



OPEN EIDU-Net: edge-preserved inception DenseGCN U-Net for LiDAR point cloud segmentation

Xueli Xu^{1,2,3,5}, Jingyu Wang^{1,3,5}, Qiuquan Zhu^{1,3,5}, Ping Zhou⁴, Guohua Geng^{1,3}, Kang Li^{1,3}✉, Linzhi Su^{1,3} & Xin Cao^{1,3}✉

With the development of laser scanners and machine learning, point cloud semantic segmentation plays a significant role in autonomous driving, scene reconstruction, human-computer interaction, and other fields. In recent years, point cloud semantic segmentation based on deep learning has become one of the key research directions in point cloud processing. Due to the limited ability to exploit geometric details and contextual information in point clouds, most methods that adopt encoder-decoder architecture lose local structural information easily, especially detailed features, and extract features insufficiently. To address this issue, the edge-preserving inception DenseGCN U-Net (entitled as EIDU-Net) is proposed. EIDU-Net makes full use of the complementation between geometric details in the original point cloud and high-level features. The edge-preserved graph pooling (EGP) layer, the key module of the EIDU-Net, is designed to retain additional edge feature information from the original point cloud during pooling operations. Accordingly, the edge-preserved graph unpooling (EGU) layer can restore the feature graph more efficiently based on the additionally retained edge features. Extensive experiments demonstrate that our proposed EIDU-Net has remarkable improvements on semantic segmentation tasks under whatever S3DIS or Terracotta Warrior fragments. Our code is publicly available at <https://github.com/caoxin918/EIDU-Net>.

With the development of 3D point cloud processing technology, 3D point cloud segmentation is playing a significant role in many areas, such as scene reconstruction^{1–3}, autonomous driving⁴, and other fields. It is vital for these industries to learn the semantic information of point clouds accurately.

3D data have rich geometry, shape, and scale information, which can be represented in different ways, including voxels, meshes, and point clouds. Among them, point clouds are the most convenient for deep learning processing. Different from regular 2D images, 3D point clouds are unstructured, which makes the 2D CNN framework not applicable to point cloud semantic segmentation directly.

Early semantic segmentation methods like support vector machines (SVM), random forests, and decision trees^{5–7} showed good performance on small amounts of point cloud data but could not satisfy the big one. It is challenging for semantic segmentation in different complex scenes. To address this issue, deep learning-based methods have been proposed, such as voxel-based methods^{8–10}, projection-based methods^{11–15}, and point-based methods^{16–30}.

Voxel-based point cloud segmentation methods organized irregular point cloud data by dividing them into multiple regular voxel grids. These methods had good scalability but cost a large amount of memory and had low computational efficiency. Projection-based point cloud segmentation methods projected 3D point cloud into 2D images through various views, such as multi-view, spherical projection, etc., and then processed images by using the 2D frameworks. These methods could utilize the existing advanced 2D image processing algorithms directly but were prone to projection distortion and not suitable for complex scenes.

Point-based methods could operate the points directly, and preserve spatial information better than others. PointNet¹⁶, the classic point-wise MLP method, utilized shared multi-layer perceptions to acquire feature information of each point. It solved the problem of point cloud permutation invariance but lacked local feature extraction. PointNet++¹⁷ overcame the shortcomings of PointNet in local feature extraction, but due to the feature extraction layer by layer, its computational complexity was relatively high and required more computing

¹School of Information Science and Technology, Northwest University, Xi'an 710127, Shaanxi, China. ²Yan'an University, Yan'an 716000, Shaanxi, China. ³National and Local Joint Engineering Research Center for Cultural Heritage Digitization, Xi'an 710127, Shaanxi, China. ⁴Key Scientific Research Base of Ancient Polychrome Pottery Conservation, Emperor Qin Shihuang's Mausoleum Site Museum, Xi'an 710600, Shaanxi, China. ⁵Xueli Xu, Jingyu Wang, and Qiuquan Zhu have contributed equally to this work. ✉email: likang@nwu.edu.cn; caoxin918@hotmail.com

resources, making it unsuitable for applications that required fast processing of large amounts of point cloud data. PointWeb¹⁸ introduced the Adaptive Feature Adjustment (AFA) module for finding interactions between points to extract contextual features from local neighborhoods. It was still not able to handle information about relationships between point clouds and tasks with local features. PointNeXt²³ was an extension of the PointNet+++. It aimed to improve the performance of point cloud processing by improving the training strategy and the model extension strategy but did not explore deeply on architectural innovations. Point-M2AE²⁴ designed a multi-scale masking strategy to preserve local geometric integrity at different scales. Due to the multi-scale processing, Point-M2AE required high computational resources, and the performance of the model relied to some extent on the choice of masking strategy, and experiments were needed to determine the optimal masking ratio. By using cross-attention transformer, CASSPR²⁵ combined point-based and voxel-based approaches to fully utilize multi-scale spatial context to achieve fine matching of subtle geometric features. Due to the fusion, the computational complexity could be high, especially on large-scale point clouds, and the performance may degrade when the information in a single scan was limited to a small area. SOE-Net²⁶ demonstrated good advantages in point cloud-based location recognition tasks by combining self-attention and direction encoding mechanisms, but also faces challenges such as increased computational complexity and increased difficulty in model training. OPOCA²⁷ proposed an innovative annotation method with significant advantages, such as high annotation efficiency and high segmentation accuracy. However, this method also had some disadvantages, such as dependence on the selection of annotation points, limited performance in handling complex scenes, and its generality and applicability to be verified. SegTrans²⁸ showed significant benefits in the MLS point cloud semantic segmentation through transfer learning techniques and was a promising approach, but also faced challenges such as for some categories, the performance may still be not as good as expected.

