Machine Learning (2022) 111:1327–1348
https://doi.org/10.1007/s10994-022-06148-1

SDANet: spatial deep attention-based for point cloud classification and segmentation

Jiangjiang Gao[1] · **Jinhui Lan**[1] · **Bingxu Wang**[1] · **Feifan Li**[1]

Received: 25 June 2021 / Revised: 28 November 2021 / Accepted: 11 January 2022 /
Published online: 30 March 2022

Abstract

Using deep learning to learn point cloud features directly have become one of the research hotspots in the field of 3D point cloud processing. The existing methods usually construct local regions, extract features from local regions, and then aggregate global features through multi-layer perceptron and maximum pooling layer. However, most of these processes do not consider the contribution of point cloud local features to the final decision and the spatial relationship between neighbor points, which limits the accuracy of 3D point cloud classification and segmentation. In this article, a novel network model called spatial depth attention network is designed to improve the accuracy of point cloud classification and segmentation, which embeds local depth attention mechanism into MLP layer to learn local neighborhood geometric representation. The local deep attention of the point cloud is obtained through the SDA module, and then combined with feature learning and local deep attention to effectively capture the local geometric structure. In order to achieve the best feature extraction ability, local depth attention features are combined with global features. Experiments show that SDANet achieves the same or better performance as the most advanced methods on several challenging benchmark datasets and tasks.

Keywords Attention · Classification · Deep learning · Point cloud · Segmentation

Editor: Andrea Passerini.

Co-first authors: Jiangjiang Gao and Jinhui Lan.

✉ Jiangjiang Gao
Gaojiang0272@163.com

Jinhui Lan
lanjh@ustb.edu.cn

Bingxu Wang
wbx1701@163.com

Feifan Li
g20208658@xs.ustb.edu.cn

[1] School of Automation and Electrical Engineering, University of Science and Technology Beijing, 30 Xueyuan Road, Haidian District, Beijing 100083, China

1 Introduction

The 3D point cloud is a geometric description of the model composed of a series of spatial sampling points on the surface of the object model. With the rapid development of 3D sensors and laser radar, 3D data acquisition is more convenient. 3D point cloud contains rich geometric, shape and scale information, which has been widely used in automatic driving (Wu et al., 1809; Zermas et al., 2017), indoor navigation (Zhu et al., 2017), scene understanding (Saleh et al., 2018; Saleh et al., 2018) and other fields. Nevertheless, due to the disorder, unstructured, uneven density distribution and complex scene of 3D point cloud data, how to effectively and accurately classify and segment 3D data is one of the research hotspots in the field of computer vision.

Traditional convolutional neural networks have achieved great success in image data processing tasks. However, CNNS relies heavily on data with a standard grid structure and has low processing performance for irregular and disordered point clouds. In order to make full use of the advantages of CNNS, scholars have proposed several methods (Maturana & Scherer, 2015; Riegler et al., 2017; Su et al., 2015) to map unstructured point cloud data to standard three-dimensional grid data. However, these methods are prone to information loss and complicated calculations. In 2016, the PointNet (Qi et al., 2017) network model made breakthroughs in point cloud classification, component segmentation, and scene semantic analysis, making the use of deep learning to directly process the original point cloud has become popular and gradually dominates. PointNet++(Qi et al., 2017) expanded the PointNet network model and realized the local division and local feature extraction of point cloud by constructing hierarchical structure, which solved the problems of non-uniform sampling and local feature extraction to a certain extent. These methods have achieved good results to a certain extent.

However, the feature extraction operations of shared multi-layer perceptron (MLP) and max-pooling in PointNet++have nothing to do with the spatial structure of the local area. To solve those problems, LSANet (Chen et al., 1905) adaptively generates attention maps according to the spatial distribution of the local area. The feature learning process integrated with these attention maps can effectively capture the local geometric structure. On this basis, a spatial feature extractor (SFE) is proposed, and a branch architecture is constructed to better aggregate the spatial information with relevant features in each layer of the network. But only considering local features and ignoring the importance of local features is one of the important reasons that limit the accuracy of 3D point cloud classification and segmentation. To solve this problem, we should pay more attention to those features that are more decision-making in point cloud processing tasks.

Due to the bottleneck of information processing, humans will selectively focus on part of all information and allocate limited information processing resources to important parts. The attention mechanism in deep learning is essentially similar to the selective visual attention mechanism of human beings, and has been widely used in object detection (Paigwar et al., 2019), semantic segmentation (Hu et al., 2020), biomedical image enhancement (Xiaobin et al., 2006), and three-dimensional reconstruction (Yang et al., 1808). Inspired by the attention mechanism, we mainly focus on the local features that contribute greatly to the final decision in the tasks of 3D point cloud classification and component segmentation. In this article, we propose the SDA module to obtain the attention degree of the local area of space, and make full use of the spatial relationship of the points, so as to obtain the attention feature of the local point cloud. In order to ensure the completeness of feature extraction, we extract global features of

point cloud data and fuse local attention features with global features to achieve optimal classification and segmentation accuracy. We verified the effectiveness of the proposed network through four experiments, including classification, part segmentation, scene semantic segmentation and complexity analysis and ablation.

The main contributions of our work are as follows:

- We propose a local spatial depth attention mechanism, allowing SDA layer to calculate attention coefficient by considering local neighborhood correlation and local projection depth.
- We fused the local features of depth attention with global features to obtain sufficient feature extraction capabilities.
- Our proposed SDANet has good robustness. By using uncomplete point cloud data for testing, the method in this article performs better than other similar algorithms.

The rest of this article is organized as follows. The second section reviews the related work of 3D point cloud classification and segmentation. The third part describes the proposed local depth attention mechanism and the network structure based on SDA module. The fourth part gives the experimental results and discussion. The fifth part is the conclusion.

2 Related work

2.1 Multi-view point cloud feature learning

As early as 1995, researchers in the field of visual recognition used a large number of 2D images to automatically represent poses and lighting parameters to obtain low-dimensional subspaces. In geographic information science, there is also the practice of combining data provided by airborne laser scanning with existing 2D floor plans of buildings to achieve automatic 3D data capture. Multi-view approach converts the 3D point sets to a collection of 2D views so that the popular 2D convolutional operations can be applied on the converted data. Multi-view convolutional neural network (MVCNN) (Su et al., 2015) was proposed for the first time. By capturing 2D images from multiple perspectives, the convolutional layer and pooling layer were aggregated into 3D shape descriptors, and then the aggregated features were input into the network to return the classification or segmentation results. MVCNN has shown good results in segmentation and classification tasks, and the computational efficiency has been improved, but the position of the viewpoint is set in advance to make it impossible to dynamically select views. At the same time, because a large amount of key geometric spatial information is ignored, the accuracy of MVCNN segmentation and classification is also affected, and it is not suitable for large-scale complex scenes. Some improvement methods GVCNN (Feng et al., 2018), SnapNet (Boulch et al., 2017), MHBN (Tan et al., 2018), 3D-MiniNet (Alonso et al., 2020), although enhance the accuracy of point cloud segmentation and classification tasks in some degree, 2D projection is limited to the surface modeling of the object, unable to capture the internal structure of 3D, resulting in information loss.

