ISPRS Journal of Photogrammetry and Remote Sensing 182 (2021) 37–51

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

ResDLPS-Net: Joint residual-dense optimization for large-scale point cloud semantic segmentation

Jing Du [a], Guorong Cai [a,*], Zongyue Wang [a], Shangfeng Huang [a], Jinhe Su [a],
José Marcato Junior [b], Julian Smit [c], Jonathan Li [d,*]

[a] School of Computer Engineering, Jimei University, Xiamen 361021, China
[b] Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Campo Grande 79070900, MS, Brazil
[c] School of Architecture, Planning and Geomatics Faculty of Engineering and the Built Environment, University of Cape Town, Cape, South Africa
[d] Departments of Geography and Environmental Management and Systems Design Engineering, University of Waterloo, 200 University Avenue West, Waterloo, ON N2L 3G1, Canada

ARTICLE INFO

Keywords:
Large-scale point clouds
Semantic segmentation
Joint residual-dense optimization
Deep learning

ABSTRACT

Semantic segmentation methods based on three-dimensional (3D) point clouds are mostly limited to input point clouds that have been divided into blocks for training. This is mainly attributed to the requirement of constant trade-offs between computational resources and accuracy for directly processing large-scale point clouds. Specifically, the block dividing strategy will add the data preprocessing time to some extent and may disturb the complete geometry of the object. Therefore, this paper proposes a large-scale point cloud semantic segmentation network without block dividing operation, referred to as ResDLPS-Net. This network can take the complete point cloud of the whole large scene as input and process up to nearly a million points on one single GPU. In particular, a novel feature extraction module is designed to efficiently extract neighbor, geometric, and semantic features. The learned features are then aggregated through the attention mechanism to form local feature descriptors. In addition, the proposed ResDLPS-Net is jointly trained by residual connections and dense convolutional connections to optimize the feature aggregation operation. As a result, the ResDLPS-Net performs brilliantly on multiple objects, such as windows, road markings, fences, etc. For example, the Mean Intersection over Union (mIoU) of road markings on the Toronto-3D dataset is 37.76% higher than the state-of-the-art algorithm. Moreover, this paper outperforms most deep learning methods on three well-known benchmark datasets, including the indoor dataset S3DIS and the outdoor large-scale scene datasets Semantic3D and Toronto-3D. The proposed ResDLPS-Net achieves the best performance on the S3DIS dataset. The average accuracy (mA) and overall accuracy (OA) are 82.3% and 88.1%, respectively. Notably, the proposed ResDLPS-Net attains a mIoU of 80.27% on the Toronto-3D dataset, which is 6.00% higher than the best results published currently.

1. Introduction

Achieving precise environmental perception is a key demand for autonomous vehicles to realize dynamic monitoring of the surrounding environment in complicated and moving spaces and then make judgments based on the acquired information (Janai et al., 2020). The task of environmental perception first needs to obtain abundant data from the real environment accurately and then process the acquired data (Van Brummelen et al., 2018). Although mono and stereo cameras are capable of providing 3D geometry, it is difficult to do so precisely in an occluded environment. Camera-based perception systems are more susceptible to occlusion, illumination variations, and camera pose misalignment (Wang and Zhou, 2019). LiDAR can acquire relatively accurate 3D geometry compared to camera-based perception systems. Deep learning research (Fang and Lafarge, 2019; Xia et al., 2021; Li et al., 2020a; Liu et al., 2020) with regard to feature learning from LiDAR point clouds leads to accelerated advancement of autonomous driving. The applications of LiDAR point clouds can be broadly divided into two areas in the autonomous driving field. The first one is environmental perception and process, which can be used for scene understanding and object detection (Yang et al., 2018). The second is for producing and building high-definition maps and city models, which can be applied for localization

* Corresponding authors at: School of Computer Engineering, Jimei University, Xiamen 361021, China.
E-mail address: guorongcai.jmu@gmail.com (G. Cai).

https://doi.org/10.1016/j.isprsjprs.2021.09.024
Received 9 October 2020; Received in revised form 29 September 2021; Accepted 30 September 2021

and reference (Levinson et al., 2011). Applications of LiDAR point clouds generally encompass these types of tasks: classification, segmentation, detection, identification, and localization (Li et al., 2020c). This paper mainly concentrates on the semantic segmentation of 3D point clouds based on deep learning, which is a popular research subject in environmental perception at present.

Semantic segmentation of point clouds requires an accurate output of the semantic label for each point. Some segmentation algorithms based on deep learning aim to convert point clouds into voxels (Tchapmi et al., 2017; Brock et al., 2016; Çiçek et al., 2016; Shi et al., 2020) or multiviews (Lawin et al., 2017; Qi et al., 2016; Milioto et al., 2019) to extract point features, which may cause an increase in computational workload. PointNet (Qi et al., 2017a) is a milestone for deep learning networks in the field of semantic segmentation. It is the first algorithm that proposes to input point clouds directly to the network for training. Currently, many methods are based on improvements of PointNet and PointNet++ (Qi et al., 2017b), such as point convolutional neural network (PointCNN) (Li et al., 2018), kernel point convolution (KPConv) (Thomas et al., 2019), relational-shape CNN (RS-CNN) (Liu et al. 2019) and dynamic graph CNN (DGCNN) (Wang et al., 2019b), all of which have achieved excellent segmentation results. However, these methods divide the point cloud into small point cloud blocks and then sample multiple points from each point cloud block as the input to the network for training. Generally, the block-based approach will increase the data pre-processing time and may affect the complete geometry of the point cloud. Therefore, it is difficult for the network to effectively perceive the overall geometry of the object. In PointNet, LU-Net (Biasutti et al., 2019), 3DMV (Dai and Nießner, 2018), RS-CNN, etc., feature aggregation is achieved by a max-pooling operation. However, the max-pooling operation is related to the number of points. If there are too many points in the point cloud scene, the missing feature information will increase after the max-pooling operation. Therefore, the max-pooling operation performed on a large-scale point cloud may cause a large portion of the point information to be lost. The proposed ResDLPS-Net chooses attentive pooling as the pooling mechanism. This method devotes more attention to the feature information that is more critical to the current segmentation task from many point feature information and then automatically learns and aggregates this part of the features. The useless interference information will be disregarded to some extent.

High computational resources are demanded searching the neighboring points directly on the primary point cloud because of the large scale and disorderly nature of the point cloud data. Presently, the universal solution is to downsample the original point cloud and then perform subsequent processing on the downsampled points. This solution can reduce the computational workload to a significant degree. The prevailing approaches for sampling are farthest point sampling, grid sampling, random sampling, etc. Most algorithms (Qi et al., 2017b; Li et al., 2018; Liu et al., 2017; Zhang et al., 2019) tend to use farthest point sampling because of its uniform distribution of sampling points, which makes the spatial coverage of the downsampled points higher. However, farthest point sampling may lead to the relationship between points and time complexity is O(N^2), so it is suitable for point clouds of small scenes. When the amount of points in the whole scene is enormous, farthest point sampling will substantially increase computational workload. Voxel-based approaches (Choy et al., 2016; Thomas et al., 2019) typically tend to use grids as a data structuring method and then perform grid sampling. Points are associated with positions in the grid, and features of neighboring voxels are extracted by 3D convolution. While the grid data structure is efficacious, it wastes unnecessary computational space due to the excessive number of empty voxels in outdoor point clouds (Zhou and Tuzel, 2018). Random sampling may make the sampling points randomly distributed, and thus it is sensitive to areas of density imbalance. Downsampling the point cloud randomly may result in useful point features being discarded by chance. However, all three types of downsampling operations almost universally result in some degree of information loss. The time complexity for the random sampling is 0(1), which is not affected by the number of input points and is well suited for large-scale point clouds. Moreover, there is no need for data preprocessing operations such as converting point clouds into voxel grids. Random sampling is simple to operate. After weighing different sampling methods, the proposed ResDLPS-Net chooses random sampling that has high computational efficiency and low memory consumption as the sampling strategy. Then, a new feature extraction module, feature aggregation module, and residual-dense module are designed to improve the impact of information loss caused by random sampling on the semantic segmentation task.

In this paper, a novel semantic segmentation network ResDLPS-Net based on deep learning is proposed. ResDLPS-Net can be effectively applied to large-scale point cloud scenarios. Whether applied to the large indoor or outdoor dataset, this method has achieved good segmentation results. In summary, the contributions of the proposed ResDLPS-Net can be highlighted in three aspects:

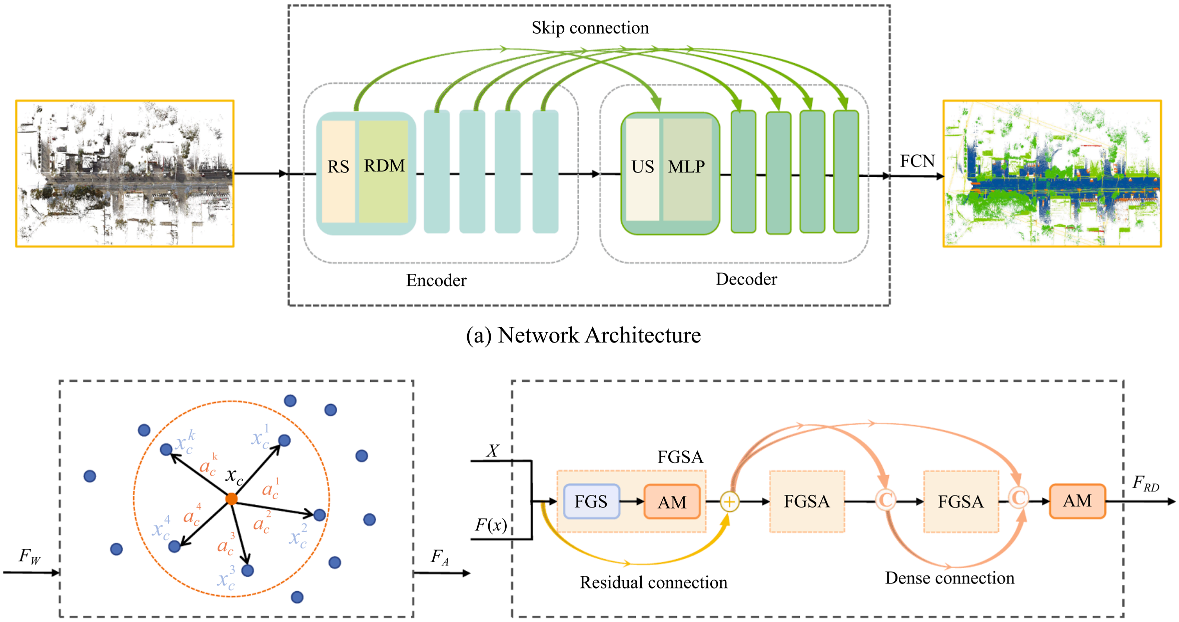
1. An effective feature extraction module (FGS) is proposed, which can efficiently extract neighbor features, geometric features, and semantic features. Then a feature aggregation module (FGSA) containing the attention mechanism is applied to learn and aggregate the crucial features of the neighborhood feature set.
2. A residual-dense module (RDM) is presented to add residual connections and dense convolutional connections to the FGSA module so that the ResDLPS-Net can extract more high-level distinguishable features.
3. The proposed ResDLPS-Net achieves state-of-the-art performance on the large-scale indoor dataset S3DIS (Armeni et al., 2016), outdoor dataset Toronto-3D (Tan et al., 2020), and exhibits comparable performance on the Semantic3D dataset (Hackel et al., 2017).