Due to the correspondence between the graph structure and the point cloud data, graph-based methods were widely emerged recently to promote the development of point cloud processing. They converted point cloud data into the graph to preserve the geometric features between nodes and edges of the point cloud, and graph convolutional networks could be used for feature extraction. DGCNN²⁹ dynamically constructed point cloud structural information using KNN, and DeepGCNs³⁰ solved the problem that GCNs could not iterate multiple layers, making huge contributions to the subsequent development of GCNs. Graph U-Nets³¹ was an encoder-decoder model for graph data representation learning that introduced the gPool and gUnpool layers for pooling and unpooling graphs, respectively. These operations make it possible to implement U-Net-like structures on graph data, but they are depth-limited.

This paper proposes a novel segmentation network EIDU-Net. It makes full use of the complementation between geometric details in the original point cloud and high-level features to ensure learning effectiveness while reducing information loss. The edge-preserved graph pooling (EGP) layer, the key module of the EIDU-Net, is designed to retain additional edge feature information from the original point cloud during pooling operations. Accordingly, the edge-preserved graph unpooling (EGU) layer can restore the feature graph more efficiently based on the additionally retained edge features.

Our contributions are as follows:

- We propose a novel encoder-decoder network (EIDU-Net), which integrates Inception DenseGCN to ensure features can be learned at multiple levels and scales.
- We propose edge-preserved graph pooling (EGP) operation and edge-preserved graph unpooling (EGU) operation, which improve the traditional graph pooling and unpooling layer to maximize retention of spatial position feature information during the iteration process.
- We build the Terracotta Warrior dataset to challenge the proposed EIDU-Net on point cloud semantic segmentation task. The final results demonstrate that the proposed EIDU-Net achieves significant improvements in self-built datasets and public datasets.

Experiment and results

In this section, we test the proposed EIDU-Net on both Area 5 and 6-fold of S3DIS³². We compare our network with previous models on point cloud semantic segmentation. The experiments demonstrate that our EIDU-Net achieves good results in point cloud semantic segmentation. Some ablation studies are conducted to examine the contribution of different parameter settings in the model to performance improvement.

Datasets and evaluation metrics

We perform experiments on the S3DIS dataset to evaluate the robustness of the EIDU-Net model. The momentum and initial learning rate are 0.9 and 0.001. Each input point is represented with a 9-dim vector. The models are trained with 100 epochs and batch size 12. The S3DIS dataset contains rich indoor structural information, covering point clouds of over 610 rooms, about 2.73 million points. The point clouds are annotated by semantic categories, including 13 object categories, such as walls, windows, doors, tables, chairs, and a clutter category. Evaluating semantic segmentation performance on this dataset can reflect the algorithm's generalization ability for complex indoor scenes.

To normatively evaluate the segmentation performance of the EIDU-Net, the OA, mIoU, and mAcc are used in this article. They are defined as follows:

$$OA = \frac{\sum_{i=0}^k c_{ii}}{\sum_{i=0}^k \sum_{j=0}^k c_{ij}} \quad (1)$$

Method	Input	OA	mAcc	mIoU
PointNet	Point	-	49.0	41.1
SegCloud	Point	-	57.4	48.9
PointCNN	Point	85.9	63.9	57.3
SPGraph	Point	86.4	66.5	58.0
HPEIN	Point	87.2	68.3	61.9
MinkowskiNet	Voxel	-	71.7	65.4
PAG	Point	86.8	-	59.3
PointWeb	Point	87.0	66.6	60.3
PCT	Point	-	70.8	60.1
SegGCN	Point	88.2	70.4	63.6
KPConv	Point	-	72.8	67.1
RandLA-Ne	Point	87.2	71.4	62.4
DPFA-Net	Point	88.0	-	55.2
JSNet++	Point	89.1	64.7	58.0
Ours	Point	87.7	73.8	60.0

Table 1. Results on S3DIS Area-5 dataset for semantic segmentation. Significant values are in bold.

Method	Input	OA	mAcc	mIoU
PointNet	Point	78.5	66.2	47.6
DGCNN	Point	84.1	-	56.1
PointCNN	Point	88.1	-	65.4
SPGraph	Point	86.4	75.6	62.1
PAG	Point	88.1	-	65.9
PointWeb	Point	87.3	76.2	66.7
KPConv	Point	-	79.1	70.6
RandLA-Net	Point	88.0	82.0	70.0
DPFA-Net	Point	89.2	-	61.6
JSNet++	Point	88.7	72.8	62.4
Ours	Point	88.6	75.8	63.1

Table 2. Results on S3DIS 6-fold dataset for semantic segmentation. Significant values are in bold.

$$mACC = \frac{1}{k+1} \sum_{i=0}^k \frac{c_{ii}}{\sum_{j=0}^k c_{ij}} \quad (2)$$

$$mIoU = \frac{1}{k+1} \sum_{i=0}^k \frac{p_{ii}}{\sum_{i=0}^k p_{ij} + \sum_{i=0}^k p_{ji} - p_{ii}} \quad (3)$$

where k means the dataset has k categories. c_{ij} is the number of points from ground truth i and predicted as class j .

Evaluation on the S3DIS dataset

The quantitative results of our network are compared with previous methods, including PointNet¹⁶, DGCNN²⁹, SegCloud³³, PointCNN¹⁹, SPGraph³⁴, HPEIN³⁵, MinkowskiNet³⁶, PAG³⁷, PointWeb¹⁸, PCT³⁸, SegGCN³⁹, KPConv⁴⁰, RandLA-Net⁴¹, DPFA-Net⁴², and JSNet++⁴³. The results of these methods are mostly taken from existing literature. Our experimental results are shown in Tables 1 and 2, which demonstrate our method achieves improvements over both baselines. Overall, compared with other methods, our method achieves better accuracy on the S3DIS dataset.