2.2 Voxel-based point cloud feature learning

Voxelization is an intuitive method that converts unstructured, sparse point clouds into standard neural network processing. VoxNet (Maturana & Scherer, 2015) was first proposed in 2015, by voxelizing unstructured point cloud data into grid data and apply it to a 3D deep learning network. This method constructs multiple 3D grids, normalizes the corresponding voxels in the grid, and then enters the network convolutional layer extracts features and performs maximum pooling processing on non-overlapping voxels. VoxNet solves the problem of unstructured point cloud to a certain extent, but still has the problem of occupying a large amount of memory during calculation. As the resolution increases, the number of squares increases. In order to solve the problem of voxelized grid occupying large memory and complex training, FPNN (Li & Pirk, 1605) represented 3D space as 3D vector field as network input, and used field detection filter to extract effective features. PointGrid (Le et al., 2018) uses a simple point quantization strategy to sample a fixed number of points in each grid cell, enabling the network to extract local geometric features. The voxel-based method solves the unstructured problem of 3D point clouds, but low-resolution voxels lead to the loss of useful information, and high-resolution voxels lead to large and complex computation.

2.3 Learning features from unstructured point cloud directly

In order to reduce computational complexity and make full use of 3D point cloud data, Qi et al. pioneered PointNet (Qi et al., 2017), directly input the original point cloud data, normalize the data through the T-Net network, and then use MLP to learn each point In addition, the maximum pooling layer is used to aggregate global features, which solves the problems of point cloud disorder, displacement and rotation invariance at a low cost. PointNet++(Qi, Li, et al., 2017) based on PointNet expansion, by building a local layering module, local features are captured along a multi-resolution hierarchical structure, which solves the problems of non-uniform sampling and local feature extraction. The PointSIFT module (Jiang et al., 1807), which can be embedded in various PointNet networks, uses directional coding convolution (OEC) to integrate information from eight directions to obtain a representation of coded orientation information. RandLA-Net (Hu et al., 2020) designed a local feature aggregation module to effectively learn complex local structures by gradually increasing the size of the receptive field of each neural layer. First introduce local spatial coding (LocSE) units for each 3D point to retain local geometric structure information, and then use the attention mechanism to aggregate useful local features, and increase the effectiveness of each point by stacking multiple LocSE units and pooling layers Feel the wild. This method improves the efficiency of calculation and storage while showing a good classification effect. RD3D (Qiang et al., 2101) proposed a RGB-D SOD framework, which is based on 3D CNNs and conducts cross-modal feature fusion in a progressive manner. RD3D first utilizes 3Dconvolutions for pre-fusion between RGB and depth, and then conduct explicit fusion of modality-aware features by a 3D decoder augmented with rich back-projection paths and channel-modality attention modules. LAM-Net (Cui et al., 2020), EPC-Net (Hui et al., 2101) by designing a lightweight network model, calculation and storage efficiency are improved while ensuring accuracy. These lightweight modules can be easily embedded in other deep learning networks.

2.4 Learning features based on graph theory

Since the graph neural network (GNN) (Scarselli et al., 2009) was proposed, the graph convolutional neural network (GCNN) (Kip & Welling, 1609) has performed well in semi-supervised classification tasks. Some studies SpecGCN (Wang et al., 1806), RGCNN (Gusi et al., 1803), DGCNN (Wang et al., 2019), LDGCN (Zhang et al., 1904), PointN-GCNN (Lu et al., 2020), have successfully applied graph neural networks to the task of point cloud classification and segmentation. KC-Net (Shen et al., 2018) contains a KNN graph to extract the local structural feature of the point clouds and aggregates the neighbor information through grapy max pooling. LKPO-GNN (Zhang et al., 2020) converts 3D unordered points into 1D ordered sequences. In the LKPO-GNN module, the omnidirectional local k-NNs pattern graph is used to represent and learn the rich local topological structure of the point cloud, and then the ball query module (Ball Query) is used to extract the points rich feature information. The LKPO-GNN module and the Ball Query module are alternately used to simplify the central point in the 3D point cloud scene and enrich the characteristic information of the central point aggregation, which has a good ability to express spatial geometric information.

2.5 Learning features based on attention mechanism

The attention mechanism allows the processing of variable-sized inputs, focusing on the most relevant parts of the input to make decisions. Some studies A-SCN (Liu, 2018), PAN(Li et al., 1805), GACNet (Wang et al., 2019) assign appropriate attention weight coefficients through the relationship between neighbor points. LSANet (Chen et al., 1905) uses the hierarchical local spatial feature extractor (SFE) to abstract the input point cloud to obtain high-dimensional spatial information, and generates the spatial distribution weight (SDW) hierarchically according to the spatial relationship of the local neighborhood, which has a more powerful Spatial information extraction function. Other methods PryramNet (Zhiheng & Ning, 1906), LAENet (Feng et al., 1909) embed the graph attention mechanism in the multi-layer perceptron (MLP) to learn local geometric representations. GAPNet (Chen et al., 2021) learns local geometric features by embedding the graph attention mechanism into the stacked MLP. GAPNet is proposed, which assigns different attention weights to points in the neighborhood to learn point features, and then introduces a multi-headed attention (MHA) mechanism to aggregate different GAPLayer Output features, thereby obtaining multi-attention features and multi-image features. It not only pays attention to the channel relationship, but also guarantees the quality of spatial coding to a certain extent, making the point cloud learning more adequate. GATs (Veličković & Cucurull, 1710) operates on graph-structured data, leveraging masked self-attentional layers. By stacking layers enable nodes to pay attention to the characteristics of their neighborhoods, thereby (implicitly) assigning different weights to different nodes in the neighborhood, the key to this method is the need for a reasonable central node (as shown in Fig. 2b). Because the point cloud data is relatively complex, relying only on the central node will cause insufficient information utilization, and the error caused by the central node will directly affect the final feature, so we need to consider the relationship between the point and its neighbor nodes at the same time (as shown in Fig (c)).

3 Our method

First, we introduce the description of the local depth features of point clouds. Then describe the local features of depth attention based on spatial point clouds in detail. We elaborate on the integration of global features with other operations and introduce our SDANet finally.

3.1 Spatial depth attention feature

Suppose the point cloud P is composed of N points $\{p_1, p_2, \cdots p_N\}$, the neighborhood of any point p_i is defined as a sphere with p_i as the center of the sphere and r as the radius, denoted as $p_n^j = \{p_i^j | j = 1, 2, \cdots, k\}$, where k is the number of points in the local area. Assume the sampling surface of the point cloud is smooth everywhere and have access to the whole 3D surface. Then the local neighborhood of the point cloud can be fitted with a surface, so that the normal vector n_i on the surface where the point p_i is located can be obtained. In the same way, n_i^j can be obtained, then take n_i as the local coordinate axis. Along the positive direction of the local coordinate axis, define the two-dimensional plane E as the projection plane, as shown in Fig. 1. The point set $p_n^{'i} = \{p_i^{'j} | j = 1, 2, \cdots, k\}$ is obtained by projecting all points in the neighborhood of p_i onto E. Then the distance between p_i^j and $p_i^{'j}$ is the local depth, and the value range is [0,2r].