2. Related work

2.1. Semantic segmentation networks

Recently, some studies have begun to attempt to address large-scale point clouds directly. For example, the fuzzy counter-propagation network (FCPN) (Rethage et al., 2018) combines voxels to segment large-scale point clouds. Biasutti et al. (2019) propose an end-to-end architecture LU-Net for semantic segmentation, which takes into account the topology of the sensor and effectively creates multichannel distance images using the learned 3D features. This range image is then employed as an input to the U-Net network for segmentation. ScanComplete (Dai et al., 2018) can handle large-scale scenes with different spatial ranges and manages the cubic growth of the data size as the scene size increases. The multiscale fully convolutional network VIASeg (Zhong et al., 2019), based on a super squeeze residual module and semantic connection, is proposed. This network projects the fused red–greenblue (RGB) point cloud into a 2D spherical plane. The superpoint graph (SPG) structure (Landrieu and Simonovsky, 2018) uses superpoints and superpoint graphs to represent large scene point clouds. The input point clouds are divided into geometrically simple shapes, which are referred to as superpoints. The information is then transferred along the superedges that connect superpoints. The final segmentation result is optimized by the gated recurrent unit (GRU). Wu et al. (2019) is inspired by the SPG, which divides the point cloud into superpoints. This is the first time that a cross-attention method has been applied to a semantic segmentation network on a large-scale 3D sparse point cloud for autonomous driving scenarios.

The above methods have achieved promising segmentation results, but such preprocessing operations of voxelization or translation to superpoint graphs may lead to increased computation or high memory usage. PASS3D (Kong et al., 2019) proposes a novel two-stage 3D semantic segmentation framework. In the first stage, point clouds are segmented using an accelerated clustering algorithm, which does not require ground truth for fine clustering and can improve the recall rate



(a) Network Architecture

(b) Attention Mechanism

(c) Residual-Dense Module

Fig. 1. The framework of the proposed ResDLPS-Net. RS: random sampling. RDM: residual-dense module. US: upsampling. MLP: multi-layer perceptron. FCN: fully connected neural network. FGS: feature extraction module. AM: attention mechanism. FGSA: feature aggregation module. ⊕: element-wise addition. ©: channel-wise concatenation.

in a relatively short period of time. In the second stage, the neural network is further processed to estimate the semantic label of each point, and a novel data expansion method is proposed to enhance the network's segmentation ability of all categories (especially nonrigid objects). RandLA-Net (Hu et al., 2020) introduces a feature aggregation module to extend the influence domain of each point so that the overall geometric details of the input point cloud can be better preserved.

2.2. Deeper neural networks

Deep convolutional neural networks are widely used in the detection, classification, and segmentation (Krizhevsky et al., 2012; Sermanet et al., 2014). They integrate different levels of features in the end-to-end multi-layer network. The distinguishable features can be enriched by stacking the number of network layers (Simonyan and Zisserman, 2015; Szegedy et al., 2015), which shows that network depth is beneficial (He et al., 2016a). However, the increase in network depth may cause a vanishing gradient problem or an exploding gradient problem during backpropagation. The vanishing gradient problem indicates that the gradient of the deeper network layers in the neural network will gradually tend to zero. The exploding gradient problem means that the gradient between the network layers grows exponentially and becomes too large. Both vanishing and exploding gradient problems cause the network to converge difficult or even impossible to converge. That is, the loss value keeps oscillating sharply or stays high. This is mainly because the deep network is a stack of multi-layer nonlinear functions. The entire deep convolutional network can be considered as one composite multivariate nonlinear function (activation function), then seeking the partial derivative of the weight of different network layers for the loss function is equivalent to applying the chain rule of gradient descent. A chain rule is a form of continuous multiplication, so when the number of layers is deeper, the gradient will propagate exponentially. If the gradient value of the activation function close to the output layer is greater than 1, then the final gradient is prone to exponential growth, which will result in the exploding gradient problem; If the gradient value is less than 1, then it will easily decay to 0 after the chain rule, and the vanishing gradient problem will occur. In order to increase the number of network layers and then extract more discriminative features, He et al. (2016a) propose a residual network for image recognition, referred to as ResNet. ResNet introduces identity mappings, which element-wise adds the input and output of the residual block through a residual connection. This addition operation will not bring extraneous parameters to the network, but it is more useful to adjust the weights and can improve the training speed of the model. Therefore, the vanishing or exploding gradient problem of deep learning neural networks can be well improved by the residual network. In the case of the same number of layers, the convergence speed of the residual network is also faster. Many approaches (He et al., 2016b; Xie et al., 2017; Hu et al., 2018) have made a series of effective improvements based on ResNet. DenseNet (Huang et al., 2017) does not consider the problem of feature extraction from the perspective of increasing the number of network layers but instead uses continuous feature reuse to improve network performance. DenseNet connects the feature maps learned in different layers, which can increase the input changes of subsequent layers and improve work efficiency. Since the dense module does not need to relearn the redundant feature maps, it also has fewer parameters than a traditional convolutional network.

2.3. Knowledge gap

In general, there are still considerable challenges in directly processing large-scale point clouds:

(1) The size of large scenes and the number of points are uncertain, which requires the network to have some flexibility in the number of input points.
(2) Large-scale point clouds are typically unevenly distributed, which means that direct processing of point clouds may lead to regions with sparse points being discarded during sampling. Therefore, there are difficulties in semantic segmentation for small objects and edge points.
(3) Taking the whole large scene as input will increase the class of points and the number of objects included in each segmentation

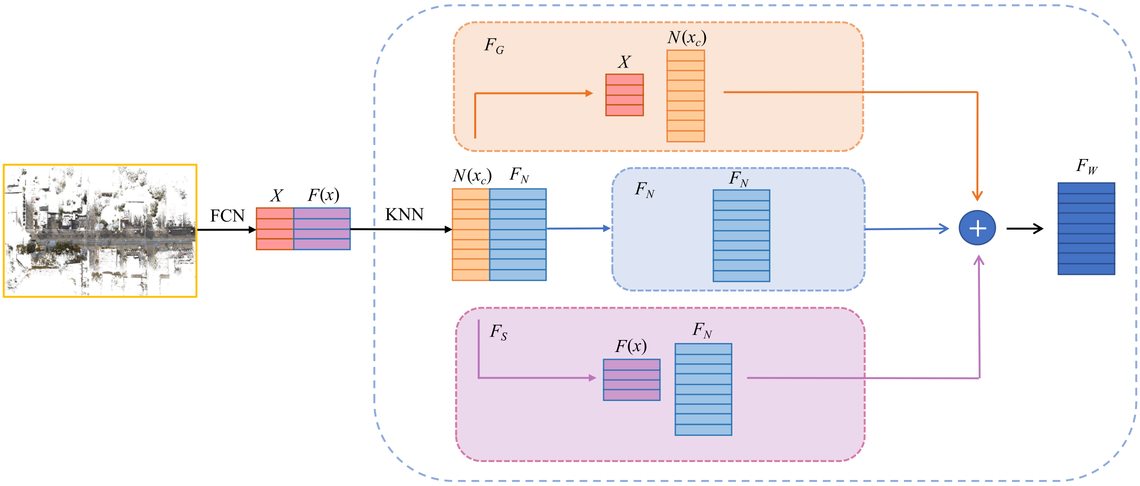


Fig. 2. Illustration of the feature extraction module (FGS). This module mainly consists of the extraction of neighbor features F_N, geometric features F_G, and semantic features F_S.

process. More complex point cloud geometry will increase the difficulty of segmentation.

(4) The increase of computation is a great challenge to the design of deep learning networks and the existing computational resources.

(5) Stacking too many network layers to extract more distinguishable features is difficult to improve the network performance and may lead to the vanishing or exploding gradient problem.

Considering the excellent performance of residual networks and dense convolutional networks in the image domain, the proposed ResDLPS-Net combines both to optimize the semantic segmentation of 3D point clouds. Large-scale point clouds contain abundant low, medium, and high-level features. The deep learning networks need to acquire these features to achieve good segmentation results. The residual network and dense convolutional network can satisfy the network for feature extraction and fusion in different network layers. The features extracted from the shallow network layers contain more 3D geometric information but less semantic information. Therefore, the proposed ResDLPS-Net uses the residual connection to element-wise sum the input and output of the residual blocks. With the increase in the number of network layers, the network will extract more distinguishable features with semantic information but relatively few geometric features. The dense convolutional networks can enhance the propagation between features and encourage the reuse of features. Therefore, the proposed ResDLPS-Net fuses features by dense convolutional connections. For that, a large-scale point cloud semantic segmentation network based on the joint optimization of residual connections and dense convolution connections is proposed. The ResDLPS-Net is an effective method to migrate mature networks in the image field to the processing of point cloud semantic segmentation problems.

In addition, the proposed ResDLPS-Net is designed with a novel feature extraction module to extract neighbor features, geometric features, and semantic features. However, ResDLPS-Net does not extract global features, so the network is flexible for the number of input points. Moreover, the receptive field of each point is increased by stacking multiple feature aggregation modules; thereby, the segmentation of small objects and edge points is also greatly improved. The proposed ResDLPS-Net is relatively lightweight and can be run on only one NVIDIA GeForce GTX 1080 Ti GPU.

3. Method

3.1. Network architecture

The generalized framework of ResDLPS-Net is depicted in the first figure of the graphical abstract and Fig. 1. ResDLPS-Net utilizes the encoder-decoder structure to construct the network framework. The network contains three main modules: the feature extraction module (FGS), the feature aggregation module (FGSA), and the residual-dense module (RDM). The original point cloud is first fed to the encoder module. Each encoder layer is composed of a random sampling module and an RDM. The FGS module and the attention mechanism constitute the FGSA module. The FGSA modules are connected via residual and dense connections to constitute the RDM. The purpose of FGS module is to extract neighbor features, geometric features obtained by encoding 3D coordinate information, semantic features between center points and neighboring points, and then output the neighborhood feature set. The goal of FGSA module is to learn and aggregate the key information of the neighborhood feature set via the attention mechanism. The target of RDM is to enable the network to extract more distinguishable features through the joint training of residual connections and dense convolutional connections. Each decoder layer incorporates the upsampling module and the multi-layer perceptron (MLP). The features are transferred between the encoder and decoder via skip connection. Finally, three fully connected layers are employed to predict the semantic label of each point.

The network contains a five-layer encoder, as shown in Fig. 1(a). Different from the traditional encoder-decoder structure, the proposed ResDLPS-Net adapts the internal structure of each encoder layer for the special application scenario of large scenes. Adjacent encoding layers employ RDM modules with different numbers of FGSA modules. Each encoder layer has a random sampling module, which gradually decreases the number of points in the point cloud. Since the number of points in different encoding layers is distinct, the number of FGSA modules inside each encoder layer is set variably. The number of FGSA modules for each of the five encoding layers on the S3DIS dataset and the Toronto-3D dataset are set to 2, 3, 2, 3, 2, respectively, denoted as the combination (2, 3, 2, 3, 2). The combination (2, 3, 2, 3, 2, 2) is the setting of the FGSA module in six encoding layers on the Semantic3D dataset.

The proposed ResDLPS-Net contains a five-layer decoder that uses the nearest neighbor interpolation algorithm to acquire the features of the upsampling points on the S3DIS dataset and Toronto-3D dataset. Similarly, the network comprises a six-layer decoder on the Semantic3D dataset. The skip connection connects the obtained point features with the intermediate features generated by the encoder layer, and then the network executes the shared MLP operation. Finally, the semantic labels are obtained after three fully connected layers.