As shown in Fig. 1, EIDU-Net achieves good performance on indoor scene segmentation. EIDU-Net can effectively segment complex indoor structures. We attribute this decent result to the EGP and EGU, as they help the model better aggregate local information in the data, leading to better-detailed segmentation results.

Table 3 demonstrates the experimental results for specific categories in semantic segmentation on the S3DIS dataset. It is worth noting that our method attains better accuracy on the wall and sofa classes. The results indicate that our EIDU-Net is adequate for the large-scale point cloud semantic segmentation task.

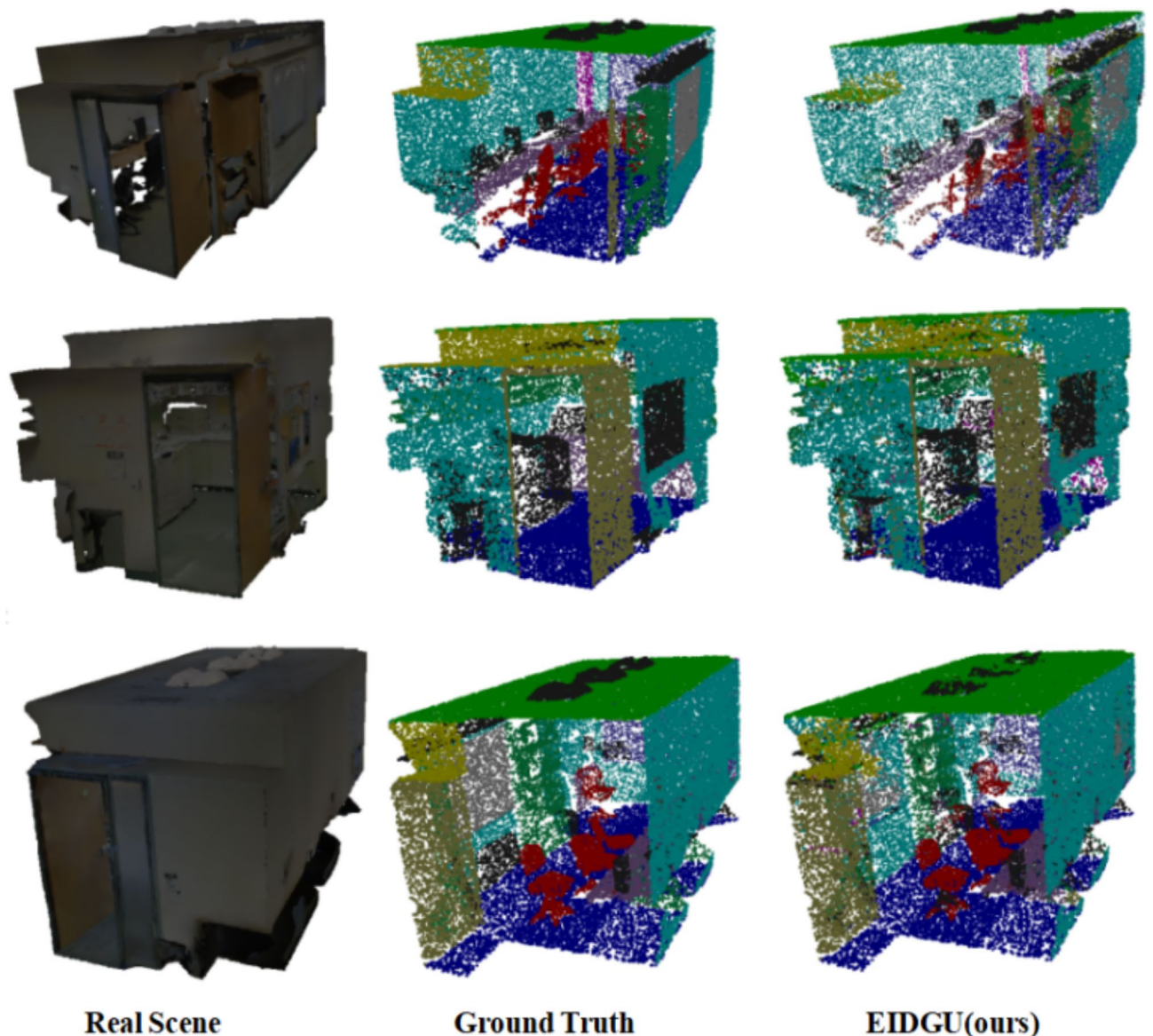


Fig. 1. Visualization results on S3DIS.

Different components

To verify the effect of different modules in the model, we conduct ablation experiments on Area 5 of S3DIS. The result is presented in Table 4. When replacing the EGP and EGU modules with the Graph Pooling and the Graph UnPooling of Graph U-Nets³¹, the OA and the mACC values decreased to 86.9 and 74.4. Similarly, lacking either of these two modules alone also leads to a decrease in OA and mACC. Based on the analysis of the results, it can be seen that aggregating feature details during pooling and unpooling helps reduce feature loss and improve final segmentation performance. Therefore, It is credible that optimizing pooling operations and unpooling operations for 3D data is effective and meaningful.

The influences of different numbers of DenseGCN layers

Although DeepGCNs solved the problem that GCNs could not iterate deeply, considering EIDG is just a basic module in the dual-nested structure, the appropriate number of iteration layers inside the DenseGcn needs to be evaluated through experiments. We conduct experiments with the number of layers set to 2–5.