The specific expression of local depth is

$$d_j = r - n_i \cdot \left(p_i^j - p_i\right) \tag{1}$$

Let $S = \left\{X_i \in \mathbb{R}^K | i = 1, 2, \ldots, N\right\}$ be a local area point set, where X_i is the point with K-dimensional attributes in the local point set S, N is the total number of points in the local area. The local feature of depth attention consists of two parts, one is the spatial feature of the local area where the point is located, and the other is the spatial feature of the point itself. Taking into account the different contribution of each local point cloud feature to the final decision (for example, it is easier to classify the target as a vehicle by using the wheel feature). In this article, a high degree of emphasis is given to the local point cloud features with obvious changes in shape. The size of the original point cloud data is $N \times K$. After sampling and grouping, the local point cloud $G = (L, K)$ is obtained, where L is the number of points and K is the dimension of each point. Establish a three-dimensional space coordinate system with centroid $G_o = (g_{ox}, g_{oy}, g_{oz})$ as the origin (as shown in Fig. 4). Project to $N_x - N_y$, $N_y - N_z$, $N_z - N_x$ planes respectively, find the closest point P_N and the farthest point P_L, expressed as follows:

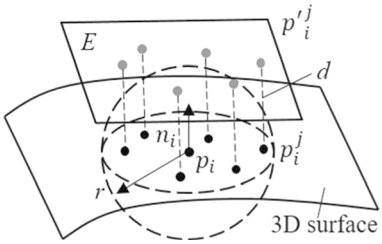

Fig. 1 Schematic diagram of the local depth of the 3D point cloud

$$P_N = \underset{P}{argmax}\left\{\underset{P\in L_{xy}}{max} \parallel P_{xy} - G_{oxy} \parallel_2, \underset{P\in L_{yz}}{max} \parallel P_{yz} - G_{oyz} \parallel_2, \underset{P\in L_{xy}}{max} \parallel P_{zx} - G_{ozx} \parallel_2\right\} \quad (2)$$

$$P_L = \underset{P}{argmin}\left\{\underset{P\in L_{xy}}{min} \parallel P_{xy} - G_{oxy} \parallel_2, \underset{P\in L_{yz}}{min} \parallel P_{yz} - G_{oyz} \parallel_2, \underset{P\in L_{xy}}{min} \parallel P_{zx} - G_{ozx} \parallel_2\right\} \quad (3)$$

Put P_N and P_L into the local depth formula (1) to get the local feature attention:

$$d_{fc} = |d_{P_N} - d_{P_L}| \quad (4)$$

Then the spatial feature of the local area is encoded and expressed as:

$$F^g = d_{fc} \cdot \frac{1}{N}\sum_{i}^{N} W_1 X_i \quad (5)$$

where $W_1 \in \mathbb{R}^{C\times K}$, F^g is the feature obtained by encoding all local points.

The spatial characteristics of points can be expressed as:

$$F_i^p = W_0 X_i \quad (6)$$

where $W_0 \in \mathbb{R}^{C\times K}$,$F_i^p \in \mathbb{R}^C$, which is the spatial feature of point itself. Since each point in the point cloud does not exist in isolation, it has a spatial relationship with surrounding neighbor points as shown in Fig. 2 (c), the point spatial feature after correction is expressed as:

$$\hat{F}_i^p = W_0 \hat{X}_i \quad (7)$$

where

$$\hat{X}_i = X_i^k - \rho \sum_{j\in N_i} SGN\left(X_i - X_j\right) - y_{ij} N_i^k \quad (8)$$

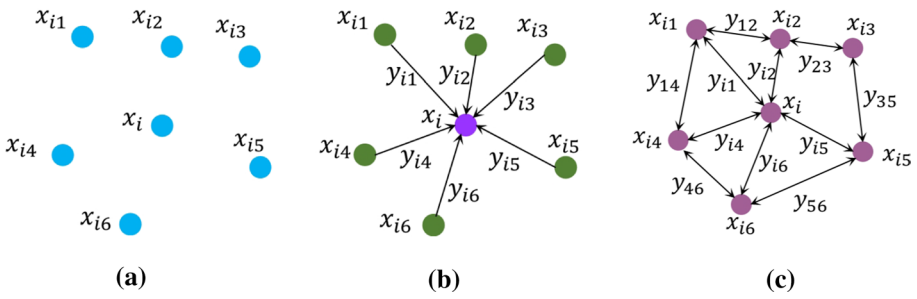

Fig. 2 **a**, **b**, **c** are three identical local regions. **a** shows that the weight of each point is fixed and has nothing to do with the spatial relationship. **b** shows that the weight of each point is related to the center point of the local area. **c** shows that the weight of each point is related to all neighbor points in the local area. Among them, x_i represents the center point of the local area, and y_{ij} represents the spatial relationship between neighboring nodes

$SGN(.)$ is the sign function 0–1. $y_{ij} = x_{ij_1} - x_{ij_2}$ is the boundary coefficient(as shown in Fig. 2c), and $y_{ij} \in [0, 2r]$. N_i^k is the normal vector direction of the plane where the current point is located. This method only needs to use the partial order information of the state of the neighboring points in the local point cloud (information that is only valid for some individuals in the network) and does not require accurate relative state and total order information (information that is valid for all individuals in the network).

The final local feature of depth attention is obtained by combining the spatial feature of point and local region:

$$\hat{F}_i = \left[\hat{F}_i^p, F^g\right] \tag{9}$$

where $[,]$ denotes the concatenation operation, $\widehat{F}_i \in \mathbb{R}^{2C}$. The local features of depth attention are not only related to the spatial position of each point itself, but also related to the local overall information. Different points in the same local area have different $\widehat{F}_i^p$, but have the same $\widehat{F}^g$. In order to achieve accurate classification and segmentation of point cloud data, it is not enough to use local features alone. For this reason, we will construct a fusion feature of deep attention local features and global features as an effective feature for point cloud classification and segmentation.

3.2 Feature aggregation for point set learning

The point cloud global feature extraction module is an important part of the network structure. After the input point cloud is subjected to Spatial transform, it is directly operated on the whole to obtain the global feature. The point cloud global feature extraction network draws on the main ideas of the PointNet network structure and optimizes it, such as increasing the number of MLP layers to upgrade the input point cloud to higher-dimensional features, and designing the convolution kernel size to increase the convolution process feel the wild. The input is each point of the point cloud data, including spatial coordinate information (or normal vector, color information, etc.). Through the multi-layer perceptron MLP, the input points are upgraded to high-dimensional features. The high-dimensional features are mapped by the maximum symmetric function and the nonlinear activation function, and the global features of the input point cloud are extracted finally.

In the design of the SDANet network structure, the depth attention local feature of the input point cloud and the global feature extraction are paralleled. The two processes operate separately and do not affect each other. In order to improve the effect of point cloud classification and segmentation, we added a local feature and global feature fusion module to the network structure design. This module uses the concat operation to fuse local features and global features.

The concat connects multiple tensor channel inputs and serves as the input of the next layer of the network. The concat operation expression is

$$concat(A, B, axis) \tag{10}$$

where, A is the local feature tensor of deep attention, B represents the high-dimensional feature tensor obtained by the global feature extraction module, $axis$ is the number of dimensions when fusion is performed.

3.3 Network architecture

The SDANet architecture is shown in Fig. 3. It mainly includes point cloud alignment transformation, SDA module extracting local depth attention features, global feature extraction, and local and global feature fusion. The original point cloud data (N × K) is standardized by the spatial transform module to realize the normalization of the original point cloud with different rotation and translation. The normalized point cloud data is subjected to two parallel processes of deep attention local feature and global feature extraction to achieve feature extraction. Finally, use the concat module to fuse the local features and global features, and then perform point cloud processing tasks according to different needs.