3.2. Feature extraction module

This section introduces three types of feature extraction in the FGS module as shown in Fig. 2. The features include neighbor features, geometric features, and semantic features. The network takes the 3D points $X = \{x_1, x_2, x_i, ..., x_n\}$ with $X \in R^{n\times f}$ as input, where f represents the feature dimension of each point x_i. Each point includes a semantic label y_j from the semantic label set $Y = \left\{y_1, y_2, y_j, ..., y_m\right\}$. Firstly, the random sampled point cloud is input to the fully connected neural network (FCN) which achieves the extraction of initial features $F(x) = \{f_{x_1}, f_{x_2}, f_{x_i}, ..., f_{x_n}\}, F(x) \in R^{n\times(3+d)}$ for the 3D points X. Each 3D point is denoted as a center point x_c. FGS utilizes 3D points X and the initial features $F(x)$ as input. Then the k-nearest neighbor algorithm (KNN) is adopted to find k neighbors $N(x_c) = \{x_c^1, x_c^2, x_c^3, ..., x_c^k\}$ of each center point x_c.

3.2.1. Neighbor feature extraction

Since the initial features $F(x)$ of each point have been obtained, the corresponding neighboring point feature set $F(N(x_c)) = \left\{f_{x_c}^1, f_{x_c}^2, f_{x_c}^3, ..., f_{x_c}^k\right\}$ is accessed by the stored index of the neighboring points, as shown in Eq. (1).

$$F_N = F(I(N(x_c) \rightarrow X)) \tag{1}$$

where I represents the mapping function that maps the indexes of neighboring points to the indexes of their corresponding original points. Neighbor features $F_N \in R^{n\times k\times d}$ will be further processed in the geometric feature extraction operation and semantic feature extraction operation. Therefore, the FGS module extracts the neighbor features directly with a combination of balancing computational time and computational resources.

3.2.2. Geometric feature extraction

Large-scale point clouds contain rich geometric features. The proposed ResDLPS-Net encodes the position information of the center and neighboring points to realize the extraction of geometric features. The location information covers four types of coordinate information, which are the 3D coordinates $P_{x_c} = (x, y, z)$ of the center points $X_C = \{x_c\}|_{c=1}^n$, the coordinates $P_{x_c^k} = (x, y, z)$ of neighboring points$x_c^k \in N(x_c)$, the relative coordinates $P_r = [X_{x_c}, Y_{x_c}, Z_{x_c}]^T - \left[X_{x_c^k}, Y_{x_c^k}, Z_{x_c^k}\right]^T, \quad x_c^k \in N(x_c)$ of each center point and its neighboring points, and the Euclidian distance. The distance vector between P_{x_c} and $P_{x_c^k}$ is defined as $\Delta P = P_{x_c} - P_{x_c^k}$, thus the vectors corresponding to the three axes are ΔP_x, ΔP_y, ΔP_zrespectively. Then the Euclidian distance can be further measured as $P_s = \sqrt{\Delta_x^2 + \Delta_y^2 + \Delta_z^2}$. The four types of location information are combined as follows:

$$F_G = g\left(L\left(P_{x_c}, P_{x_c^k}, P_r, P_s\right)\right) \tag{2}$$

where the L function represents the linkage of the four location information, and the paper uses the concatenation connection. The g function encodes position information into geometric features. The MLP operation is applied as the g function to adjust the weights of the four location information in this paper, which facilitates the extraction of geometric features $F_G \in R^{n\times k\times d}$.

3.2.3. Semantic feature extraction

Semantic features comprise abundant point cloud contextual information. In Eq. (3), the high-level semantic features $F_S \in R^{n\times k\times d}$ are obtained by encoding the center and neighboring point features.

$$F_S = Relu(Conv(Concat(F(x_c), F(N(x_c))))) \tag{3}$$

Notably, the neighbor features, the geometric features, and the semantic features all contribute to the final neighborhood features. The final neighborhood feature set $F_W \in R^{n\times k\times 3d}$ obtained via the concatenation operation $\oplus$ is given as follows:

$$F_W = F_N \oplus F_G \oplus F_S \tag{4}$$

3.3. Feature aggregation module

The computational complexity of directly inputting large-scale point clouds into the network for training is significantly higher than dividing large point clouds into small point cloud blocks. Therefore, in order to efficiently utilize the available computational resources, the attention mechanism is applied to aggregate and optimize the neighborhood features set F_W, as shown in Fig. 1(b). The attention weights between center points and neighbors can be expressed as:

$$\widetilde{a_c^k} = T_W(X_C, N(x_c), F_W) \tag{5}$$

where T_W denotes the fully connected network, and then the softmax function is used for $\widetilde{\underset{c}{\overset{k}{a}}}$ as follows:

$$\alpha_c^k = softmax\left(\widetilde{\underset{c}{\overset{k}{a}}}\right) = \frac{exp\left(\widetilde{\underset{c}{\overset{k}{a}}}\right)}{\sum_{i=1}^k exp\left(\widetilde{\underset{c}{\overset{k}{a}}}\right)} \tag{6}$$

where α_c^k is the normalized attention weight. The output $F_A \in R^{n\times d'}$ of the feature aggregation module is indicated in Eq. (7), and b denotes a learnable bias.

$$F_A = \sum_{x_c^k \in N(x_c)} \alpha_c^k F_W + b \tag{7}$$

The FGSA module has the following main contributions. Firstly, the attention mechanism can select the information that is more critical to the current task from a large number of features. Therefore more distinguishable semantic information can be learned. Secondly, the attention mechanism can reduce attention to unimportant features or irrelevant features.

When a large-scale point cloud is taken to be trained directly, the number of points in different encoding layers will vary remarkably. The proposed ResDLPS-Net stacks different numbers of FGSA modules in distinct encoding layers to minimize the number of redundant layers. This approach can guarantee high accuracy while increasing computational efficiency to a larger extent. The encoder contains five encoding layers. When the number of FGSA modules in each of the five encoding layers is 2,3,2,3,2, i.e., the combination is (2,3,2,3,2), the proposed ResDLPS-Net achieves the highest accuracy, as shown in the ablation experiment. Fig. 1(b) illustrates the combination of three FGSA modules. Connecting multiple FGSA modules in sequence can increase the receptive field of each point. Therefore, although some points are randomly discarded during the downsampling process, the FGS module can also extract rich geometric features. In the initial downsampling stage, most of the points in the point cloud are still retained, which means that stacking too many FGSA modules will greatly increase the calculation time. As the downsampling ratio of the point cloud progressively increases, the point cloud gradually becomes sparse, which leads to a loss of geometric information to some extent. In this case, relatively more FGSA modules need to be stacked to increase the

Table 1
The dimension of each variable.

Variable	Dimension
X	(n,f)
f_{x_i}	$(1,3+d)$
$F(x)$	$(n,3+d)$
$N(x_c)$	(n,k,d)
F_N	(n,k,d)
F_G	(n,k,d)
F_S	(n,k,d)
F_W	$(n,k,3d)$
F_A	(n,d')
F_{RD}	(n,d'')

receptive field of each point. Thus, the final combination is (2, 3, 2, 3, 2).

3.4. Residual-Dense module

Fig. 1(c) illustrates the introduction of two types of cross-layer connections between each FGSA module to construct an RDM. The first coding layer takes the initial features $F(x)$ of 3D points extracted by the fully connected network as the input of the residual connection. The input of the residual connection of the subsequent coding layers is the features extracted by the previous coding layers. The mapping with the introduction of the residual block is more sensitive to changes in the output and is more efficient in adjusting the weights, so the segmentation performance is better. Therefore, a residual connection is added to the first FGSA module. The features extracted by the lower-level FGSA module contain more neighbor and geometric information. However, relatively few distinguishable features can be acquired due to fewer convolution operations. The features derived from the higher-level FGSA modules have more distinguishable semantic features, but the FGSA modules have a poor perception of local details. Therefore, the proposed ResDLPS-Net unifies the dimensions of the features extracted by each FGSA module. Then the dense connection operations are performed to fuse the features. Finally, the attention mechanism is implemented to optimize the final fused features to achieve the complementary advantages of features in different layers. Consequently, the number of network layers can be increased, and more distinguishable features $F_{RD} \in R^{n\times d''}$ can be extracted. Meanwhile, the problem of network degradation due to the increase of network layers can be improved to some extent.

The dimension of each variable is shown in Table 1. n represents the total number of input points. f indicates the feature dimension of each input point. d denotes the feature dimension of each F_N, F_G and F_S. d' stands for the feature dimension of each F_A. d'' is the feature dimension of each F_{RD}.

4. Experiments

To fully validate the scalability of the proposed ResDLPS-Net for point clouds obtained from different sensors and for various scenes, the indoor scene dataset S3DIS dataset and the large outdoor scene datasets Semantic3D and Toronto-3D are selected for the experiments. The S3DIS dataset is collected by RGB-D sensors. The Semantic3D dataset is acquired with the terrestrial laser scanner. The Toronto-3D dataset is obtained by Teledyne Optech Maverick. The proposed ResDLPS-Net is compared with the state-of-the-art semantic segmentation networks. Four commonly used evaluation metrics, including OA, mA, IoU and mIoU, are applied to evaluate the proposed network. It is worth mentioning that a detailed ablation study is provided in Section 4.3.

Table 2
Segmentation results (%) of different methods on the S3DIS dataset (6-fold cross-validation).

Method	OA	mA	mIoU
PointNet (Qi et al., 2017a)	78.6	66.2	47.6
SPG (Landrieu and Simonovsky, 2018)	86.4	73.0	62.1
3P-RNN (Ye et al., 2018)	86.9	–	56.3
RSNet (Huang et al., 2018)	–	66.5	56.5
PointCNN (Li et al., 2018)	**88.1**	75.6	65.4
PointWeb (Zhao et al., 2019)	87.3	76.2	66.7
ShellNet (Zhang et al., 2019)	87.1	–	66.8
KPConv (Thomas et al., 2019)	–	79.1	**70.6**
FPConv (Lin et al., 2020)	–	–	68.7
RandLA-Net (Hu et al., 2020)	88.0	82.0	70.0
ResDLPS-Net	**88.1**	**82.3**	70.2

4.1. Indoor scene segmentation

4.1.1. Dataset description

The RGB-D sensor has a limited measurement range, so the point cloud density of the S3DIS dataset is relatively low. The scene consists of five large indoor areas in three different buildings, covering 6,020 square meters approximately. The S3DIS dataset has over 215 million points as well as 13 semantic elements. The semantic labels consist of structural elements (door, window, wall, ceiling, floor, beam, and column), furnitures (board, sofa, table, chair, and bookcase), and clutter. Each point is composed of seven attributes: X, Y, Z, R, G, B, and label. Artificial indoor spaces usually have the same categories of objects, but the geometric structure varies greatly. Therefore, the segmentation of the S3DIS dataset is challenging.

4.1.2. Experimental setup

First, the number of neighbors for each center point is set to 16. Second, the batch sizes for training and testing are 4 and 12, respectively. The number of training and validation steps per epoch is 500 and 100, respectively. Finally, the experiments are implemented on the NVIDIA GeForce RTX 1080 Ti GPU.