As shown in Fig. 2, when the number of layers reaches 3, the segmentation effect no longer improves significantly and even shows a downward trend. The results show that 3 layers of DenseGCN are already sufficient. Due to the overfitting, further increasing the number of iteration layers will reduce the efficiency.

The influences of different numbers of EIDU-Net layers

In this section, the reason for conducting the ablation study is also to consider that too many iterations will affect the final segmentation effect. We set the number of layers of EIDU-Net from 1 to 5 for experiments. Figure 3

Method	mIoU	Ceiling	Floor	Wall	Beam	Column	Window	Door	Table	Chair	Sofa	Bookcase	Board	Clutter
PointNet	41.1	88.8	97.3	69.8	0.1	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
SegCloud	48.9	90.1	96.1	69.9	0.0	18.4	38.4	23.1	70.4	75.9	40.9	58.4	13.0	41.6
PointCNN	57.3	92.3	98.2	79.4	0.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	56.7
SPGraph	58.0	89.4	96.9	78.1	0.0	42.8	48.9	61.6	84.7	75.4	69.8	52.6	2.1	52.2
PCT	60.1	92.5	98.4	80.6	0.0	19.4	61.6	48.0	76.6	85.2	46.2	67.7	67.9	52.3
PointWeb	60.3	91.6	98.5	79.4	0.0	21.1	59.7	34.8	76.3	88.3	46.9	69.3	64.9	52.5
JSNet++	58.0	93.7	98.5	80.5	0.0	16.9	57.2	41.9	76.8	84.7	30.5	60.2	58.3	54.9
Ours	60.0	92.1	97.5	80.5	0.0	20.6	59.6	41.6	77.7	84.8	46.4	61.5	63.4	54.2

Table 3. mIoU results on S3DIS area-5 dataset for semantic segmentation.

EGP	EGU	OA	mACC
✗	✗	86.9	72.4
✓	✗	87.2	73.0
✗	✓	87.1	73.1
✓	✓	87.7	73.8

Table 4. Ablation results on S3DIS area-5 dataset.

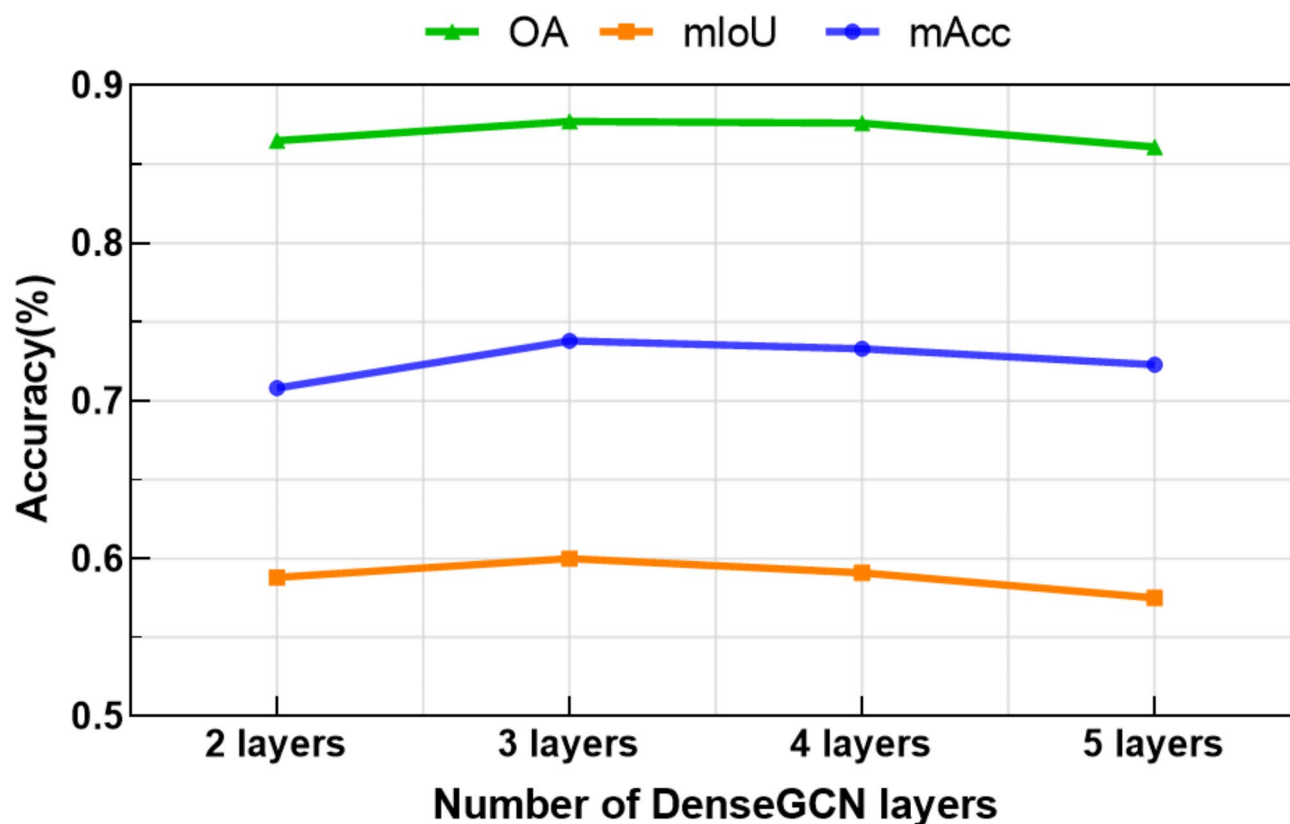


Fig. 2. Ablation Study on different numbers of DenseGCN layers.

shows the number of layers of EIDU-Net has a significant impact and 3 seems to be the most suitable number of layers.