When using CNN to classify traditional images, it is usually necessary to consider the locality of the input sample, translation invariance, reduction invariance, rotation invariance, etc., to improve the accuracy of classification. Similarly, when processing point cloud data, it is also necessary to ensure the invariance of input samples. A more effective method is to introduce a spatial transformation network between certain two layers of the neural network. The spatial transformation network includes two parts: The first part is localization net, which used to generate affine transformation system structure for design. And the parameters in the localization net are the parameters that the spatial transformation network needs to train; The second part is spatial transformation, which is affine transformation. The affine transformation coefficient θ is generated through the local network (It can also be other types of spatial transformations, and local networks can be designed as needed to obtain the corresponding spatial transformation coefficients θ), After obtaining the affine transformation coefficient θ, we can perform affine transformation on the input of the previous layer and input the affine transformation result to the next layer.

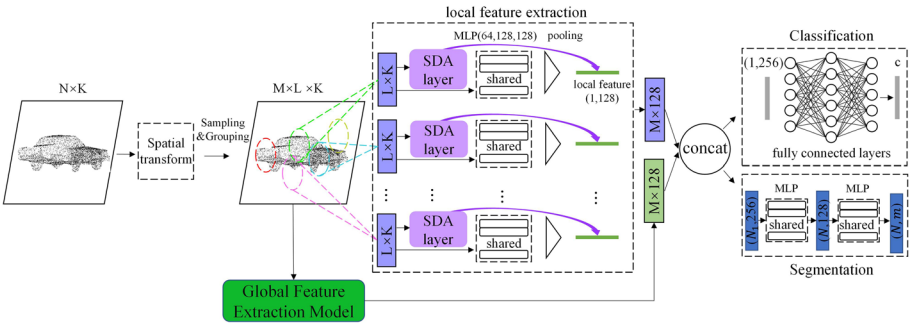

Fig. 3 SDANet architecture. The SDA layer is shown in Fig. 4. The Global Feature Extraction Module is shown in Fig. 5. The architecture contains two parts: classification (top branch) and segmentation (bottom branch). The classification model takes N points as input normalizes the data through the spatial transform model, followed by SDA layer and MLP to obtain local depth attention features, then merges with global features and inputs them to the fully connected layers to obtain classification score for category c finally. The feature extraction process of the segmentation model is consistent with the classification model. After the effective features of the point cloud are obtained, the category score of each point is output after multiple MLP layers

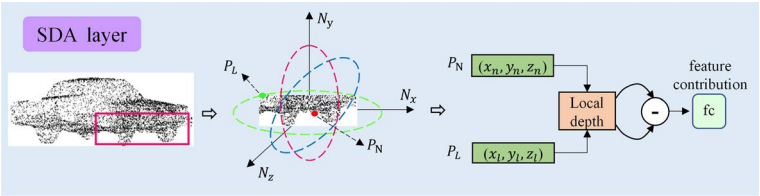

Fig. 4 Local point cloud spatial depth attention layer. SDA layer is mainly used to obtain the attention coefficient of the local feature of the point cloud. For the sampled and grouped point cloud data, a three-dimensional space coordinate system is established with the centroid, then all the point clouds in the local area are respectively projected to the three projection surfaces to obtain the key points of P_N and P_L. Code the two points to get local feature attention finally

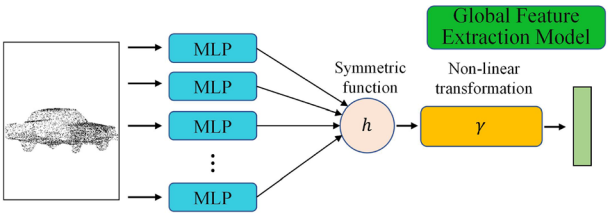

Fig. 5 Global Feature Extraction Model. This module is mainly used to obtain the global characteristics of the point cloud. The multi-layer perceptron MLP promotes the input points to high-dimensional features. After the high-dimensional features are multiplied by the maximum symmetric function and the nonlinear activation function mapping, the global features of the input point cloud are extracted

$$\begin{pmatrix} x^\tau \\ y^\tau \\ z^\tau \end{pmatrix} = A_\theta \begin{pmatrix} x^\sigma \\ y^\sigma \\ z^\sigma \\ 1 \end{pmatrix} = \begin{pmatrix} \theta_{11} & \theta_{12} & \theta_{13} & \theta_{14} \\ \theta_{21} & \theta_{22} & \theta_{23} & \theta_{24} \\ \theta_{31} & \theta_{32} & \theta_{33} & \theta_{34} \end{pmatrix} \begin{pmatrix} x^\sigma \\ y^\sigma \\ z^\sigma \\ 1 \end{pmatrix} \quad (11)$$

.Where, (x^τ, y^τ, z^τ) represents the input point coordinates, $(x^\sigma, y^\sigma, z^\sigma)$ represents the output coordinate point. The transformation matrix parameter θ represents the output of the local network.

4 Experiments

Through a large number of experiments, we tested and analyzed the object classification, component segmentation and scene segmentation tasks of the SDANet network model proposed in this article, then we compared and analyzed it with the latest methods (Figs. 4, 5, 6 and 7).

4.1 Classification

4.1.1 Data set

The data set is ModelNet40 (Wu et al., 2015) and Part- ModelNet40 (cut and delete data for each sample in the ModelNet40 data set, and retain half of the data information), ModelNet40 has 40 types of samples, the training set and the test set have 9943 and 2468

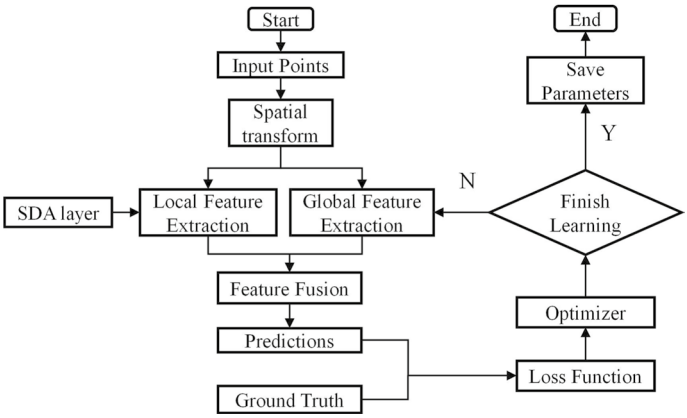

Fig. 6 Training algorithm of classification and segmentation networks

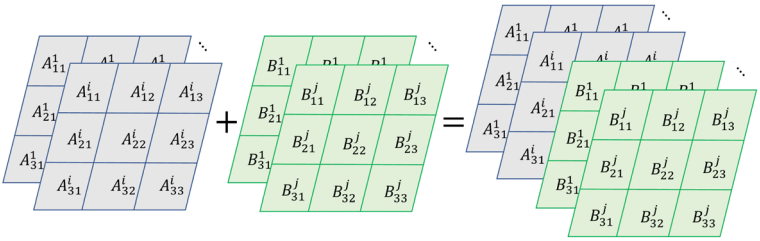

Fig. 7 Fusion of the depth attention local feature and global feature. **a** the depth attention local feature. **b** global feature. **c** fusion feature

samples respectively; Part- ModelNet40 sample category, training set and test set number are the same as ModelNet40, but each sample information rate is only half. The data set visualization is shown in Fig. 8. 1024 points were sampled from each 3D model for feature extraction and model training.

