij}^r of the reference central point p_i is obtained through the position information of its adjacent point p_{ij} ,which can be represented as follows:

$$l_{ij}^r = MLP\left(\left[p_i, p_j, \left(p_{ij} - p_i\right), \left\| p_{ij} - p_i \right\| \right]\right) \tag{10}$$

Where p_i and p_{ij} are coordinate points with xyz feature channels, $\|\cdot\|$ is the calculation of Euclidean distance, and $[,]$ is the feature concatenation.

Then, the edge feature e_{ij} of each reference central point p_i and the corresponding local relative feature s_{ij}^r are combined to obtain the enhanced edge feature $\hat{e}_{ij}$, as shown in formula (11) :

$$\hat{e}_{ij} = \left[e_{ij}, l_{ij}^r \right] \tag{11}$$

In order to make more effective use of the most critical information in each enhanced edge feature, the attention pooling to graph pooling strategy based on self-attention mechanism is used to aggregate features. Each enhanced edge feature $\hat{e}_{ij}$ of reference center point p_i is aggregated to extract local geometric features with high discriminating power. For the resulting set of enhanced edge features $\hat{E}_{ij} = \left\{ \hat{e}_{i1}, \hat{e}_{i2}, \ldots, \hat{e}_{ij} \right\}$,an attention coefficient α_{ij} is learned for each enhanced edge feature using the shared activation function $\sigma(\cdot)$ and $Soft\max$ classifier. The aggregation process is shown in formula (12):

$$\alpha_{ij} = Soft\max\left(\frac{\exp\left(\sigma\left(\hat{e}_{ij}, A\right)\right)}{\sum_{j=1}^{k} \exp\left(\sigma\left(\hat{e}_{ij}, A\right)\right)} \right) \tag{12}$$

Where $A \in R^{1\times d}$ is the learnable attention coefficient matrix.

Finally, each enhanced edge feature $\hat{e}_{ij}$ is multiplied and weighted with the corresponding learned attention coefficient α_{ij} , and the central point feature is updated as shown in formula (13):

$$f_i^p = \sum_{j=1}^{k} \left(\hat{e}_{ij} \cdot \alpha_{ij} \right) \tag{13}$$

Where f_i^p is the aggregation feature of the reference central point p_i in the local graph, and $\cdot$ is the dot product operator.

Reference ResNet [21] residual module design, FGCO chose to enhance the perceptual range of each point to capture the features of neighboring points by connecting and stacking two FGCO and APL with dense jumps. In order to fully demonstrate the performance of the feature aggregation unit, the local feature extraction capability in the point cloud is visualized. As shown in Figure 5, the KNN search range is set to 5 in this paper, with red dots representing the central point, purple dots representing the nearest neighbors of the central point, dotted arrows representing the flow direction of information aggregation, and dotted circles representing the range of receptive fields of the central point. In the first stage, shallow local features are captured, and after information flow, the information receptive field of the central point is further expanded in the second stage, and the context information in a larger area is extracted by using the feature maps accumulated in the previous stage. By expanding the sensing domain, MFFM promotes feature propagation, ensures computational efficiency, and enhances the ability of the network to capture local features of point cloud data, thus improving the depth extraction ability of the network for multi-scale features.

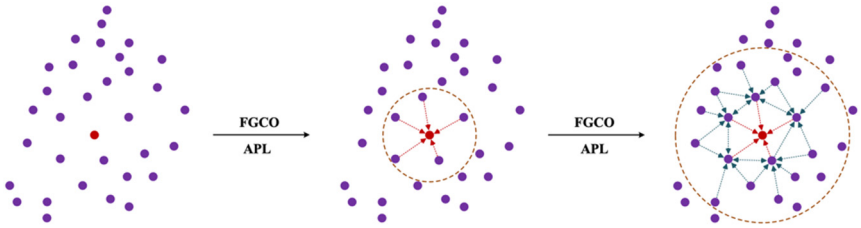


Fig 5. Visualization of local feature extraction capabilities

2.3. Cross-layer Multi-Loss Monitoring Module

The dense nested network architecture proposed in this paper spans multiple scales in feature fusion, thus increasing the complexity of feature fusion. However, this complexity also makes the network training process difficult to control. To ensure effective training, we need to integrate features efficiently. Inspired by deep supervision network[22] and multi-scale structure sensing network[23], MDNN proposes a cross-layer multi-loss supervision module. The module can deeply optimize multiple feature fusion processes across the network, improve the ability of network training control and feature fusion, and further adjust the weight of categories to improve the stability of the network learning process, thus improving the segmentation accuracy of high point cloud.