Application on virtual restoration of real-world terracotta warriors

In recent years, point cloud processing for cultural heritage digitization can effectively avoid secondary damage to artifacts, which is of great significance for cultural relic protection and restoration. Among them, point cloud semantic segmentation can help researchers restore cultural relics more efficiently and accurately by semantically segmenting the digitized cultural relic models. As one of the focal points of our laboratory research is the virtual restoration of ceramic cultural relics, the Terracotta Warriors are the most notable ceramic cultural relics in our research, which are considered as one of the world's eighth wonders. In order to further confirm the effectiveness of the proposed EIDU-Net, we apply it to the challenged real-world datasets, i.e. Terracotta Warrior models.

Generally, 3D laser scanners can only scan partial point clouds of an object. In order to reconstruct the complete object, it is necessary to scan point clouds from different viewpoints and calculate view transformations based on point correspondences between point clouds. Then it is necessary to align point clouds from individual viewpoints to the same coordinate system and merge them into a complete 3D model. By following the steps above, we collect 200 complete Terracotta Warrior models by 3D object scanner, each composed of about 500,000 points, including xyz data, vertical normals, and RGB data. 160 models are utilized for training and 40 aside for testing. As shown in Fig. 4(a), we eliminate the vertical normals and RGB data, remaining xyz coordinates as input.

In traditional studies of point cloud semantic segmentation on the Terracotta Warriors datasets, the Terracotta Warriors 3D models are generally divided into six parts: head, body, left hand, right hand, left leg, and right leg. To better facilitate the cultural relic restoration and test the performance of our EIDU-Net, we separate the hands

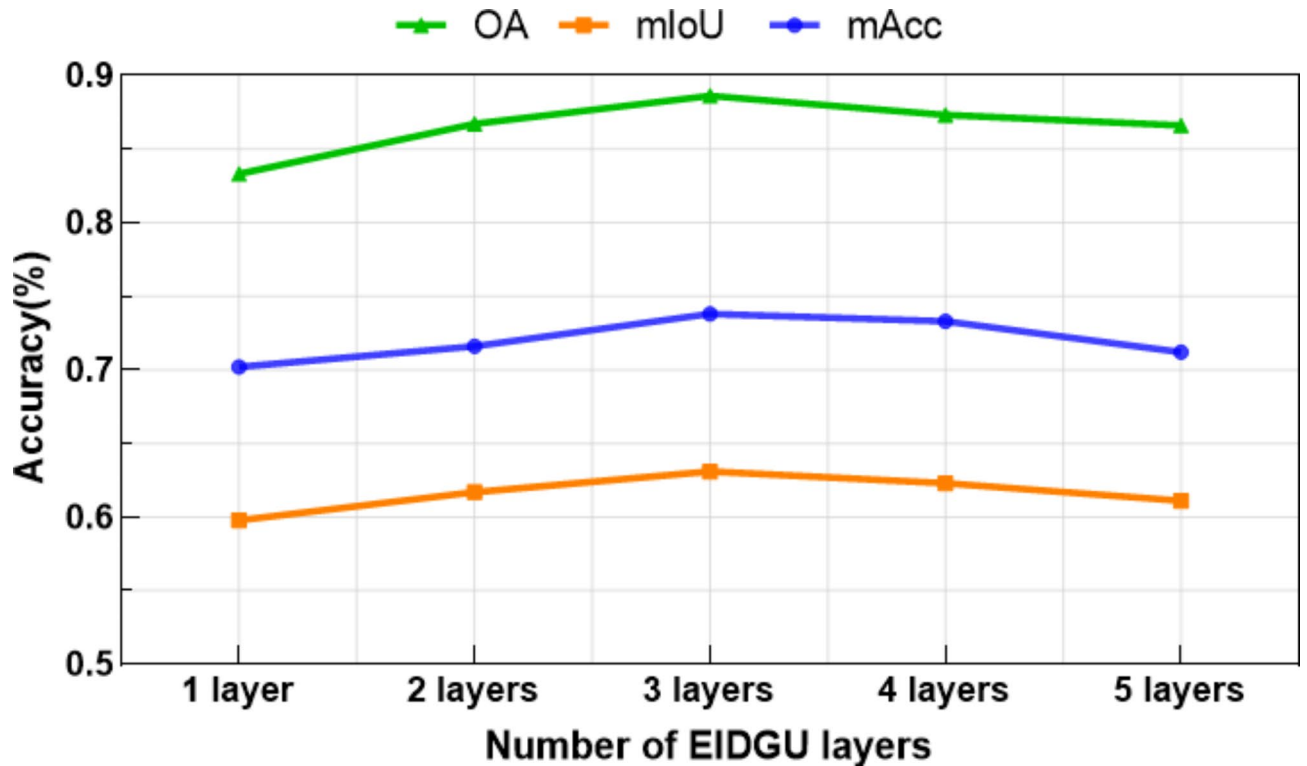


Fig. 3. Ablation Study on different numbers of EIDU-Net layers.

and arms of the original Terracotta Warriors models, dividing them into eight parts: head, body, left hand, left arm, right hand, right arm, left leg, and right leg.