4.1.2 Training

Unless otherwise specified, the experiments in this article are all carried out in the Pytorch environment. We use the Adam optimization algorithm with an initial learning rate of 0.001, the decay ratio is 0.8 applied every 40 epochs. Set the batch size of the classification network to 24. Using 1 NVIDIA GTX1080Ti GPU, the number of training is 200 epochs.

4.1.3 Results

In order to compare classification accuracy and computational complexity, we use indicators such as overall accuracy and forward times to verify the performance of the algorithm. Table 1 compares our results and complexity with some recent new methods. Although PointNet achieves the best computational complexity, our model accuracy is 3.5% higher than it. At the same time, the accuracy of LSA has reached the highest level of 93.2%,

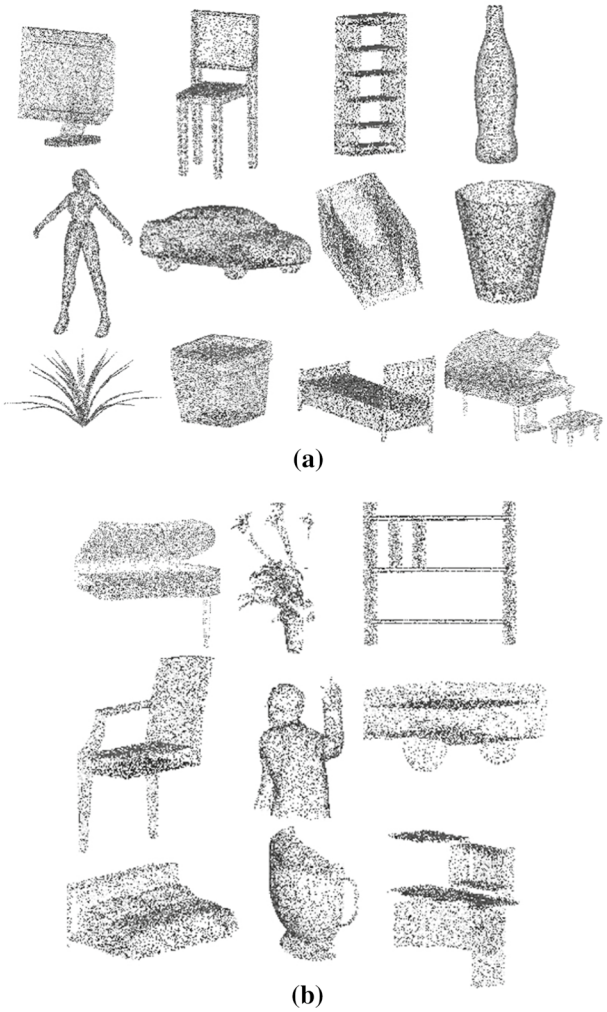

Fig. 8 Visualization of the data set used in the experiment. **a** ModelNet40 data set. **b** Part-ModelNet40 data set

Table 1 Classification results on ModelNet40 dataset

Method	MA (%)	OA (%)	Forward time (ms)
PointNet (Qi et al., 2017)	86.2	89.2	**15.2**
PointNet++(Qi et al., 2017)	–	91.9	35.3
PointNGCNN (Lu et al., 2020)	–	92.8	
GAPNet (Chen et al., 2021)	89.7	92.4	27.9
LSANet (Chen et al., 1905)	90.3	93.2	60.0
LAM-PointNet++(Cui et al., 2020)	–	91.3	33.6
PCT (MengHao et al., 2012)	–	93.2	72.4
OURS	**91.5**	**93.7**	40.3

The bold indicates that the method achieves the best performance in this metric compared to other methods

Table 2 Classification results on Part-ModelNet40 dataset

Method	MA (%)	OA (%)	Forward time (ms)
PointNet (Qi et al., 2017)	62.5	69.8	**14.8**
PointNet++(Qi et al., 2017)		72.3	35.0
PCT (MengHao et al., 2012)		77.4	70.3
OURS	**79.6**	**83.5**	38.6

The bold indicates that the method achieves the best performance in this metric compared to other methods

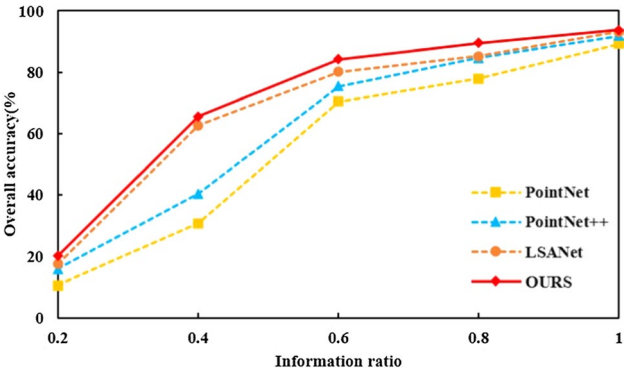

Fig. 9 Classification of point clouds: overall accuracy with the decreasing of information. Information ratio reflects the completeness of information

and the time complexity of our model is 32.8% lower than it. This means that our model achieves the best balance between accuracy and complexity.

Table 2 shows the classification accuracy results of various methods when the input data information is halved. The results show that the classification method has incomplete encoding of shape features due to the halving of the amount of input data information, which generally reduces the overall accuracy. Compared with Table 1, the reason why the forward time of the network has no obvious difference is that although the amount of information of the input information is halved, the number of points input to the neural network for processing after subsampling does not change, so that the forward time does not fluctuate significantly. Our method pays more attention to the descriptive features of the category. For example, it is easier to identify the target as a vehicle based on the characteristics of the wheel. Although the accuracy is reduced, it still has better performance than other methods.

We did another set of experiments to test the overall accuracy of the algorithm by changing the point clouds information ratio, that is, compared with the original data, the amount of information in the existing data accounts for. Figure 9 shows the results of the experiment. The results show that as the input information ratio decreases, the classification accuracy generally shows a downward trend. This is because when the effective information of the input data is reduced, the shape features will be much reduced. It is extremely difficult to use a greatly small amount of data features to classify correctly. However, our method always maintains superior performance under different information ratios. This

proves that the SDANet model has strong adaptability to the classification of incomplete information objects.

4.2 Part segmentation

4.2.1 Data set

In the Part Segmentation task, we use the ShapeNet (Yi et al., 2016) data set. ShapeNet contains 16,881 3D models in 16 shape categories and 50 different parts. Besides, each shape model is labeled with several but less than 6 parts. We selected 9943 models for training, 2468 models for testing. 2048 points were sampled from each 3D model, and each point was associated with a part label (Fig. 10).

4.2.2 Training

We use the Adam optimization algorithm with an initial learning rate of 0.001. The decay ratio is 0.8 applied every 40 epochs. Batch size is set to 10. Using 1 NVIDIA GTX1080Ti GPU. The number of part segmentation model training is 250 epochs.