4.1.3. Results and visualization of S3DIS

4.1.3.1. Overall evaluation. The experimental part evaluates three metrics that are widely used in semantic segmentation algorithms (Qi et al., 2017a; Landrieu and Simonovsky, 2018; Thomas et al., 2019), including mIoU, OA, and mA. Table 2 shows that the performance of the proposed ResDLPS-Net is approaching or better than that of the state-of-the-art methods. For example, ResDLPS-Net achieves the best performance on the evaluation criteria mA, 0.3% higher than the state-of-the-art method RandLA-Net, and 9.3% higher than SPG with the whole large scene as input. The proposed ResDLPS-Net has a slightly lower mIoU than KPConv on the S3DIS dataset, but also achieves a good result of 70.2%, exceeding (Qi et al.,2017a,b; Wang et al,2019b; Ye et al,2018; Huang et al,2018; Chen et al,2019; Li et al,2018; Zhao et al,2019; Zhang et al,2019; Jiang et al,2019; Hu et al,2020). The proposed ResDLPS-Net and PointCNN are tied for first place in OA computing.

4.1.3.2. Visualization comparison. Fig. 3 shows the visual segmentation results of the proposed ResDLPS-Net on the indoor dataset S3DIS. The proposed ResDLPS-Net is compared with two point cloud semantic segmentation methods (Landrieu and Simonovsky, 2018; Hu et al., 2020). The difficulty in segmenting artificial scene datasets is that the same category generally has different shapes. The segmentation results of ResDLPS-Net are approaching the ground truth in most categories. As shown in Fig. 3d (1st row), RandLA-Net segments the edges of some cluttered objects into bookcases. The proposed ResDLPS-Net has a similar situation, but only a relatively small number of cluttered objects are incorrectly segmented in Fig. 3e (1st row). The proposed ResDLPS-

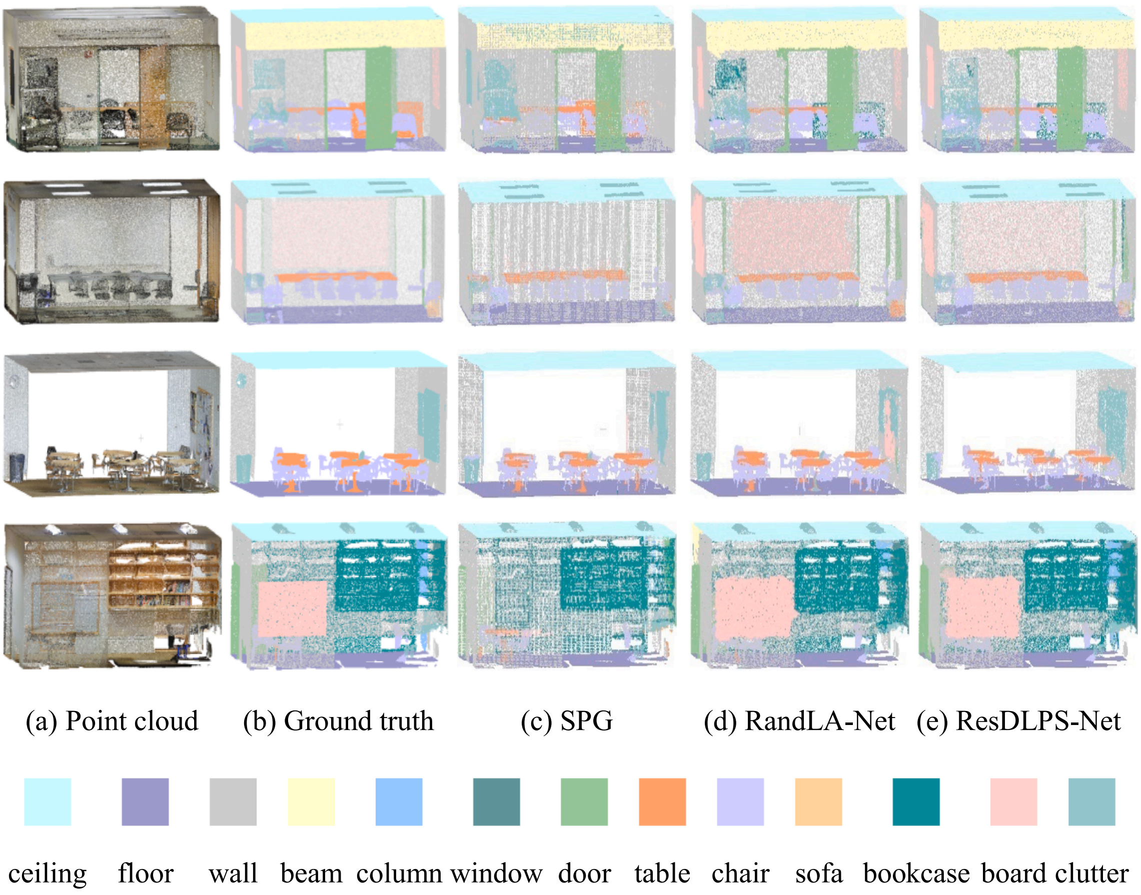


Fig. 3. Segmentation results on the S3DIS dataset. Figures from top to bottom are the conference room in Area 6, the conference room in Area 4, the lobby in Area 5 and the office in Area 5.

Net outperforms the latest algorithm RandLA-Net in the segmentation of object edges. Fig. 3d (2nd row) shows that RandLA-Net segments the sofa into a table and is unable to clearly identify the boundary between the wall and the board. RandLA-Net also segments the edge of the wall into the beam in Fig. 3d (4th row). Fig. 3c (2nd row, 4th row) indicates that SPG cannot segment the boards, nor can it adequately segment the edges of clutter and wall in Fig. 3c (3rd row). After comparison, it is observed that almost all of these classes belong to planar objects, and thus only limited geometric information is provided. The proposed ResDLPS-Net performs well on these planar objects. This result is mainly attributed to the FGS module that can capture the geometric features of each object well and RDM that helps the network to extract more distinguishable features. However, the proposed ResDLPS-Net does not perform well on some objects with similar structures. It may confuse these structurally similar categories, such as the table and bookcase in Fig. 3d (1st row). RandLA-Net also has a similar mis-segmentation phenomenon. SPG performs effectively on the segmentation of the table, mainly because SPG constructs a superpoint graph that can capture the non-planar geometric features validly.

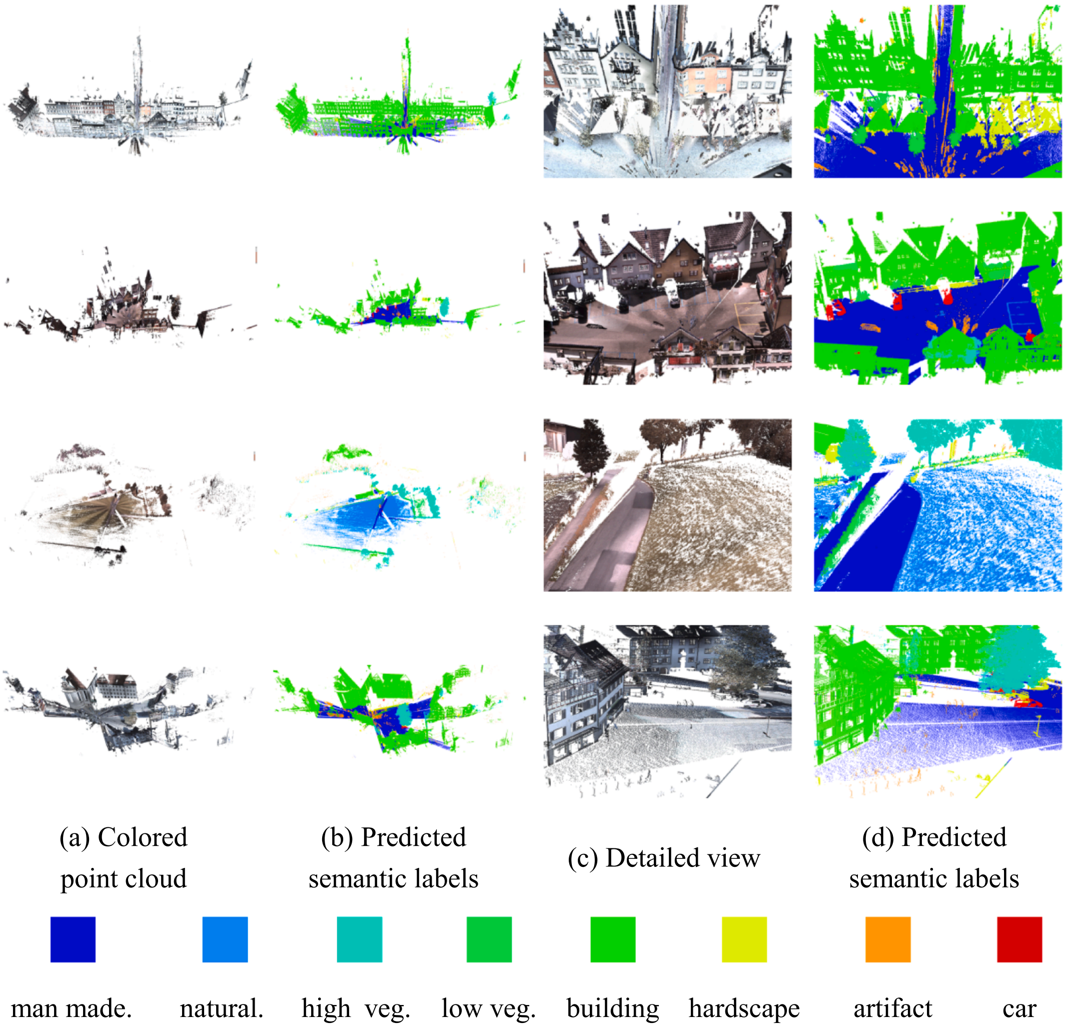
4.2. *Outdoor scene segmentation*

4.2.1. *Dataset description*

The Semantic3D dataset acquired by the terrestrial laser scanner contains about 4 billion 3D points. Each point consists of eight attributes: X, Y, Z, R, G, B, intensity and label. The scenes in the Semantic3D dataset include a variety of different natural and artificial scenes, which can effectively prevent the network from overfitting. Although the

Table 3
Segmentation results (%) of different methods on the Semantic3D dataset (reduced-8).

Method	mIoU	OA	Man made terrain	Natural terrain	High vegetation	Low vegetation	Buildings	Hardscape	Scanning artifacts	Cars
SnapNet (Boulch et al., 2017)	59.1	88.6	82.0	77.3	79.7	22.9	91.1	18.4	37.3	64.4
SEGCloud (Tchapmi et al., 2017)	61.3	88.1	83.9	66.0	86.0	40.5	91.1	30.9	27.5	64.3
RF-MSSF (Thomas et al., 2018)	62.7	90.3	87.6	80.3	81.8	36.4	92.2	24.1	42.6	56.6
MSDeepVoxNet (Roynard et al., 2018)	65.3	88.4	83.0	67.2	83.8	36.7	92.4	31.3	50.0	78.2
ShellNet (Zhang et al., 2019)	69.3	93.2	96.3	90.4	83.9	41.0	94.2	34.7	43.9	70.2
GACNet (Wang et al., 2019a)	70.8	91.9	86.4	77.7	88.5	**60.6**	94.2	37.3	43.5	77.8
SPG (Landrieu and Simonovsky, 2018)	73.2	94.0	97.4	92.6	87.9	44.0	83.2	31.0	63.5	76.2
KPConv (Thomas et al., 2019)	74.6	92.9	90.9	82.2	84.2	47.9	94.9	40.0	**77.3**	**79.7**
RGNet (Truong et al., 2019)	74.9	94.5	**97.5**	**93.0**	88.1	48.1	94.6	36.2	72.0	68.0
RandLA-Net (Hu et al., 2020)	**77.4**	**94.8**	95.6	91.4	86.6	51.5	**95.7**	**51.5**	69.8	76.8
ResDLPS-Net	76.5	94.4	95.6	90.7	**89.2**	53.4	94.7	50.8	58.9	78.6


(a) Colored point cloud

(b) Predicted semantic labels

(c) Detailed view

(d) Predicted semantic labels

Fig. 4. Segmentation results on the Semantic3D dataset.