Our experimental results in Table 5 show that the OA can reach the best effect of 89.7%, the mIoU can reach 70.3% and the mAcc can reach 84.2%. From Fig. 4(b), we can see the eight parts of the Terracotta Warriors are segmented well, but the segmentation boundaries of some contacting parts are not clear enough (as shown in the red circles). The reason for this problem is that the unique manufacturing process of the Terracotta Warriors causes some parts to stick together, which leads to the incompleteness of the Terracotta Warriors models. For example, the inner arm parts and body parts of some Terracotta Warriors models are adherent, which affects the final training results. Subsequently, we consider optimising the point cloud segmentation model jointly with the point cloud completion method. For example, ASFNet⁴⁴ achieves high-quality point cloud completion through an asymmetric twin feature matching mechanism and a multi-scale feature extraction module, providing richer geometric and semantic information for the subsequent segmentation task.

Overall, under conditions of segmenting models into more categories and with incomplete parts in some models across the dataset, our method achieves good performance on semantic segmentation, which demonstrates the robustness and efficacy of EIDU-Net.

Discussion

The proposed EIDU-Net model is used for supervised learning of point cloud segmentation and achieves good results on the S3DIS and the Terracotta Warriors dataset. As graph structures can better represent irregular data such as point clouds, this paper solves the problem that traditional GCN cannot iterate multiple layers by using DenseGCN, and learns multi-scale features through different dilation scales to obtain more detailed and deeper semantic feature information, so that GCNs can process large-scale and more complex graph data; the EGP and EGU proposed in this paper aggregate the feature information of edge points in the original point cloud to global features, reduce the loss of feature information in pooling and unpooling operations, and solve the problem of being unable to continue GCN; U-shaped structure is designed to fuse low-level and high-level features, and the performance and accuracy of the model are improved in segmentation tasks. Ablation studies prove that although increasing the number of network layers can improve feature learning to some extent, too many layers will lead to a gradual decline in performance due to overfitting. Therefore, while deepening the model architecture, we should also find ways to optimize the effectiveness of each individual feature extraction module.

Methods

This section introduces the proposed Edge-preserved Inception DenseGCN U-Net (EIDU-Net), which is designed as an encoder-decoder structure. Inception DenseGCN is set as the feature extraction module and some adjustments are made. In particular, two kinds of skip connections that work with the feature extraction module are utilized to ensure the learning of multi-scale and hierarchical features of 3D points. To adapt to

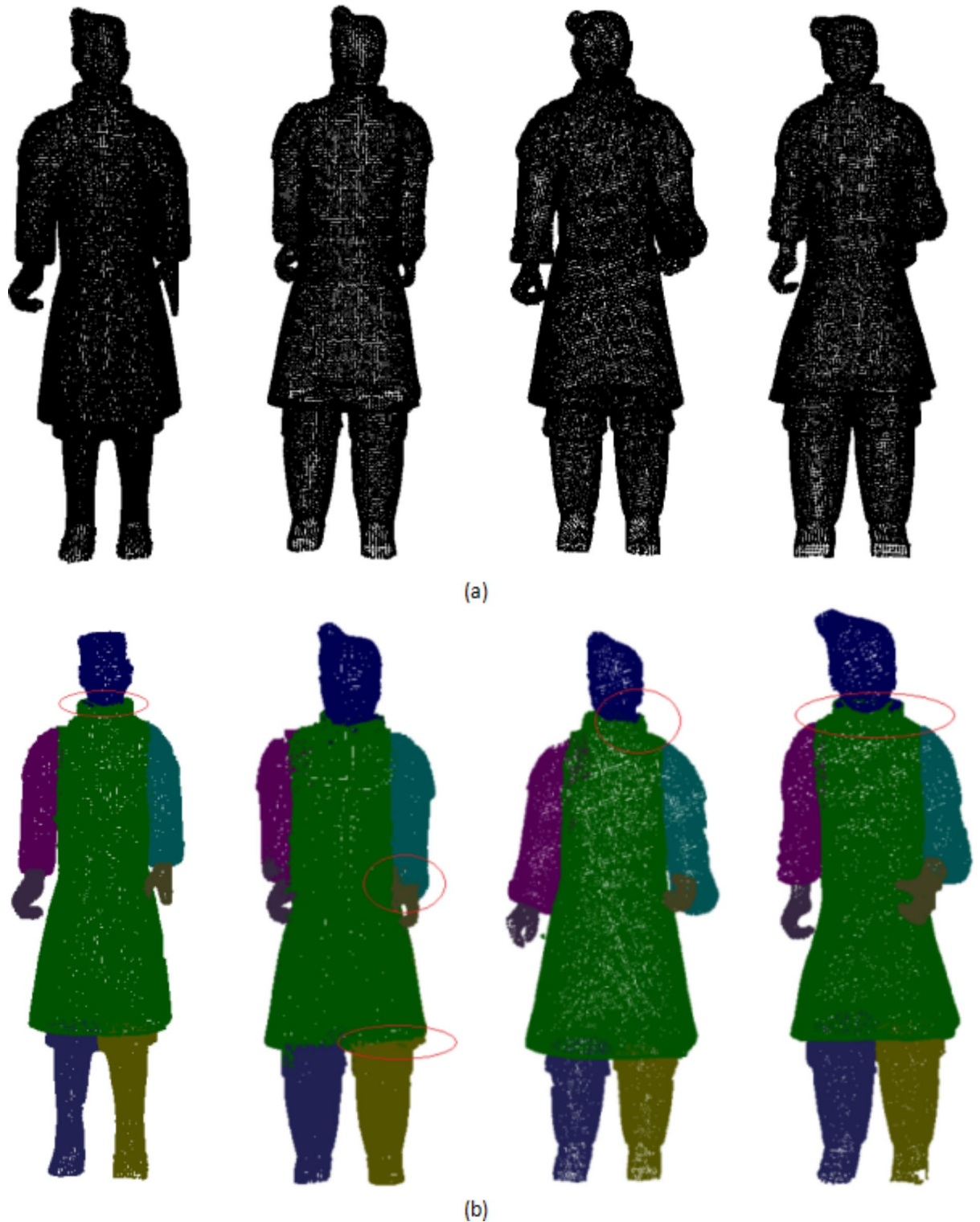


Fig. 4. Visualization results on Terracotta Warriors. (a) Input point clouds. (b) Ours.