4.2.3 Results

In order to verify the accuracy of segmentation, we use the mean Intersection over Union (mIoU) as an evaluation index to verify the performance of the algorithm. Figure 11 shows some visualized results of the output of part segmentation. According to the visualization results, for the cup, we classify the grip and the cup body well, the grip part is blue, and the cup body part is red. For hats, we classify the brim and the top of the hat very well. The top of the hat is divided into blue, and the brim is red. For knives, we classify the handle and the blade well. The blade part is blue and the handle part is red. For cars, we correctly classify the roof, body, tires and hood. The top of the car is divided into blue, the body part is red, the tires are yellow, and the hood is green. For motorcycles, we clearly classify the wheels, body, and fuel tanks. The wheels are green, the body is red, and the fuel tank is blue. For the table, we correctly classify the table legs and the table board, the table leg part is red, and the table board part is blue. Table 3 shows the results of our method on the ShapeNet dataset, for some reason, we didn't show the airplane result data. Our model

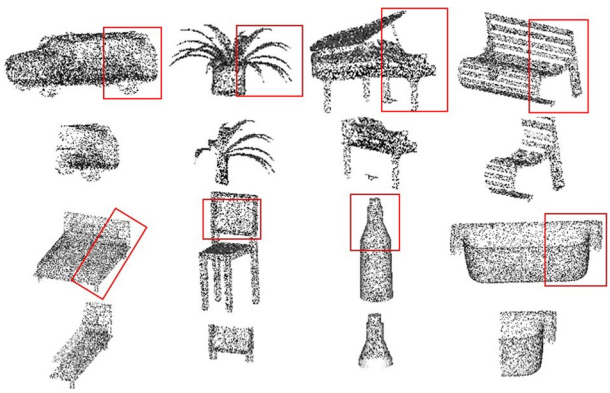

Fig. 10 Visualization of changing the point clouds information ratio

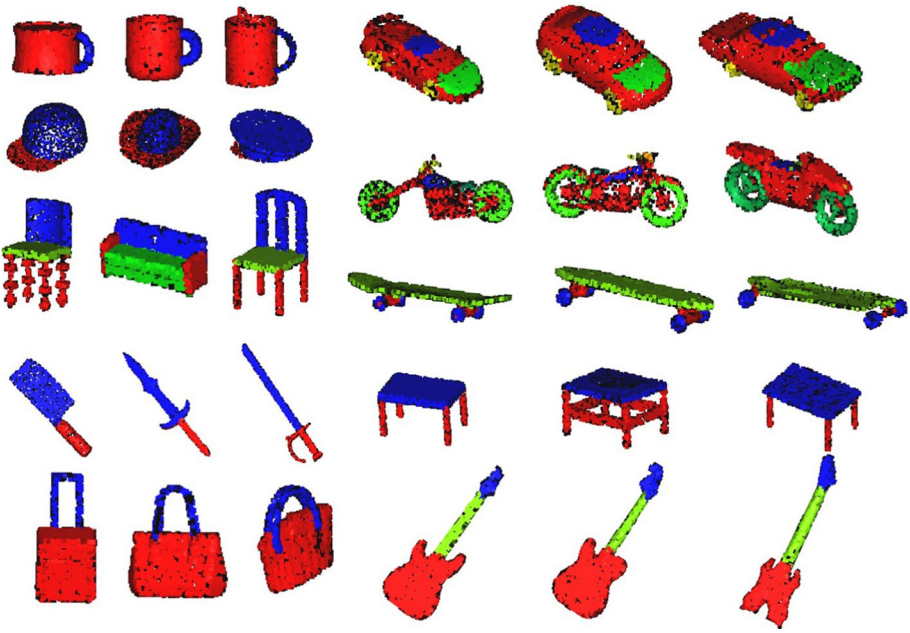

Fig. 11 Visualization results of part segmentation on ShapeNet dataset. We randomly select 8 types of items as the result visualization display, which are Mug, Cap, Chair, Knife, Car, Motorbike, Skateboard, Table. Each type of item itself is classified according to different parts

wins 4 categories for part segmentation compared with 1 winning categories from LAM-Pointnet++, although it is the same accuracy as ours. Although the PCT (MengHao et al., 2012) performed an excellent level, the algorithm in this paper is only 1.1% smaller than that, and our method is still relatively good in classification tasks.

4.3 Scene segmentation

4.3.1 Data set

In the Scene Segmentation task, we choose Stanford Large-Scale 3D Indoor Space data set (S3DIS) (Armeni et al., 2016) to test the performance of our model for indoor scene segmentation. S3DIS contains 6 3D point clouds of indoor scenes, covering a total of 272 rooms. We select areas1-5 as the training set and area6 as the test set. Sample 4096 points from each scene, each point contains XYZ, RGB and normalized location as to the room information.

4.3.2 Training

We use the Adam optimization algorithm with an initial learning rate of 0.001. The epoch number is 64. The decay ratio is 0.7 applied every 8 epochs. batch size is set to 4. And we distribute the task to two NVIDIA GTX1080Ti GPU.

Table 3 Segmentation results on the ShapeNet dataset

Method	IoU	Air-plane	Bag	Cap	Car	Chair	Ear phone	Guit	Knif	Lamp	Lapt	Motor	Mug	Pistol	Rocket	Skate board	Table
PointNet (Qi et al., 2017)	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
PointNet++(Qi et al., 2017)	85.1	82.4	79.0	87.7	77.3	90.8	71.8	91.0	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6
LAM-Point++(Cui et al., 2020)	85.3	83.0	79.2	87.5	78.4	90.9	70.7	91.3	88.1	84.0	95.3	71.9	94.3	81.9	58.9	76.7	82.8
DGCNN (Wang et al., 2019)	85.1	84.2	83.7	84.4	77.1	90.9	78.5	91.5	87.3	82.9	96.0	67.8	93.3	82.6	59.7	75.5	82.0
SAWNet (Kaul et al., 1905)	84.8	82.0	85.5	88.7	78.0	90.9	77.0	91.0	88.7	82.5	95.5	63.6	94.2	77.6	57.0	74.8	81.9
PCT (MengHao et al., 2012)	86.4	85.0	82.4	89.0	81.2	91.9	71.5	91.3	88.1	86.3	95.8	64.6	95.8	83.6	62.2	77.6	83.7
OURS	85.3	–	82.9	84.8	79.0	90.7	68.6	91.0	87.1	82.8	95.6	73.0	95.6	82.5	61.2	76.2	83.1

4.3.3 Results

In order to verify the effect of scene segmentation, we selected reasonable evaluation indicators. The IoU of each shape is calculated by averaging IoUs for all parts that fall into the same category, then the mIoU is the mean IoUs for all shapes from testing dataset. Figure 12 shows the visualization results of Scene segmentation on the S3DIS data set. Compared with ground truth, there are many differences between the results obtained by our method and PointNet. For conference room, Point incorrectly classifies the door as a wall and the wooden board as a ceiling. In addition, for the lobby, point mistakenly classified part of the trash can as chairs and mistakenly classified wooden boards as tables. According to the comparison of the results, the segmentation effect of our model at some key positions is closer to Ground truth, which intuitively shows that our method helps to improve the accuracy of point cloud scene segmentation. Table 4 shows the results of our model and PointNet on the S3DIS dataset. The results show that our model accuracy is 5.7% better than PointNet.