As shown in Figure 6, the module adds sub-loss at the end of each decoder, and through cross-level sub-loss supervision, sufficient feedback can be obtained in the training process of different deep networks, thus realizing the training control of the network local architecture. Each subloss consists of a 1×1 convolution layer and a weighted cross entropy loss function L_{wce}, L_{wce} advance along the decoder path and dynamically adjust the weights based on the points in each category to solve the class imbalance problem and ensure the effective fusion and backpropagation of multi-scale supervision information. The weighted cross entropy loss function is shown in equation (14):

$$L_{wce} = -\sum_{i}^{N_p} w_i p(y_i) \log(p(\hat{y}_i)) \tag{14}$$

Where N_p represents the number of total sample points, $p(y_i)$ is the true distribution value of the target point, $p\left(\hat{y}_i\right)$ is the predicted distribution value of the target point, w_i represents the weight coefficient of the i th sample point, which can be expressed as:

$$w_i = \frac{\sum_{t=1}^{13} N_n}{N_n + 0.02}, n = 1,2,3,\cdots,13 \tag{15}$$

Where N_n represents the number of sample points belonging to the n th category, and 13 represents the number of categories in the S3DIS dataset. To avoid the situation where the denominator is zero, add a fraction of 0.02 to the denominator.

The final loss is calculated based on the predicted output of each sub-loss, and the entire network uses the multi-scale loss function L_m to precisely supervise the training. L_m consists of two parts, namely the cross-entropy loss function L_{wce} and the correlation loss L^i_{wce}. It can be expressed as:

$$L_m = L^0{}_{wce} + \beta \cdot \sum_{i=1}^{I} L^i{}_{wce} \tag{16}$$

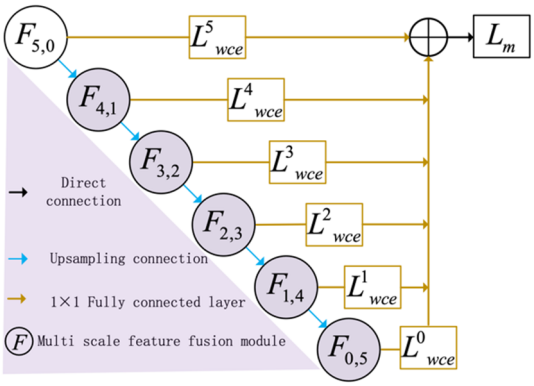


Fig 6. Cross-layer multi-loss monitoring module

Where β is the scale factor controlling the two loss functions, I is the number of network layers, which is set as 5 in this study, $L^0{}_{wce}$ and $L^i{}_{wce}$ represent the output loss of layer 0 and layer i of the network.

3. Experimental Results and Analysis

In this paper, the indoor data set S3DIS[24] is used to verify the performance of MDNN network in 3D point cloud semantic segmentation task. Firstly, comparison experiments were conducted with mainstream networks such as PointNet[10], RandLA-Net[12] and PointCNN[13], and then the effectiveness of each innovation point of MDNN was verified through ablation experiments. Finally, based on all the experimental results, the overall performance of MDNN is analyzed and summarized.

3.1. Experimental Data Set

Stanford large 3D indoor spatial dataset S3DIS[24] is a large indoor 3D point cloud dataset with pixel level semantic annotation provided by Stanford University. It contains six teaching and office areas with a total of 695,878,620 3D points, with color information and semantic labels. The data is semantically divided into 272 rooms, annotated with 12 semantic elements and an additional clutter label, and the characteristics of the points consist of 9 dimensions in total, including 3D coordinates, RGB features, and normalized positions. It is commonly used for indoor semantic segmentation[25]. Select area 5 as the test set and the rest as the training set. The data in region 5 and other regions are not in the same building, and there are certain differences in the objects in the scene. This division can more accurately test the accuracy of MDNN semantic segmentation and the generalization ability of point cloud data recognition.

3.2. Environment Configuration and Evaluation Indicators

The network proposed in this study trained 200 rounds on two GeForce RTX 3090 Gpus (24GB video memory). The

CUDA10.6 GPU running environment was built on Ubuntu20.04 system, and the deep learning framework was based on Python and pytorch platform. In this paper, the nearest neighbor point K value used in the training and testing model is 16, the learning rate starts from 0.01, decutes at a rate of 0.3 after every 10 rounds, and the momentum is 0.95. Adam optimization algorithm is used to minimize the loss function.