Semantic3D dataset has a relatively high point density, only a limited number of views are feasible. This dataset is divided into eight categories: man made terrain, natural terrain, high vegetation, low vegetation, buildings, hardscape, scanning artifacts, and cars. The Toronto-3D dataset is the latest large-scale outdoor point cloud dataset for semantic segmentation, which is obtained by the vehicle-mounted MLS system and covers approximately 1 km of road. The dataset contains nearly 78.3 million points. Each point is comprised of 10 attributes: X, Y, Z, R, G, B, intensity, GPS time, scan angle rank and label. The dataset is composed of four parts. L001, L003, and L004 are selected as the train set. L002 is chosen as the test set, which covers about 250 m. The dataset is classified into nine categories: road, road marking, natural, building, utility line, pole, car, fence, and the unclassified object. New and challenging categories such as road markings and utility lines are included. Road markings include a variety of markings, such as crosswalks and arrows. The proximity of road markings to the road surface increases the difficulty of segmentation.

4.2.2. Experimental setup

First, the batch size is programmed to 4 for training and 10 for evaluation on both datasets. Then, the number of training steps per epoch is 500, and the validation steps of each epoch are 100. Furthermore, the maximum training epoch on both datasets is set to 150. Eventually, The other settings are the same as those on the S3DIS dataset.

4.2.3. Results and discussions of semantic3D

4.2.3.1. Overall evaluation. Table 3 illustrates the quantitative results of the mainstream methods on the Semantic3D dataset. The proposed ResDLPS-Net outperforms most deep learning methods (Boulch et al. 2017; Thomas et al. 2018; Roynard et al. 2018; Zhang et al. 2019; Wang et al. 2019a; Thomas et al. 2019) in terms of the mIoU and OA. It can be observed from Table 3 that the proposed ResDLPS-Net performs poorly in the semantic segmentation of low vegetation, partly due to the difficulty in distinguishing low vegetation from natural terrain. The criteria for the distinction between natural terrain and low vegetation are not yet clear. Natural terrain contains a majority of grasses, while the segmentation criterion for low vegetation is flowers and bushes under 2 m. This defining criterion is relatively vague and has much overlap, making it difficult for the network to distinguish between the two boundaries accurately. This causes mis-segmentation to some extent. The proposed ResDLPS-Net is slightly worse than RandLA-Net in terms of mIoU and OA. This is mainly due to the suboptimal segmentation in the category of scanning artifacts, which severely affects the computation of OA. Scanning artifacts are generated by objects that move dynamically during the scanning process of the terrestrial laser scanner. Since scanning artifacts have no fixed shape, it is hard for deep learning networks to learn their features. Scanning artifacts are influenced by hardware devices such as scanners. The Toronto-3D dataset is obtained by the Teledyne Optech Maverick with almost no scanning artifacts.

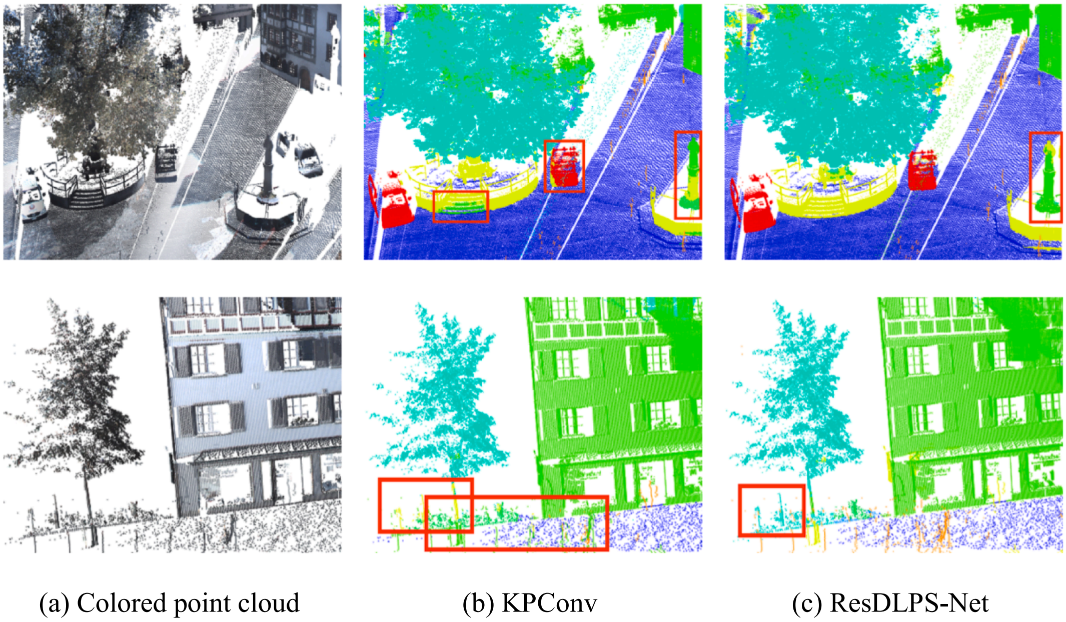

(a) Colored point cloud (b) KPConv (c) ResDLPS-Net

Fig. 5. Segmentation results on the Semantic3D dataset. Note: red boxes contain the points with incorrect semantic labels. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Segmentation results (%) of different methods on the Toronto-3D dataset.

Method	mIoU	OA	Road	Road marking	Natural	Building	Utility line	Pole	Car	Fence
PointNet++ (Qi et al., 2017b)	41.81	84.88	89.27	0.00	69.06	54.16	43.78	23.20	52.00	2.95
PointNet++ - MSG (Qi et al., 2017b)	59.47	92.56	92.90	0.00	86.13	82.15	60.96	62.81	76.41	14.43
MS-PCNN (Ma et al., 2019)	65.89	90.03	93.84	3.83	93.46	82.59	67.80	71.95	91.12	22.50
TGNet (Li et al., 2020b)	61.34	94.08	93.54	0.00	90.83	81.57	65.26	62.98	88.73	7.85
DGCNN (Wang et al., 2019b)	61.79	94.24	93.88	0.00	86.13	82.15	60.96	62.81	76.41	14.43
KPConv (Thomas et al., 2019)	69.11	95.39	94.62	0.06	96.07	91.51	**87.68**	**81.56**	85.66	15.72
MSTGNet (Tan et al., 2020)	70.50	95.71	94.41	17.19	95.72	88.83	76.01	73.97	**94.24**	23.64
RandLA-Net (Hu et al., 2020)	74.27	88.43	87.43	22.04	**96.36**	**92.69**	85.93	75.50	86.60	**47.64**
ResDLPS-Net	**80.27**	**96.49**	**95.82**	**59.80**	96.10	90.96	86.82	79.95	89.41	43.31

4.2.3.2. Visualization comparison. Fig. 4 shows four different natural and artificial scenes, including castles, squares, farms, sports fields, churches, etc. The proposed ResDLPS-Net achieves good segmentation performance in each scene. Fig. 5 illustrates the visual comparison of the proposed ResDLPS-Net with the latest algorithm KPConv. The red boxes indicate the segmentation errors. It is obvious that KPConv has segmentation errors in many places, such as stairs, cars, high vegetation, etc. ResDLPS-Net performs relatively satisfactorily in these places. However, it can be observed that there are still some incorrect segmentations when segmenting the boundaries of objects. The proposed ResDLPS-Net segments the poles next to the high vegetation as high vegetation. This is because the points of these poles are relatively few and sparsely distributed, resulting in comparatively fewer features provided by this part of the data. Meanwhile, the poles and trunks have similar shapes, and the proposed ResDLPS-Net cannot segment different objects with similar shapes well.

4.2.4. Results and discussions of Toronto-3D

4.2.4.1. Overall evaluation. Table 4 illustrates the results of the proposed ResDLPS-Net compared with the latest algorithms tested on the Toronto-3D dataset. ResDLPS-Net achieves the best performance in the computation of mIoU and OA. The optimal results are also obtained on the two classes of the dataset. The mIoU of the proposed ResDLPS-Net on the Toronto-3D dataset is 9.77% and 6% higher than that of MSTGNet and RandLA-Net, respectively. This validates the advantage of ResDLPS-Net in segmenting large urban scene datasets. RandLA-Net does not present segmentation results for the Toronto-3D dataset. To ensure fairness, the segmentation results in Table 4 are obtained from the GitHub of the MSTGNet algorithm, which provides the Toronto-3D dataset. The proposed ResDLPS-Net achieves 59.80% IoU in the new challenge category (road marking), which is 42.61% and 37.76% higher than MSTGNet and RandLA-Net, respectively. It can be clearly seen that PointNet++, TGNet, DGCNN, and KPConv can hardly segment road markings. The proposed ResDLPS-Net also has a good performance in the segmentation of fences with 19.67% higher than MSTGNet on mIoU. This is mainly attributed to the addition of residual connections and dense convolutional connections in ResDLPS-Net, which enables the network to extract more distinguishable high-level features. In addition, the inclusion of the attention mechanism optimizes the fusion of features.

*4.2.4.2. **Performance on** road markings.* Table 4 exhibits that the IoU of the proposed ResDLPS-Net on road markings is much higher than the current segmentation algorithm. A detailed visual comparison of road markings on the Toronto-3D dataset is presented in Fig. 6 to validate this result. The red boxes indicate the segmentation errors. It can be explicitly observed that KPConv can hardly segment zebra crossings and other signs on the road and segments this part of the point cloud into the road in Fig. 6c. Fig. 6d (2nd row) indicates that MSTGNet can segment a small number of simple road markings, but it is unable to segment the complex zebra crossings in Fig. 6d (1st row). This is because the road

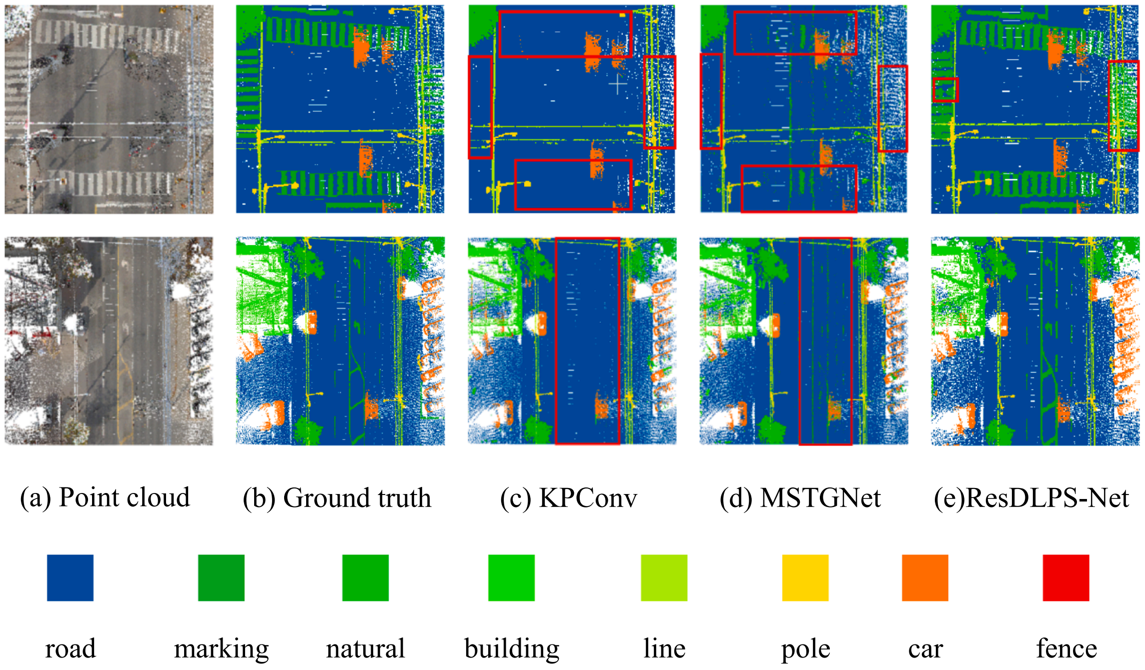

(a) Point cloud (b) Ground truth (c) KPConv (d) MSTGNet (e)ResDLPS-Net

Fig. 6. Segmentation results of road markings on the Toronto-3D dataset. Note: red boxes contain the points with incorrect semantic labels. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