4.4 Complexity analysis and ablation experiments

We further compare both space and time complexities with other methods, in which the classification network is used. Table 5 shows that our SDANet has proper parameters with

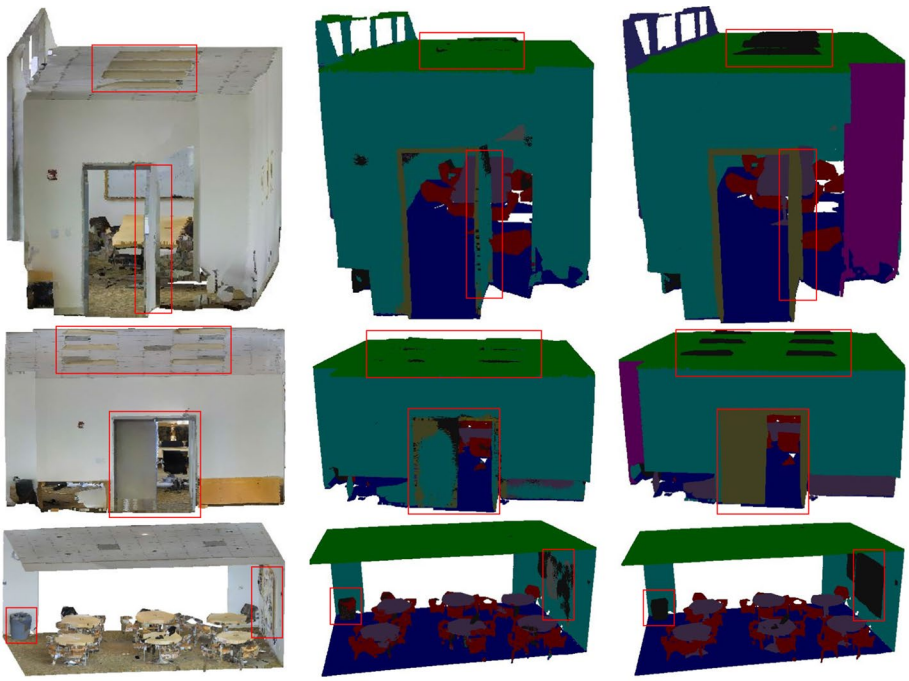

Fig. 12 Scene segmentation on the S3DIS data set. **a** Ground truth. **b** Results of PointNet. **c** Results of OURS. We compared our method and PointNet in the results with ground truth, and the obvious differences are marked with red borders

Table 4 Scene segmentation results on the S3dis dataset

Method	mIoU (%)	Ceiling	Floor	Wall	Beam	Column	Window	Door	Table	Chair	Sofa	Book-case	Board	Clutter
PointNet (Qi et al., 2017)	53.9	89.3	97.0	72.4	49.2	20.8	38.7	30.4	68.0	61.9	26.5	**60.7**	48.1	38.2
OURS	**59.6**	**91.6**	**97.3**	**73.0**	**51.6**	**25.8**	**52.3**	**56.1**	**69.6**	**63.0**	**39.2**	56.4	**48.1**	**50.6**

The bold indicates that the method achieves the best performance in this metric compared to other methods

Table 5 Comparison of different methods on the numbers of parameters and inference time

Method	Parameters (M)	Inference time(ms)
PointNet (Qi et al., 2017)	3.48	24.0
PointNet++(Qi et al., 2017)	1.48	74.2
GAPNet(Chen et al., 2021)	4.41	98.7
LSANet(Chen et al., 1905)	2.30	114.7
LAM-PointNet++(Cui et al., 2020)	1.68	72.0
PCT (MengHao et al., 2012)	2.88	93.7
OURS	2.16	64.8

fast inference time. In addition, our segmentation network involves fewer parameters than our classification network (see Table 6).

To verify the effectiveness of the spatial depth attention, global feature extraction, and spatial transformation network, we have done ablation experiments under the above conditions, and the final results are shown in Table 7. In the ablation experiment, we tested the network parameters under the condition of missing specific function modules and the final point cloud target overall accuracy. The results show that without spatial depth attention, the network model parameters have been significantly reduced, compared with SDANet by 37.5%, but the overall accuracy is also reduced by 16.4%. Without global feature extraction, the number of network model parameters decreased slightly, and the classification accuracy also decreased by 11.4%. Without the spatial transformation network, the number of network model parameters is almost unchanged, but the classification accuracy directly drops by 74.6%, which shows that the input point cloud data is not processed by the spatial transformation, and the performance of directly using the deep learning network is very poor.

5 Conclusion

In this article, we propose a neural network based on local depth attention features, called SDANet, to learn point clouds represented by shapes. Based on the new design of the network, our SDANet has powerful spatial information extraction capabilities, and has

Table 6 The numbers of our SDANet's parameters on datasets

Dataset	Task	Parameters (M)
ModelNet40 (Wu et al., 2015)	Classification	2.16
ShapeNet (Yi et al., 2016)	Segmentation	1.84
S3DIS (Armeni et al., 2016)	Segmentation	1.44

Table 7 The results of ablation experiments

	Parameters (M)	OA(%)
Remove spatial depth attention	1.35	78.3
Remove global feature extraction	1.84	83.0
Remove spatial transformation network	2.14	23.8
The full framework (SDANet)	2.16	93.7

achieved results equal to or better than the most advanced methods in shape classification, part segmentation and scene semantic segmentation tasks. We also provide visual results and detailed information of related experiments. The success of this model verifies the effectiveness of local deep attention features in point cloud classification and segmentation tasks. With the development of sensor technology, the information obtained in the field of autonomous driving will be more refined, but it will inevitably cause the amount of information to skyrocket. Therefore, how to process information quickly and effectively is one of the future research directions.

Authors' contributions JG, BW contributed to the conception of the study; JG performed the experiment; JG, JL contributed significantly to analysis and manuscript preparation; JG performed the data analyses and wrote the manuscript; JL, BW, and FL helped perform the analysis with constructive discussions.

Funding This research was funded by the 13th Five-Year Research Project (4149020102).

Availability of data and material The datasets used can be found at: ModelNet40: http://modelnet.cs.princeton.edu/, ModelNet40_resampled: https://pan.baidu.com/s/1-2Wldi-in1XbomBOraU5eg (psd: zntc), ShapeNet: https://shapenet.org/, S3DIS: http://buildingparser.stanford.edu/dataset.html

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

Alonso, I., Riazuelo, L., Montesano, L., Murillo, A. C., & Letters, A. (2020). 3D-MiniNet: Learning a 2D representation from point clouds for fast and efficient 3D LIDAR semantic segmentation. *IEEE Robotics and Automation Letters, 5*, 5432–5439.

Armeni, I, Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., & Savarese, S. (2016). 3D Semantic parsing of large-scale indoor spaces. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1534–1543.

Boulch, A., Guerry, Y., Saux, B. L., & Audebert, N. J. C., (2017) Graphics.: SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks. Computers & Graphics 71: 189–198.

Chen, L. Z., Li, X. Y., Fan, D. P., Cheng, M. M., Wang, K., & Lu, S. P. (2019). LSANet: Feature learning on point sets by local spatial attention. arXiv preprint arXiv: 1905.05442.

Chen, C., Fragonara, L. Z., & Tsourdos, A. (2021). GAPointNet: Graph attention based point neural network for exploiting local feature of point cloud. *Neurocomputing, 438*, 122–132.

Cui, Y., An, Y., Sun, W., Hu, H., & Song, X. (2020). Lightweight attention module for deep learning on classification and segmentation of 3-D point clouds. *IEEE Transactions on Instrumentation and Measurement, 70*, 99.