This paper follows the standard practice of ISPRS[26] (International Society for Photogrammetry and Remote Sensing) competition, using mean crossover ratio (mIoU), class mean accuracy (mAcc) and overall accuracy (OA) as evaluation indicators. mIoU represents the intersection/union of true values and predicted values, and is used to evaluate the accuracy of all object class segmentation; mAcc represents the average recognition accuracy of the model for each category, and is used to measure whether the recognition ability of the model for all categories is balanced. OA represents the percentage of the total input category that predicts the correct category and is used to assess the accuracy of the overall segmentation. Assume that TP, FP, TN, and FN represent true, false positive, true negative, and false negative respectively, and k represents the number of semantic categories in the dataset. Its calculation formula is as follows:

$$mIoU = \frac{1}{k+1}\sum_{i=0}^{k}\frac{TP}{TP+FP+FN} \tag{17}$$

$$mAcc = \frac{1}{k+1}\sum_{i=0}^{k}\frac{TP_i}{TP_i+FN_i} \tag{18}$$

$$OA = \frac{TP+TN}{TP+FP+FN+TN} \tag{19}$$

3.3. Analysis of Experimental Results

In this paper, the performance of MDNN on large-scale scene semantic segmentation task was tested on S3DIS point cloud dataset. Table 1 shows the comparison of evaluation indexes of semantic segmentation between the network in this paper and nine advanced networks.

Table 1. Quantitative experimental results of different networks on the S3DIS dataset (6-fold cross-validation)

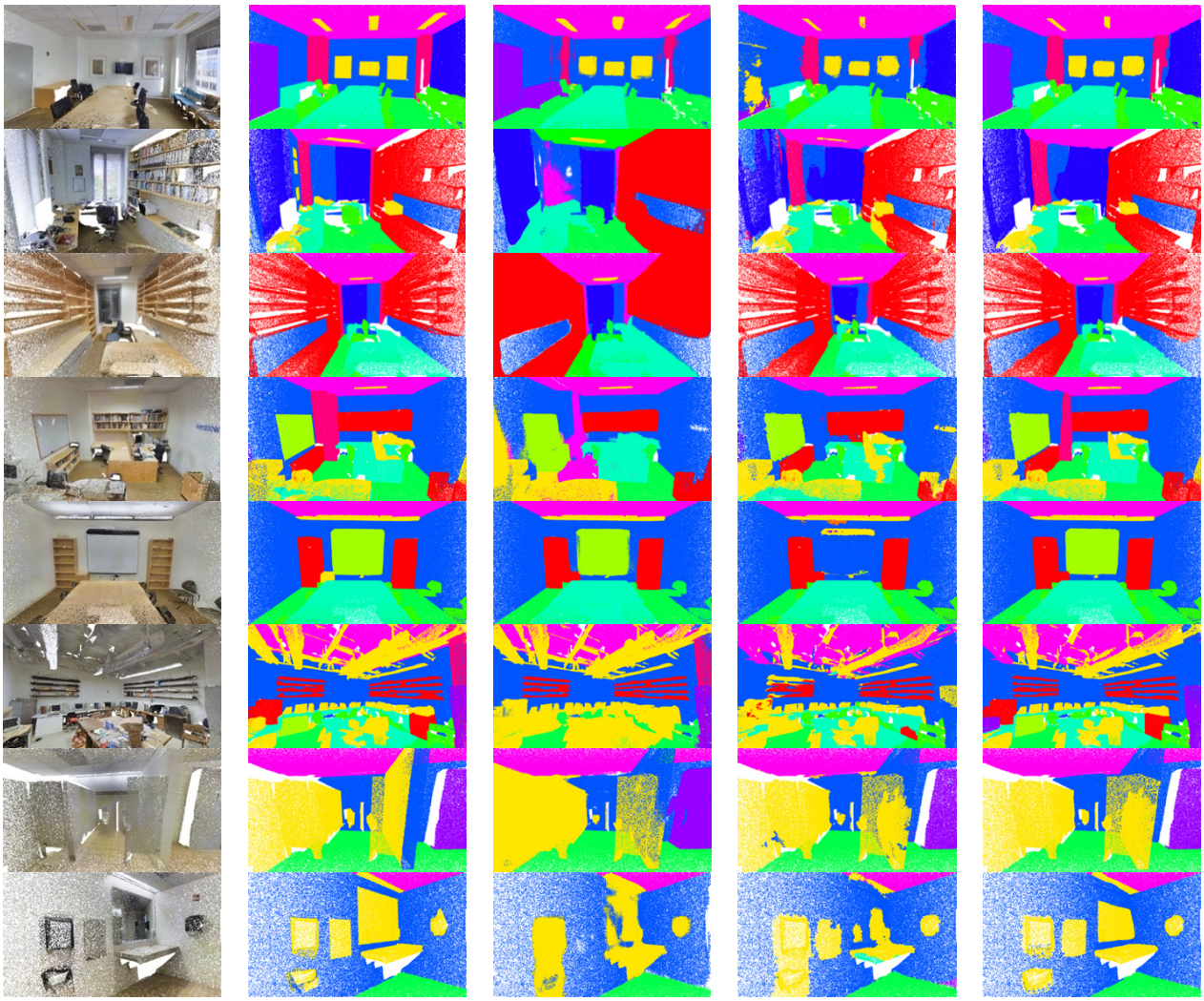
Network	mIoU(%)	mAcc (%)	OA (%)
PointNet[10]	47.6	66.2	78.6
PointNet++[11]	54.5	67.1	81.0
DGCNN[27]	56.1	-	84.1
PointCNN[13]	65.4	75.6	88.1
GA-Net[28]	63.7	-	87.6
PointWeb[29]	66.7	76.2	87.3
RandLA-Net[12]	70.0	82.0	88.0
BAF-LAC[30]	71.7	82.5	88.2
SCF-Net[31]	71.6	82.7	88.4
MDNN	71.2	83.9	88.7