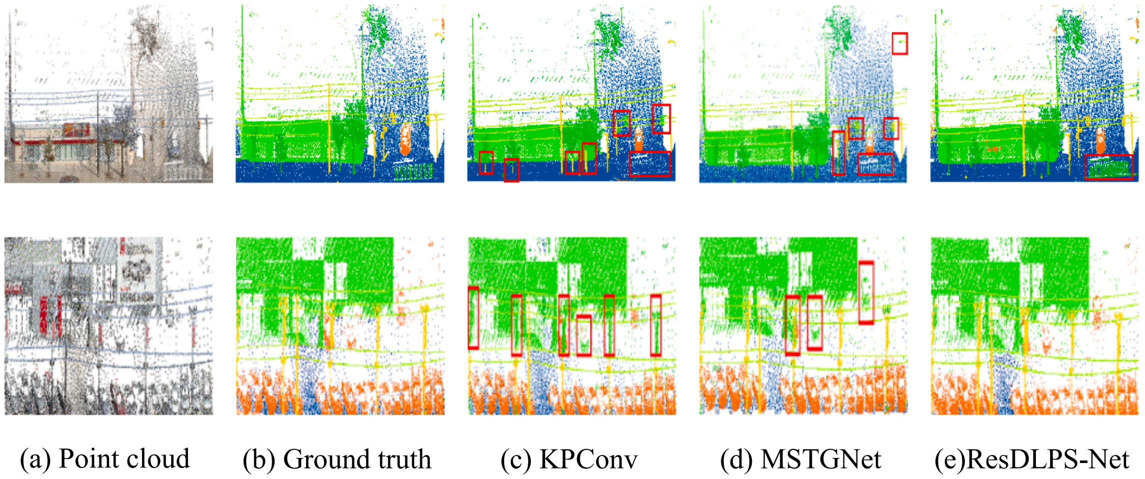

(a) Point cloud (b) Ground truth (c) KPConv (d) MSTGNet (e)ResDLPS-Net

Fig. 7. Segmentation results of poles on the Toronto-3D dataset.

and the road markings are on the same plane. In this case, the data provides few 3D geometric features and distinguishable features. Fig. 6e illustrates that the proposed ResDLPS-Net can segment lane lines clearly and zebra crossings effectively. These visualization results show the advantages of the proposed FGS module for planar object segmentation and the effectiveness of RDM for obtaining distinguishable features.

*4.2.4.3. **Performance on poles and** fences.* Figs. 7 and 8 compare the segmentation results of the poles and fences on different networks. KPConv segments tree trunks into poles in Fig. 7c (1st row) and the top of poles into buildings in Fig. 7c (2nd row). MSTGNet cannot segment the cars next to the building in Fig. 7d (2nd row). The proposed ResDLPS-Net performs well in the categories of natural, poles, and cars. Fig. 8c shows that KPConv cannot segment the fences next to trees. KPConv generates different shifts for each convolution, improving the network's ability to adapt to the geometry of the scene objects. However, it is difficult for KPConv to achieve relatively broad spatial coverage when segmenting large-scale outdoor scene point clouds, which leads to some classes with fewer points being segmented incorrectly. MSTGNet is capable of segmenting a portion of the fences with complete shapes, as shown in Fig. 8d. Fig. 8e indicates that the proposed ResDLPS-Net can segment the fences relatively well, but there are still some slight errors in the segmentation of the edges. This phenomenon is mainly attributed to the relatively few points in the edge part, which provides an insufficient number of effective features and thus increases the difficulty of semantic segmentation. The edge segmentation problem of objects is common in current semantic segmentation methods. However, it can be observed that ResDLPS-Net has a significant improvement compared to other algorithms.

4.3. Ablation study

Table 5 shows the impact of the various components of the proposed ResDLPS-Net on point cloud semantic segmentation. CRB

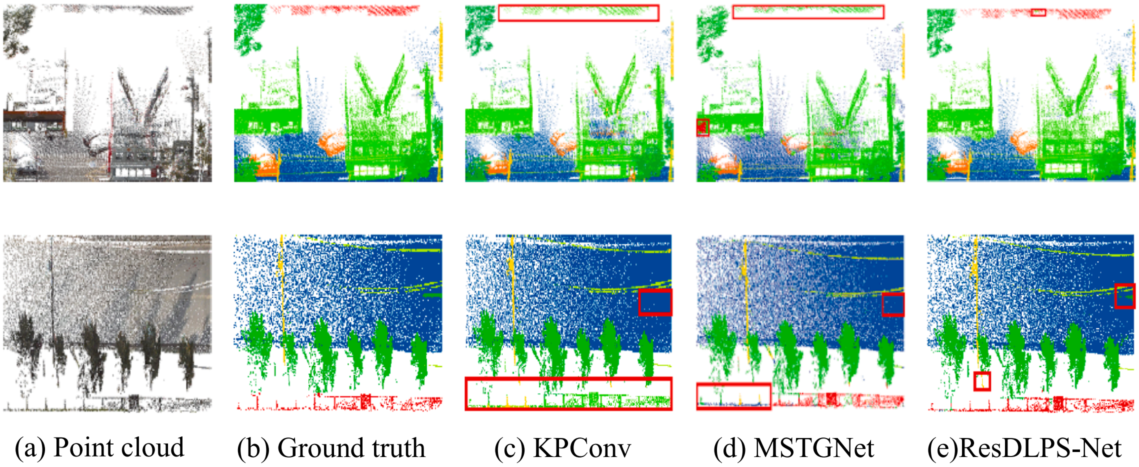


Fig. 8. Segmentation results of fences on the Toronto-3D dataset.

Table 5
Segmentation results (%) of different comparison experiments on the Semantic3D dataset. CRB: (concatenation, residual connection). CDB: (concatenation, dense convolutional connection). PRDB: (parallel, residual connection, dense convolutional connection). CRDB: (concatenation, residual connection, dense convolutional connection).

Method	mIoU	OA	Man made terrain	Natural terrain	High vegetation	Low vegetation	Buildings	Hardscape	Scanning artifacts	Cars
CRB	74.0	93.5	**96.4**	90.3	85.7	44.8	93.7	45.5	58.8	76.4
CDB	70.3	91.8	95.2	88.3	80.2	43.3	91.2	36.5	56.7	71.2
PRDB	72.4	92.4	95.8	88.8	83.5	**53.4**	90.9	42.8	50.8	73.2
CRDB	**76.5**	**94.4**	95.6	**90.7**	**89.2**	**53.4**	**94.7**	**50.8**	**58.9**	**78.6**

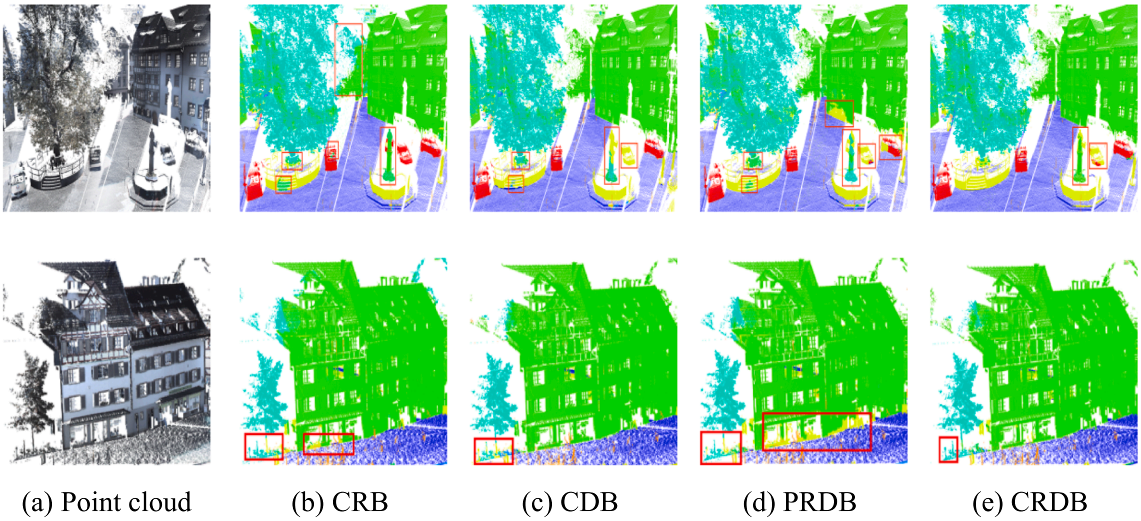


Fig. 9. Segmentation results of different comparison experiments on the Semantic3D dataset. CRB: (concatenation, residual connection). CDB: (concatenation, dense convolutional connection). PRDB: (parallel, residual connection, dense convolutional connection). CRDB: (concatenation, residual connection, dense convolutional connection).

(concatenation, residual connection) means that there is no dense convolution connection in ResDLPS-Net. CDB (concatenation, dense convolutional connection) indicates that ResDLPS-Net has no residual connection. PRDB (parallel, residual connection, dense convolutional connection) represents that the connection method of FGSA modules in parallel. CRDB (concatenation, residual connection, dense convolutional connection) denotes that the connection approach of the FGSA modules is concatenation. According to the quantitative comparison results of the changes of each component, it can be observed that when there is no residual module (concatenation, dense convolutional connection), the mIoU on the Semantic3D dataset is the lowest, as illustrated in Table 5. The segmentation is best when all modules are included. The experimental results validate that the proposed optimization with joint training of residual and dense convolutional connections is effective. Multiple FGSA modules in concatenation do help to optimize the final results of semantic segmentation.