Feng, M., Zhang, L., Lin, X., Gilani, S. Z., & Mian, A. (2019). Point attention network for semantic segmentation of 3D point clouds. arXiv preprint arXiv: 1909.12663.

Feng, Y., Zhang, Z., Zhao, X., Ji, R., & Yue, G. (2018). GVCNN: Group-view convolutional neural networks for 3D shape recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 264–272.

Gusi, T., Wei, H., Zongming, G., & Amin Z. (2018). Local spectral graph convolution for point set feature learning. arXiv preprint arXiv: 1803.05827.

Hu, Q., Yang, B., Xie, L., Rosa, S., & Markham, A. (2020b). RandLA-Net: Efficient semantic segmentation of large-scale point clouds. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13–19.

Hu, Z., Zhang, D., Li, S., & Qin, H. J. C. (2020). Graphics.: Attention-based relation and context modeling for point cloud semantic segmentation. *Computers & Graphics, 90*, 126–134.

Hui, L., Cheng, M., Xie, J., & Yang, J. (2021). Efficient 3D point cloud feature learning for large-scale place recognition. arXiv preprint arXiv: 2101.02374.

Jiang, M., Wu, Y., Zhao, T., Zhao, Z., & Lu, C. (2018). PointSIFT: A SIFT-like network module for 3D point cloud semantic segmentation. arXiv preprint arXiv: 1807.00652.
Kaul, C., Pears, N., & Manandhar, S. (2019). SAWNet: A spatially aware deep neural network for 3D point cloud processing. arXiv preprint arXiv: 1905.07650.
Kip, F. T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv: 1609.02907 .
Le, T., & Ye, D. (2018). PointGrid: A deep network for 3D shape understanding. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 18–23.
Li, H., Xiong, P., An, J., & Wang, L. (2018). Pyramid attention network for semantic segmentation. arXiv preprint arXiv: 1805.10180.
Li, Y., & Pirk, S. (2016). FPNN: Field probing neural networks for 3D data. arXiv preprint arXiv: 1605.06240.
Liu, S. (2018). Attentional shapecontextnet for point cloud recognition. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4606–4615.
Lu, Q., Chen, C., Xie, W., & Luo, Y. J. C. (2020). Graphics.: PointNGCNN: Deep convolutional networks on 3D point clouds with neighborhood graph filters. *Computers & Graphics, 86*, 42–51.
Maturana, D., & Scherer, S. (2015). VoxNet: A 3D convolutional neural network for real-time object recognition. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 922–928.
MengHao, G., JunXiong, C., Zheng-Ning, L., Tai-Jiang, M., Martin, R. R., Hu, S. M. (2020): Pct: Point cloud transformer. arXiv preprint arXiv: 2012.09688. Doi: https://doi.org/10.1007/s41095-021-0229-5.
Paigwar, A., Erkent, O., Wolf, C., & Laugier, C. (2019). Attentional PointNet for 3D-object detection in point clouds. In IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1297–1306.
Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017a). PointNet: Deep learning on point sets for 3D classification and segmentation. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 652–660.
Qi, C. R., Li, Y., Hao, S., & Guibas, L. J. (2017b). PointNet++: Deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, pp. 5099–5108.
Qiang, C., Ze, L., Keren, F., Zhao, Q., Du, H. (2021). RGB-D salient object detection via 3D convolutional neural networks. arXiv preprint arXiv: 2101.10241v1.
Riegler, G., Ulusoy, A. O., & Geiger, A. (2017). OctNet: learning deep 3D representations at high resolutions. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3577–3586.
Saleh, K., Zeineldin, R. A., Hossny, M., Nahavandi, S., & El-Fishawy, N. (2018b). End-to-end indoor navigation assistance for the visually impaired using monocular camera. In IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 3504–3510.
Saleh, K., Attia, M., Hossny, M., Hanoun, S., & Nahavandi, S. (2018a). Local motion planning for ground mobile robots via deep imitation learning. In: IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 4077–4082.
Scarselli, F., Gori, M., Tsoi, A., Hagenbuchner, M., & Monfardini, G. J. (2009). The graph neural network model. *IEEE Transactionson Neural Networks, 20*, 61–80.
Shen, Y., Feng, C., Yang, Y., Tian, D. (2018). Mining point cloud local structures by kernel correlation and graph pooling. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4548–4557.
Su, H., Maji, S., Kalogerakis, E., & Learned-Miller, E. J. I. (2015). Multi-view convolutional neural networks for 3D shape recognition. In IEEE International Conference on Computer Vision (ICCV), pp. 945–953.
Tan, Y., Meng, J., & Yuan, J. (2018). Multi-view harmonized bilinear network for 3D object recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 18–23.
Veličković, P., Cucurull, G., Casanova, A. (2018). Graph attention networks. arXiv preprint arXiv: 1710.10903v3.
Wang, C., Samari, B., & Siddiqi, K. (2018). Regularized graph CNN for point cloud segmentation. arXiv preprint arXiv: 1806.02952.
Wang, L., Huang, Y., Hou, Y., Zhang, S., & Shan, J. (2019b). Graph attention convolution for point cloud semantic segmentation. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10288–10297.
Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M., & Solomon, J. (2019). Dynamic graph CNN for learning on point clouds. *ACM Transactions on Graphics, 38*, 1–12.

Wu, B., Zhou, X., Zhao, S., Yue, X., & Keutzer, K. (2018). SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud. arXiv preprint arXiv: 1809.08495.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., & Xiao, J. (2015). 3D ShapeNets: A deep representation for volumetric shapes. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1912–1920.

Xiaobin, H., Yanyang, Y., Wenqi, R., Hongwei, L., Yu, Z. (2020). Feedback graph attention convolutional network for medical image enhancement. In: Image and Video Processing (eess.IV). arXiv preprint arXiv: 2006.13863.

Yang, B., Wang, S., Markham, A., & Trigoni, N. (2019). Robust attentional aggregation of deep feature sets for multi-view 3D reconstruction. arXiv preprint arXiv: 1808.00758.

Yi, L., Kim, V. G., Ceylan, D., Shen, I. C., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., & Guibas, L. (2016). A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics, 35*, 1–12.

Zermas, D., Izzat, I., & Papanikolopoulos, N. (2017). Fast segmentation of 3D point clouds: A paradigm on LiDAR data for autonomous vehicle applications. In IEEE International Conference on Robotics and Automation (ICRA), pp. 5067–5073.

Zhang, K., Hao, M., Wang, J., Silva, C. D., & Fu, C. (2019). Linked dynamic graph CNN: Learning on point cloud via linking hierarchical features. arXiv preprint arXiv: 1904.10014.

Zhang, W., Su, S., Wang, B., & Sun, L. J. N. (2020). Local k-NNs pattern in omni-direction graph convolution neural network for 3D point clouds. *Neurocomputing, 413*, 487–498.

Zhiheng, K., & Ning, L. (2019). PyramNet: Point cloud pyramid attention network and graph embedding module for classification and segmentation. arXiv preprint arXiv: 1906.03299.

Zhu, Y., Mottaghi, R., Kolve, E., Lim, J. J., & Farhadi, A. (2017). Target-driven visual navigation in indoor scenes using deep reinforcement learning. In IEEE International Conference on Robotics and Automation (ICRA), pp. 3357–3364.