Table 2. Quantitative evaluation results of S3DIS dataset Area5

Network	ceiling	Floor	wall	beam	col	wind	door	table	chair	sofa	book	board	clutter
PointNet [10]	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
PointNet++ [11]	90.7	96.3	74.2	0.0	7.8	57.5	23.4	66.6	70.4	42.0	61.0	53.3	41.2
DGCNN [27]	93.0	97.4	77.7	0.0	12.0	47.8	39.8	67.4	72.4	23.2	52.3	39.8	46.6
PointCNN [13]	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
GA-Net [28]	92.9	97.8	81.3	0.0	27.8	60.3	41.7	78.3	86.7	71.4	69.9	65.8	53.9
PointWeb [29]	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
RandLA-Net [12]	91.1	95.6	80.2	0.0	24.7	62.3	47.7	76.2	83.7	60.2	71.2	70.1	53.9
BAF-LAC [30]	91.2	87.3	81.6	0.0	27.9	59.3	49.5	78.0	87.2	63.4	66.5	69.6	51.1
SCF-Net [31]	94.3	98.4	84.2	0.0	28.3	58.9	73.2	92.2	82.9	76.6	82.2	68.6	59.9
MDNN	93.8	98.6	82.4	0.0	27.7	63.1	71.6	89.4	85.5	73.8	79.3	70.3	55.9

As can be seen from Table 1, in the test results of 60% cross-validation, the mIoU of MDNN is 0.4% lower than that of SCF-Net and 0.5% lower than that of BAF-LAC, but MDNN has achieved better performance in mAcc and OA. mAcc improved by 7.7% and 1.9%, respectively, compared with PointWeb and RandLA-Net, indicating that the proposed network has a better segmentation accuracy for the average category in identifying the overall structure shape. Table 2 shows the results of the MDNN's quantitative evaluation of 13 category segmentation for S3DIS dataset region 5. It can be seen that MDNN achieves the best performance in the floor, window and panel categories, improving by 0.2%, 4.2% and 1.7%, respectively, compared with the recent representative network SCF-Net, and also achieves good segmentation accuracy for objects such as roofs, columns, tables and bookcases. Figure 7 visualizes some of the indoor scenarios

in test set area 5. It can be seen from the visualization results that MDNN is more accurate for the point segmentation at the edge of the furniture category, thanks to MFFM capturing the local geometric features and high-level abstract features of the sampling points on a multi-scale, reducing the feature difference between the same category. Compared with the segmentation results of RandLA-Net, MDNN achieves a more complete and smooth segmentation for the categories of objects such as walls, boards, and debris, which is due to the stacked feature graph convolution operator (FGCO) and attention pooling layer (APL) in the local feature aggregation unit, which retain the overall morphological information through the relative position encoded features. Expand the receptive domain to capture fine local features, so as to capture the complete structural features of objects in complex scenes.



a. Primary point cloud b. GroundTruth c. PointNet++ d. RandLA-Net e. MDNN

Fig 7. Visualization of semantic segmentation results of S3DIS dataset

3.4. Ablation Experiment

In order to further verify the effectiveness of each module in the MDNN network and check the details of each part of the network, five ablation experiments were designed on S3DIS region 5 using the mean intersection ratio (mIoU) as an evaluation index, and the results were shown in Table 3.