the encoder-decoder structure and reduce the loss of feature information, the feature aggregation modules are designed as Edge-preserved Graph Pooling (EGP) and Edge-preserved Graph Unpooling (EGU). EGP is on the basis of the graph pooling layer and adds the operation of aggregating edge features of central nodes and neighbor nodes during the pooling operation. In subsequent operations, it reduces the loss of feature information due to pooling operations with skip connections. Similarly, in the EGU operation process, unpooling is performed better based on the previously saved information. EIDU-Net propagates feature information at different levels

Method	Input	OA	mAcc	mIoU
PointNet	Point	80.4	75.7	59.2
JSNet++	Point	87.9	81.1	68.7
Ours	Point	89.7	84.2	70.3

Table 5. Results on the terracotta warrior dataset. Significant values are in bold.

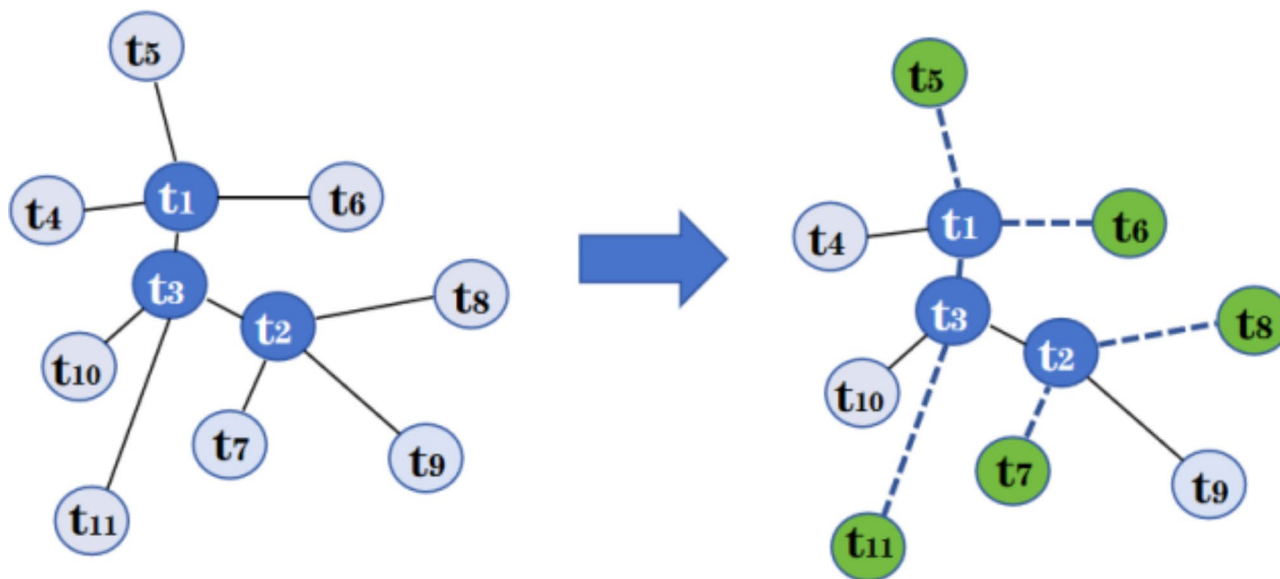


Fig. 5. Illustration of the EGP modules.

to the end through two different skip connections, ensuring the aggregation effect of local geometric details and high-level features.

Edge-preserved graph pooling

Pooling operation was first widely used in convolutional neural networks (mainly including max pooling and average pooling), and was later gradually implemented in graph convolutions. The difference is that the pooling operations of the CNN framework cannot be applied to graph convolution networks, which would destroy the spatial structure of the feature graph. The proposal of Graph U-net³¹ presented a classic pooling method suitable for graph convolution. The graph pooling layer incorporated a top-k scoring mechanism into the pooling process to ensure important feature nodes not lost.

Unlike CNN pooling operations, graph convolution pooling operations did not lose spatial information. In the Inception DenseGCN module⁴⁵, the original framework only had one layer of the feature extraction module. It directly adopted the multi-branch structure similar to CNN to integrate the feature information learned. Finally, the max pooling layer and two MLP layers were performed to obtain a fixed-length global feature for the classification and segmentation task.

The reasons for optimizing the pooling layer are: (1) to further deepen the network hierarchy of the encoder-decoder structure. The features after max pooling layer in the original method lose a lot of spatial information and cannot continue the next GCN operation; (2) because classical pooling operations for graph convolutions, such as Graph Pooling operation, also lose some feature information during the normal convolution process. We hope to optimize this step to reduce the loss of feature information.

As shown in Fig. 5, the left figure shows traditional graph pooling based on the top-k algorithm $\{t_1, t_2, t_3\}$. The right figure shows our EGP. Compared with the traditional method, after selecting k nodes by top-k, EGP constructs local neighborhood graphs for these k nodes from the original point cloud data, and selects their neighbors by random downsampling $\{t_5, t_6, t_7, t_8, t_{10}, t_{11}\}$. In this way, our model can better preserve local geometric information.