4.3.1. *Performance on object edges*

Fig. 9 shows the effect of different functional modules on the semantic segmentation of object edges. The red boxes represent the segmentation errors. It can be observed that when the functional modules of ResDLPS-Net are incomplete, the network has many errors in segmenting high vegetation, cars, and hardscapes (stairs, garden walls, etc.). These experimental phenomena also validate the superiority of the

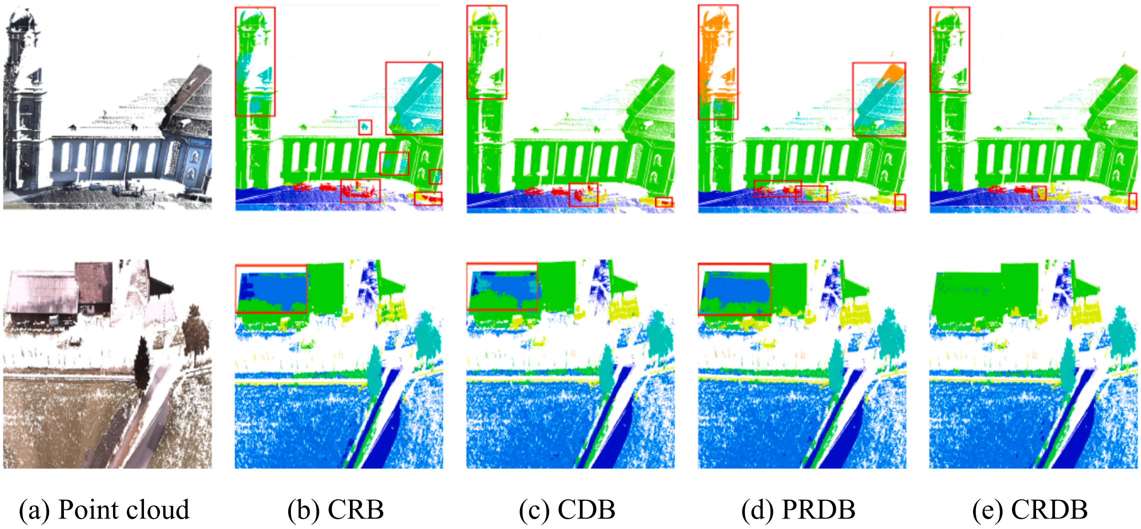


Fig. 10. Segmentation results of different comparison experiments on the Semantic3D dataset. CRB: (concatenation, residual connection). CDB: (concatenation, dense convolutional connection). PRDB: (parallel, residual connection, dense convolutional connection). CRDB: (concatenation, residual connection, dense convolutional connection).

Table 6
Segmentation results (%) of different comparison experiments on the S3DIS and Toronto-3D datasets.

Method	S3DIS			Toronto-3D	
	OA	mA	mIoU	OA	mIoU
ResDLPS-Net (residual)	87.7	81.2	69.0	95.7	78.2
ResDLPS-Net (dense)	86.9	80.8	67.8	96.0	78.3
ResDLPS-Net (residual + dense)	**88.1**	**82.3**	**70.2**	**96.4**	**80.2**

Table 7
MIoU of ablated networks on the Toronto-3D dataset.

FGSA	mIoU(%)
(2,2,3,3,3)	78.66
(3,3,3,2,2)	78.79
(2,3,2,3,2)(ResDLPS-Net)	**80.27**
(3,3,3,3,3)	78.91
(4,4,4,4,4)	78.22

proposed ResDLPS-Net. Fig. 9b (1st row) indicates that the network confuses buildings and high vegetation when there is no dense convolutional connection. Fig. 9d and 9e illustrate that the semantic segmentation is better when the connection method of each FGSA module in the RDM is concatenation rather than parallel. This improvement is attributed to the establishment of connections between each network layer and all layers in front of the layer, which is conducive to network training and increases the receptive field. The Semantic3D dataset does not display the true test labels, so it is unknown whether the sculpture in Fig. 9(d) (1st row) belongs to the building or the hardscape. However, PRDB segments most of the sculpture as high vegetation, which is obviously an incorrect result. Fig. 9(e) indicates that the segmentation of object edges is relatively best when the functional module is complete.

4.3.2. *Performance on buildings*

Fig. 10 presents a comparison of the impact of distinct functional modules on building segmentation. When no dense convolutional connection is applied, the network will segment the top of the building into high vegetation, as demonstrated in Fig. 10c (1st row). The dense convolutional connection is of great significance as it optimizes the features extracted from each network layer. Fig. 10d (1st row) displays that PRDB may result in the top of the building being segmented into scanning objects, which indicates that the network does not learn the features in this region efficiently. As illustrated in Fig. 10e (2nd row), only the complete ResDLPS-Net network can segment the roof well.

Similarly, to more convincingly verify that joint residual-dense optimization is better than separate training, this paper also conducts comparative experiments on the S3DIS dataset and the Toronto dataset in Table 6. The experiments show that joint training can obtain the best segmentation performance. This is because residual connections are helpful to increase the number of network layers and extract more distinguishable features, while dense convolutional connections focus on better reuse of features. Joint training can achieve complementary advantages.

Each coding layer is assigned an RDM. The ablation studies are conducted on the number of FGSA modules in the five RDMs in Table 7. (2, 3, 2, 3, 2) represents that the number of FGSA modules in the five RDMs are 2, 3, 2, 3, 2, respectively. (2, 3, 2, 3, 2) is also the best combination, which can continuously adjust the receptive field and has a moderate number of network layers. This approach can achieve a balance between extracting high-level semantic features and preventing network degradation. It can also be observed that although the segmentation results of other permutations are lower than (2, 3, 2, 3, 2), the mIoU of these permutations is still higher than the latest algorithms, such as MSTGNet and RandLA-Net.

5. Discussion

This paper conducts comparative experiments on three datasets collected by three different sensors. The scalability of the proposed ResDLPS-Net is discussed in terms of these three aspects: the sensors acquiring the dataset, the number of points in the objects, and the shape of the objects.

There are gaps in the overall performance of the proposed ResDLPS-Net in each dataset. ResDLPS-Net performs better on the S3DIS dataset, average on the Semantic3D dataset in general, but is most prominent on the Toronto-3D dataset. The S3DIS dataset was acquired by the RGB-D sensor in 2016. The RGB-D sensor obtains the 3D spatial location of every pixel from the depth map based on the placement of the center point of the camera, which in turn yields point clouds. However, the measurement of the RGB-D sensor is limited by the shooting light, the occlusion between objects, and the shooting angle. Therefore, the spatial architecture of the point cloud is not accurate enough, i.e., the spatial structure of the acquired point cloud is somewhat different from the original scene. However, since the structure of the objects in the indoor dataset is relatively simple, the influence is not severe. The Semantic3D

Table 8
Number of points for each class (thousand) on the Toronto-3D dataset (cited from (Tan et al., 2020)).

Section	Road	Road marking	Natural	Building	Utility line	Pole	Car	Fence	Unclassified	Total
LOO1	11,178	433	1,408	6,037	210	263	1,564	83	391	21,567
LOO2	6,353	301	1,942	866	84	155	199	24	360	10,284
LOO3	20,587	786	1,908	11,672	332	408	1,969	300	1,760	39,722
LOO4	3,738	281	1.310	525	37	71	200	4	582	6,748
Total	41,856	1,801	6,568	19,100	663	897	3,932	411	3,093	78,321

dataset was collected by terrestrial laser scanners in 2017. Although the density of the point cloud collected by terrestrial laser scanners is greater than 100 points/m^2, only limited views are viable. Also, the Semantic3D dataset has more scanning artifacts. The scanning artifacts do not have a fixed shape, so it is difficult for the network to learn their features. The Toronto-3D dataset was acquired by Teledyne Optech Maverick in 2020 and had a high density of almost 1000 points/m^2. Moreover, the Toronto-3D dataset covers the full range of the vehicle-mounted MLS sensor from the centerline of the road, nearly 100 m. Furthermore, the data has almost no scanning artifacts. The proposed ResDLPS-Net is somewhat dependent on the accuracy of the point clouds collected by the sensors, which is a limitation of ResDLPS-Net. However, with the development of sensors, the collected point clouds will become more and more accurate, and the practicability of the proposed ResDLPS-Net may be stronger.

Table 8 shows the number of points per category on the Toronto-3D dataset. The proposed ResDLPS-Net performs excellently in most of the categories with a high number of points. Moreover, ResDLPS-Net also has a great improvement in categories with fewer points compared to other networks. It can be seen from Figs. 6–8 that the proposed ResDLPS-Net has satisfactory segmentation effects on road markings, poles, and fences with fewer points. This is mainly because stacking multiple FGSA modules can effectively increase the receptive field of each point.

In terms of object shape, the proposed ResDLPS-Net can segment objects in the same plane well. For example, the boards, the clutters, and walls on the S3DIS dataset in Fig. 3, and roads and road markings on the Toronto-3D dataset in Fig. 6. The planar objects provide limited 3D geometric information, but the proposed ResDLPS-Net performs well in this case. This is the advantage of ResDLPS-Net because most of the networks have difficulty in segmenting road markings and roads, as shown in Fig. 6, as well as the boards and clutters on the wall as illustrated in Fig. 3. This result is mainly because the FGS module proposed in ResDLPS-Net can effectively capture the basic geometric and semantic features of each object. At the same time, the RDM can help the network extract more distinguishable features. However, the proposed ResDLPS-Net may confuse different objects with similar structures. For example, the tables and bookcases on the S3DIS dataset are shown in Fig. 3, and the poles and trunks, low vegetations, and natural terrain on the Semantic3D dataset in Fig. 5. This may be because objects with similar shapes initially have fewer distinguishable features. After random sampling, some key points with distinguishable features are discarded, making the objects more challenging to be segmented. Although the proposed ResDLPS-Net has a great improvement in the segmentation of object edges compared with other networks, the insufficiency of distinguishable features also leads to inaccurate segmentation of the edges of objects to some extent. For example, the clutter edge in Fig. 3, the fence edge in Fig. 8, and so on. This is a common mis-segmentation phenomenon in current semantic segmentation networks. Therefore, it is important to design an appropriate sampling strategy and neighbor query strategy so that the down-sampled point cloud achieves a higher spatial coverage in the future.

6. Conclusion

This paper introduces a novel large-scale point cloud semantic segmentation network without block dividing operation, referred to as ResDLPS-Net. The main contributions of this paper can be divided into the following parts. Firstly, a new local feature extraction module is designed to sufficiently extract neighbor features, geometric features, and semantic features. Then, the important features in the neighborhood feature set are learned and aggregated by the attention mechanism. Multiple feature aggregation modules are stacked to increase the perceptual field of each point. Further, the proposed ResDLPS-Net incorporates residual connections and dense convolutional connections into the semantic segmentation of 3D point clouds to extract more distinguishable features. Finally, the proposed ResDLPS-Net achieves satisfactory segmentation results on the indoor dataset S3DIS and the outdoor datasets Semantic3D and Toronto-3D.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant no. 41971424 and no. 61701191, the Key Technical Project of Xiamen Ocean Bureau under Grant no.18CZB033HJ11, the Key Technical Project of Xiamen Science and Technology Bureau under Grant nos. 3502Z20191018, 3502Z20201007, 3502Z20191022, 3502Z20203057, the Science and Technology Project of Education Department of Fujian Province under Grant nos. JAT190321, JAT190318, JAT190315, and the Natural Science Foundation of Fujian Province under Grant no. 2020J01701.

References

Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I.K., Fischer, M., Savarese, S., 2016. 3d semantic parsing of large-scale indoor spaces. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1534–1543, 10.1109/CVPR.2016.170.

Biasutti, P., Lepetit, V., Aujol, J., Brédif, M., Bugeau, A., 2019. Lu-net: An efficient network for 3d lidar point cloud semantic segmentation based on end-to-end-learned 3d features and u-net. In: Proceedings of the international conference on computer vision, pp. 942–950, 10.1109/ICCVW.2019.00123.

Boulch, A., Saux, B.L., Audebert, N., 2017. Unstructured point cloud semantic labeling using deep segmentation networks. Proceedings of the Eurographics Workshop on 3D Object Retrieval.

Brock, A., Lim, T., Ritchie, J.M., Weston, N., 2016. Generative and discriminative voxel modeling with convolutional neural networks. https://arxiv.org/abs/1608.04236.

Chen, L., Li, X., Fan, D., Cheng, M., Wang, K., Lu, S., 2019. Lsanet: Feature learning on point sets by local spatial attention. CoRR abs/1905.05442. http://arxiv.org/abs/1905.05442.

Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S., 2016. 3d–r2n2: A unified approach for single and multi-view 3d object reconstruction. In: Proceedings of the european conference on computer vision. Springer, pp. 628–644, 10.1007/978-3-319-46484-8_38.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3d u-net: Learning dense volumetric segmentation from sparse annotation. In: Medical Image Computing and ComputerAssisted Intervention, pp. 424–432, 10.1007/978-3-319-46723-8_49.

Dai, A., Nießner, M., 2018. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation, in: Proceedings of the european conference on computer vision, pp. 458–474. 10.1007/978-3-030-01249-6_28.

Dai, A., Ritchie, D., Bokeloh, M., Reed, S., Sturm, J., Nießner, M., 2018. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. In: in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4578–4587.

Fang, H., Lafarge, F., 2019. Pyramid scene parsing network in 3d: Improving semantic segmentation of point clouds with multi-scale contextual information. ISPRS J. Photogrammetry Remote Sens. 154, 246–258. https://doi.org/10.1016/j.isprsjprs.2019.06.010.

Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M., 2017. Semantic3d.net: A new large-scale point cloud classification benchmark. In ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. IV-1-W1,91–98.

He, K., Zhang, X., Ren, S., Sun, J., 2016a. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778, 10.1109/CVPR.2016.90.

He, K., Zhang, X., Ren, S., Sun, J., 2016b. Identity mappings in deep residual networks. In: Proceedings of the european conference on computer vision, pp. 630–645, 10.1007/978-3-319-46493-0_38.

Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-excitation networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141. https://doi:10.1109/CVPR.2018.00745.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. Randla-net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 11105–11114, 10.1109/CVPR42600.2020.01112.

Huang, G., Liu, Z., Weinberger, K.Q., 2017. Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2261–2269, 10.1109/CVPR.2017.243.

Huang, Q., Wang, W., Neumann, U., 2018. Recurrent slice networks for 3d segmentation of point clouds, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2626–2635. 10.1109/CVPR.2018.00278.

Janai, J., Güney, F., Behl, A., Geiger, A., 2020. Computer vision for autonomous vehicles: Problems, datasets and state of the art. Found. Trends Comput. Graph. Vis. 12, 1–308, 10.1561/0600000079.

Jiang, L., Zhao, H., Liu, S., Shen, X., Fu, C., Jia, J., 2019. Hierarchical point-edge interaction network for point cloud semantic segmentation. In: Proceedings of the international conference on computer vision, pp. 10432–10440, 10.1109/ICCV.2019.01053.

Kong, X., Zhai, G., Zhong, B., Liu, Y., 2019. PASS3D: precise and accelerated semantic segmentation for 3d point cloud, in: Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, pp. 3467–3473. 10.1109/IROS40897.2019.8968296.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: Proceedings of the neural information processing systems, pp. 1106–1114. 10.1145/3065386.

Landrieu, L., Simonovsky, M., 2018. Large-scale point cloud semantic segmentation with superpoint graphs, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4558–4567. http://10.1109/CVPR.2018.00479.

Lawin, F.J., Danelljan, M., Tosteberg, P., Bhat, G., Khan, F.S., Felsberg, M., 2017. Deep projective 3d semantic segmentation, in: Proceedings of the computer analysis of images and patterns, pp. 95–107. 10.1007/978-3-319-64689-3_8.

Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., Kolter, J.Z., Langer, D., Pink, O., Pratt, V.R., Sokolsky, M., Stanek, G., Stavens, D.M., Teichman, A., Werling, M., Thrun, S., 2011. Towards fully autonomous driving: Systems and algorithms. In: IEEE Intelligent Vehicles Symposium (IV), pp. 163–168, 10.1109/IVS.2011.5940562.

Li, W., Wang, F.D., Xia, G.S., 2020a. A geometry-attentional network for als point cloud classification. ISPRS J. Photogrammetry Remote Sens. 164, 26–40. https://doi.org/10.1016/j.isprsjprs.2020.03.016.

Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B., 2018. Pointcnn: Convolution on x-transformed points, in: Proceedings of the neural information processing systems, pp. 828–838.

Li, Y., Ma, L., Zhong, Z., Cao, D., Li, J., 2020b. Tgnet: Geometric graph CNN on 3-d point cloud segmentation. IEEE Trans. Geosci. Remote. Sens. 58, 3588–3600. https://doi.org/10.1109/TGRS.2019.2958517.

Li, Y., Ma, L., Zhong, Z., Liu, F., Cao, D., Li, J., Chapman, M.A., 2020c. Deep Learning for LiDAR Point Clouds in Autonomous Driving: A Review. IEEE Trans. Intelligent Transport. Syst. 1–21. https://doi.org/10.1109/TNNLS.2020.3015992.

Lin, Y., Yan, Z., Huang, H., Du, D., Liu, L., Cui, S., Han, X., 2020. Fpconv: Learning local flattening for point convolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4292–4301, 10.1109/CVPR42600.2020.00435.

Liu, F., Li, S., Zhang, L., Zhou, C., Ye, R., Wang, Y., Lu, J., 2017. 3dcnn-dqn-rnn: A deep reinforcement learning framework for semantic parsing of large-scale 3d point clouds. In: Proceedings of the international conference on computer vision, pp. 5679–5688, 10.1109/ICCV.2017.605.

Liu, J., Yu, M., Ni, B., Chen, Y., 2020. Self-prediction for joint instance and semantic segmentation of point clouds, in: Proceedings of the european conference on computer vision, Springer, pp. 187–204. 10.1007/978-3-030-58542-6_12.

Liu, Y., Fan, B., Xiang, S., Pan, C., 2019. Relation-shape convolutional neural network for point cloud analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8895–8904.

Ma, L., Li, Y., Li, J., Tan, W., Yu, Y., Chapman, M.A., 2019. Multi-scale point-wise convolutional neural networks for 3d object segmentation from lidar point clouds in large-scale environments. IEEE Trans. Intelligent Transport. Syst. 821–836 https://doi.org/10.1109/TITS.2019.2961060.

Milioto, A., Vizzo, I., Behley, J., Stachniss, C., 2019. Rangenet ++: Fast and accurate lidar semantic segmentation, in: Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, pp. 4213–4220. 10.1109/IROS40897.2019.8967762.

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 77–85. https://doi.org/10.1109/CVPR.2017.16.

Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J., 2016. Volumetric and multi-view cnns for object classification on 3d data. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5648–5656. https://doi.org/10.1109/CVPR.2016.609.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 5105–5114.

Rethage, D., Wald, J., Sturm, J., Navab, N., Tombari, F., 2018. Fully convolutional point networks for large-scale point clouds, in: Proceedings of the european conference on computer vision, pp. 625–640. 10.1007/978-3-030-01225-0_37.

Roynard, X., Deschaud, J., Goulette, F., 2018. Classification of point cloud scenes with multiscale voxel deep network. https://arxiv.org/abs/1804.03583.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y., 2014. Overfeat: Integrated recognition, localization and detection using convolutional networks. Proceedings of the international conference on learning representations.

Shi, H., Lin, G., Wang, H., Hung, T., Wang, Z., 2020. Spsequencenet: Semantic segmentation network on 4d point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4573–4582, 10.1109/CVPR42600.2020.00463.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. Proceedings of the international conference on learning representations.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9, 10.1109/CVPR.2015.7298594.

Tan, W., Qin, N., Ma, L., Li, Y., Du, J., Cai, G., Yang, K., Li, J., 2020. Toronto-3d: A large-scale mobile lidar dataset for semantic segmentation of urban roadways. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop, pp. 797–806, 10.1109/CVPRW50498.2020.00109.

Tchapmi, L.P., Choy, C.B., Armeni, I., Gwak, J., Savarese, S., 2017. Segcloud: Semantic segmentation of 3d point clouds. In: Proceedings of the international conference on 3D vision. IEEE Computer Society, pp. 537–547, 10.1109/3DV.2017.00067.

Thomas, H., Goulette, F., Deschaud, J., Marcotegui, B., LeGall, Y., 2018. Semantic classification of 3d point clouds with multiscale spherical neighborhoods. In: in: Proceedings of the international conference on 3D vision, pp. 390–398, 10.1109/3DV.2018.00052.

Thomas, H., Qi, C.R., Deschaud, J., Marcotegui, B., Goulette, F., Guibas,L.J., 2019. Kpconv: Flexible and deformable convolution for point clouds, in: Proceedings of the international conference on computer vision, pp. 6410–6419. 10.1109/ICCV.2019.00651.

Truong, G., Gilani, S.Z., Islam, S.M.S., Suter, D.F, 2019. Fast point cloud registration using semantic segmentation, in: Proceedings of the digital image computing: techniques and applications, pp. 1–8. 10.1109/DICTA47822.2019.8945870.

Van Brummelen, J., OBrien, M., Gruyer, D., Najjaran, H., 2018. Autonomous vehicle perception: The technology of today and tomorrow. Transport. Res. Part C Emerg. Technolog. 89, 384–406. https://doi.org/10.1016/j.trc.2018.02.012.

Wang, J., Zhou, L., 2019. Traffic light recognition with high dynamic range imaging and deep learning. IEEE Trans. Intell. Transp. Syst. 20, 1341–1352. https://doi.org/10.1109/TITS.2018.2849505.

Wang, L., Huang, Y., Hou, Y., Zhang, S., Shan, J., 2019a. Graph attention convolution for point cloud semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 10296–10305, 10.1109/CVPR.2019.01054.

Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2019b. Dynamic graph CNN for learning on point clouds. ACM Trans.Graph. 38, 146:1–146:12. 10.1145/3326362.

Wu, J., Jiao, J., Yang, Q., Zha, Z., Chen, X., 2019. Ground-aware point cloud semantic segmentation for autonomous driving. In: Proceedings of the 27th ACM international conference, pp. 971–979. https://doi.org/10.1145/3343031.3351076.

Xia, Y., Xu, Y., Wang, C., Stilla, U., 2021. Vpc-net: Completion of 3d vehicles from mls point clouds. ISPRS J. Photogrammetry Remote Sens. 174, 166–181. https://doi.org/10.1016/j.isprsjprs.2021.01.027.

Xie, S., Girshick, R.B., Dollár, P., Tu, Z., He, K., 2017. Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5987–5995, 10.1109/CVPR.2017.634.

Yang, B., Luo, W., Urtasun, R., 2018. PIXOR: real-time 3d object detection from point clouds, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7652–7660. 10.1109/CVPR.2018.00798.

Ye, X., Li, J., Huang, H., Du, L., Zhang, X., 2018. 3d recurrent neural networks with context fusion for point cloud semantic segmentation. In: in: Proceedings of the european conference on computer vision, pp. 415–430, 10.1007/978-3-030-01234-2_25.

Zhang, Z., Hua, B., Yeung, S., 2019. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. In: in: Proceedings of the international conference on computer vision, pp. 1607–1616, 10.1109/ICCV.2019.00169.

Zhao, H., Jiang, L., Fu, C., Jia, J., 2019. Pointweb: Enhancing local neighborhood features for point cloud processing, in. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5565–5573.

Zhong, Z., Zhang, C., Liu, Y., Wu, Y., 2019. VIASEG: visual information assisted lightweight point cloud segmentation. In: in: Proceedings of the international conference on image processing, pp. 1500–1504, 10.1109/ICIP.2019.8803061.

Zhou, Y., Tuzel, O., 2018. Voxelnet: End-to-end learning for point cloud based 3d object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, IEEE Computer Society. pp. 4490–4499. https://doi:10.1109/CVPR.2018.00472.