Table 3. MDNN network model ablation experiment

model	Dense nested network	Local feature aggregation unit	Cross-layer multi-loss monitoring module	mIoU(%)
(A)				65.8
(B)				67.4
(C)				67.9
(D)				71.2

In model (A), the advantages of cross-scale information interaction of dense nested network architecture are evaluated, and the original U-shaped architecture is used to replace the dense nested network architecture. Compared with model (A), the mIoU of model (D) is significantly increased by 5.4%, indicating that the dense nested network architecture has the greatest impact on network performance. The reason is that the dense nested network architecture is more conducive to the integration of rich geometric information and abstract semantic information, and enhance the ability of cross-scale information transmission. Model (B) evaluated the effect of local feature aggregation units on segmentation accuracy. Replace the local feature aggregation unit with the feature aggregation module in RandLA-Net [12]. It can be seen that the mIoU of model (D) is 3.8% higher than that of model (B), indicating that this unit can effectively extend the perception range of local neighborhoods, capture local spatial details and semantic information in different receptive fields, and thus enhance the feature information of each point. Model (C)

verifies the effectiveness of the cross-layer multi-loss monitoring module. The results show that the accuracy of mIoU can be improved by 3.3% after adding the cross-layer multi-loss monitoring module. This is because when the cross-layer multi-loss monitoring module is deleted, the whole network is limited to the deepest nested block to output losses, and the training process fails to use the multi-depth output prediction to enhance the output results. Model (D) is the complete network architecture proposed in this paper, which shows that each module can achieve the best performance by complementing each other.

4. Summary

In this paper, a multi-scale dense nested network MDNN is proposed for 3D point cloud semantic segmentation. Through the improved dense nested network architecture, the cross-layer flow of multi-scale features is realized by adding dense jump connections in horizontal and vertical directions. In addition, a local feature aggregation unit is designed in the multi-scale feature fusion module, which enlarges the receptive domain through feature propagation and expands the ability of the model to capture local features at different depths. At the same time, the cross-layer multi-loss monitoring module adopted in this paper controls and optimizes the feature learning process, which significantly improves the multi-scale feature fusion capability and the stability of the learning process. In this paper, quantitative experiments are carried out on the S3DIS benchmark dataset. The experiments show that compared with RandLA-Net network, the average intersection ratio is increased by 1.2%, the average accuracy is increased by 1.9%, and the overall accuracy is increased by 0.7%. The visual comparison results show that MDNN has higher segmentation accuracy for local edge details when facing sparse point clouds. The ablation research and analysis of different network designs demonstrate the high performance and effectiveness of each module in the network.

Acknowledgments

Fund Project: Science and Technology Research Project of Education Department of Jilin Province (JJKH20240314KJ).

References

[1] Wang C, Pastore F, Goknil A, et al. Automatic generation of acceptance test cases from use case specifications: an nlp-based approach[J]. IEEE Transactions on Software Engineering, 2020, 48(2): 585-616.

[2] Kolhatkar C, Wagle K. Review of SLAM algorithms for indoor mobile robot with LIDAR and RGB-D camera technology[J]. Innovations in Electrical and Electronic Engineering: Proceedings of ICEEE 2020, 2021: 397-409.

[3] Zaman F, Wong Y P, Ng B Y. Density-based denoising of point cloud[C]//9th International Conference on Robotic, Vision, Signal Processing and Power Applications: Empowering Research and Innovation. Springer Singapore, 2017: 287-295.

[4] Mandikal P, Radhakrishnan V B. Dense 3d point cloud reconstruction using a deep pyramid network[C]//2019 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2019: 1052-1060.

[5] Diab A, Kashef R, Shaker A. Deep learning for LiDAR point cloud classification in remote sensing[J]. Sensors, 2022, 22(20): 7868.

[6] Zhang J, Zhao X, Chen Z, et al. A review of deep learning-based semantic segmentation for point cloud[J]. IEEE access, 2019, 7: 179118-179133.

[7] Su H, Maji S, Kalogerakis E, et al. Multi-view convolutional neural networks for 3d shape recognition[C]//Proceedings of the IEEE international conference on computer vision, 2015: 945-953.

[8] Boulch A, Guerry J, Le Saux B, et al. SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks[J]. Computers & Graphics, 2018, 71: 189-198.

[9] Riegler G,Ulusoy O A,Geiger A. OctNet: Learning Deep 3D Representations at High Resolutions.[J]. CoRR,2016,abs/ 1611. 05009.

[10] Qi C R, Su H, Mo K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition, 2017: 652-660.

[11] Qi C R, Yi L, Su H, et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space[J]. Advances in neural information processing systems, 2017, 30.

[12] Hu Q, Yang B, Xie L, et al. Randla-net: Efficient semantic segmentation of large-scale point clouds[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020: 11108-11117.

[13] Li Y, Bu R, Sun M, et al. Pointcnn: Convolution on x-transformed points[J]. Advances in neural information processing systems, 2018, 31.

[14] Wu W, Qi Z, Fuxin L. Pointconv: Deep convolutional networks on 3d point clouds[C]//Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, 2019: 9621-9630.

[15] Du J and Cai G R. 2021. Point cloud semantic segmentation method based on multi-feature fusion and residual optimization [J]. Journal of Image and Graphics,2021, (05): 1105-1116.

[16] Liu Z, Hu H, Cao Y, et al. A closer look at local aggregation operators in point cloud analysis[C]//Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16. Springer International Publishing, 2020: 326-342.

[17] Xu M, Ding R, Zhao H, et al. Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: 3173-3182.

[18] HAO W, WANG H X, WANG Y. Semantic segmentation of three-dimensional point cloud based on spatial attention and shape feature [J]. Laser and Optoelectronics Progress, 2022, 59 (8): No. 0828004.

[19] Lin Y, Chen L, Huang H, et al. Beyond farthest point sampling in point-wise analysis[J]. arXiv preprint arXiv:2107.04291, 2021.

[20] Groh F, Wieschollek P, Lensch H P A. Flex-Convolution: Million-scale point-cloud learning beyond grid-worlds[C]// Asian Conference on Computer Vision. Cham: Springer International Publishing, 2018: 105-122.

[21] Targ S, Almeida D, Lyman K. Resnet in resnet: Generalizing residual architectures[J]. arXiv preprint arXiv:1603.08029, 2016.

[22] Lee C Y, Xie S, Gallagher P, et al. Deeply-supervised nets [C]// Artificial intelligence and statistics. Pmlr, 2015: 562-570.

[23] Ke L, Chang M C, Qi H, et al. Multi-scale structure-aware network for human pose estimation[C]//Proceedings of the european conference on computer vision (ECCV), 2018: 713-728.

[24] Armeni I, Sener O, Zamir A R, et al. 3d semantic parsing of large-scale indoor spaces[C]//Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 1534-1543.

[25] Babacan K, Chen L, Sohn G. Semantic segmentation of indoor point clouds using convolutional neural network[J]. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2017, 4: 101-108.

[26] Rottensteiner F, Sohn G, Gerke M, et al. Results of the ISPRS benchmark on urban object detection and 3D building reconstruction [J]. ISPRS journal of photogrammetry and remote sensing, 2014, 93: 256-271.

[27] Phan A V, Le Nguyen M, Nguyen Y L H, et al. Dgcnn: A convolutional neural network over large-scale labeled graphs [J]. Neural Networks, 2018, 108: 533-543.

[28] Deng S, Dong Q. GA-NET: Global attention network for point cloud semantic segmentation[J]. IEEE Signal Processing Letters, 2021, 28: 1300-1304.

[29] Zhao H, Jiang L, Fu C W, et al. Pointweb: Enhancing local neighborhood features for point cloud processing[C]// Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019: 5565-5573.

[30] Shuai H, Xu X, Liu Q. Backward attentive fusing network with local aggregation classifier for 3D point cloud semantic segmentation [J]. IEEE Transactions on Image Processing, 2021, 30: 4973-4984.

[31] Fan S, Dong Q, Zhu F, et al. SCF-Net: Learning spatial contextual features for large-scale point cloud segmentation [C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: 14504-14513.