To solve this problem, we propose an optimized graph pooling method Edge-preserved Graph Pooling (EGP), which can more effectively capture local geometric details. EGP adds a novel process operation on the basis of original graph pooling to aggregate the features of high-scoring central nodes and their edge nodes. We place the EGP module after two DenseGCN layers with different dilation scales. In this layer, we first divide and select nodes in the feature graph through the top-k algorithm of graph pooling, and obtain k top-scoring nodes in descending order of scores, as illustrated in Fig. 6. We set these k nodes as central nodes, and construct their local neighborhood graphs respectively with the original point cloud data. Then the density of edge nodes in local neighborhood graphs is reduced by random downsampling, as shown in Fig. 6(a). Finally, the features of

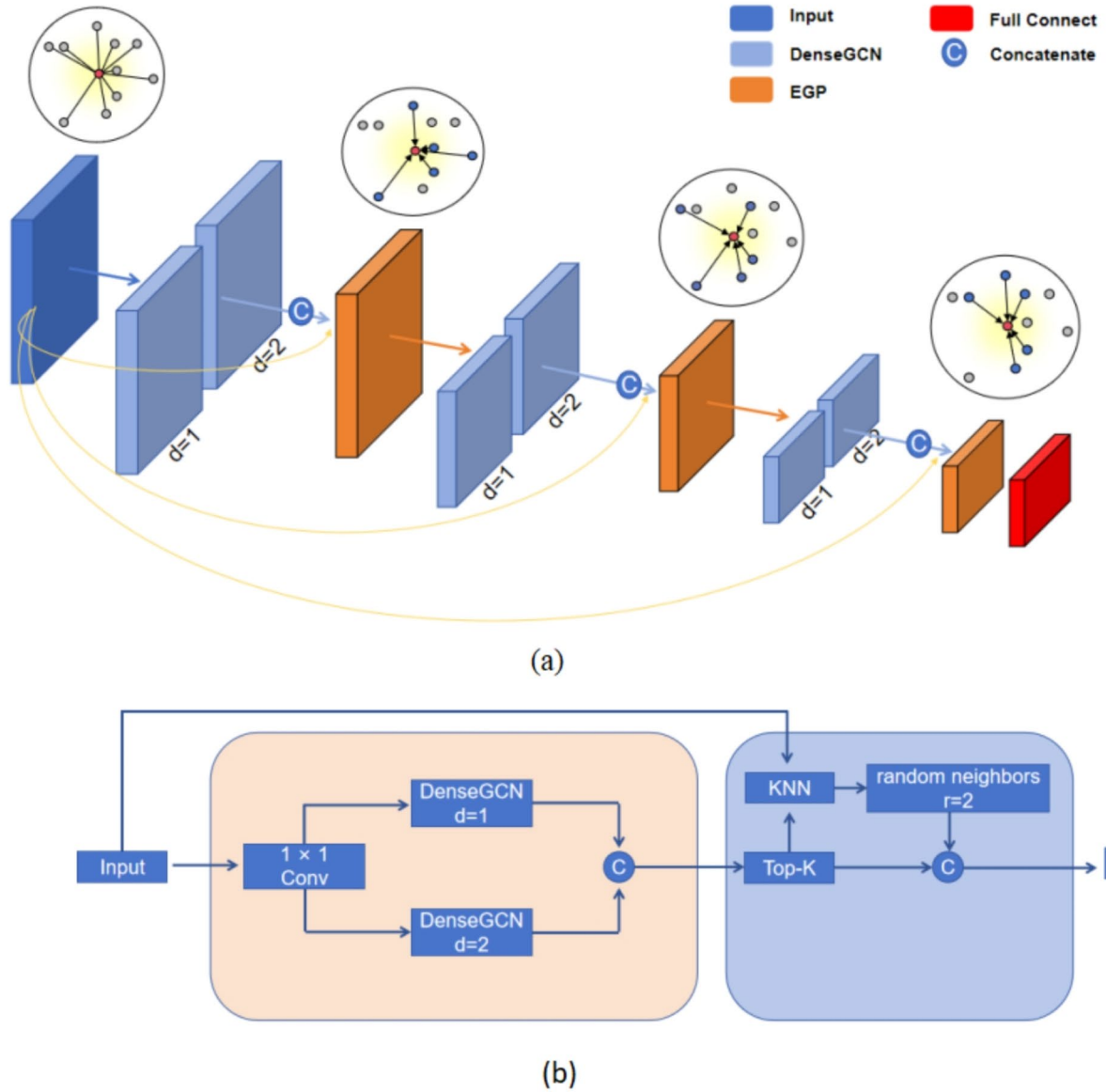


Fig. 6. Illustration of the EGP module. (a) The flow of the encoder module. The circular graph indicates that in the EGP process. (b) Illustration of the downsampling module for point cloud segmentation. d is the dilation rate and r is the random downsampling rate of EGP.

the central nodes and the remaining edge nodes are aggregated. In this way, the feature of top-k key nodes can be better retained, and their local structure information can be better aggregated.

Through EGP, the central node and its edge nodes are combined into an enhanced node and the local feature information of each central node can be retained better. Our EGP module is designed as:

$$\begin{cases} Score = X^\ell P^\ell / \|P^\ell\| \\ idx = rank(Score, k) \\ X'_p = X_p \oplus h(X_j^1, \dots, X_j^i) \\ Score = sigmoid(y(idx)) \\ \tilde{X}^\ell = X^\ell(idx, :) \\ X^{\ell+1} = \tilde{X}^\ell \odot (\widetilde{Score1}_C^T) \end{cases} \quad (4)$$

where k is the number of nodes that are selected by top-k arithmetic. The idx and $X^\dagger(idx, :)$ are the indices and feature matrices of the selected points, respectively, which are used to construct the new graph structure. X_p is the feature of the selected node and X_j^i is the feature of its selected neighbors by random downsampling.

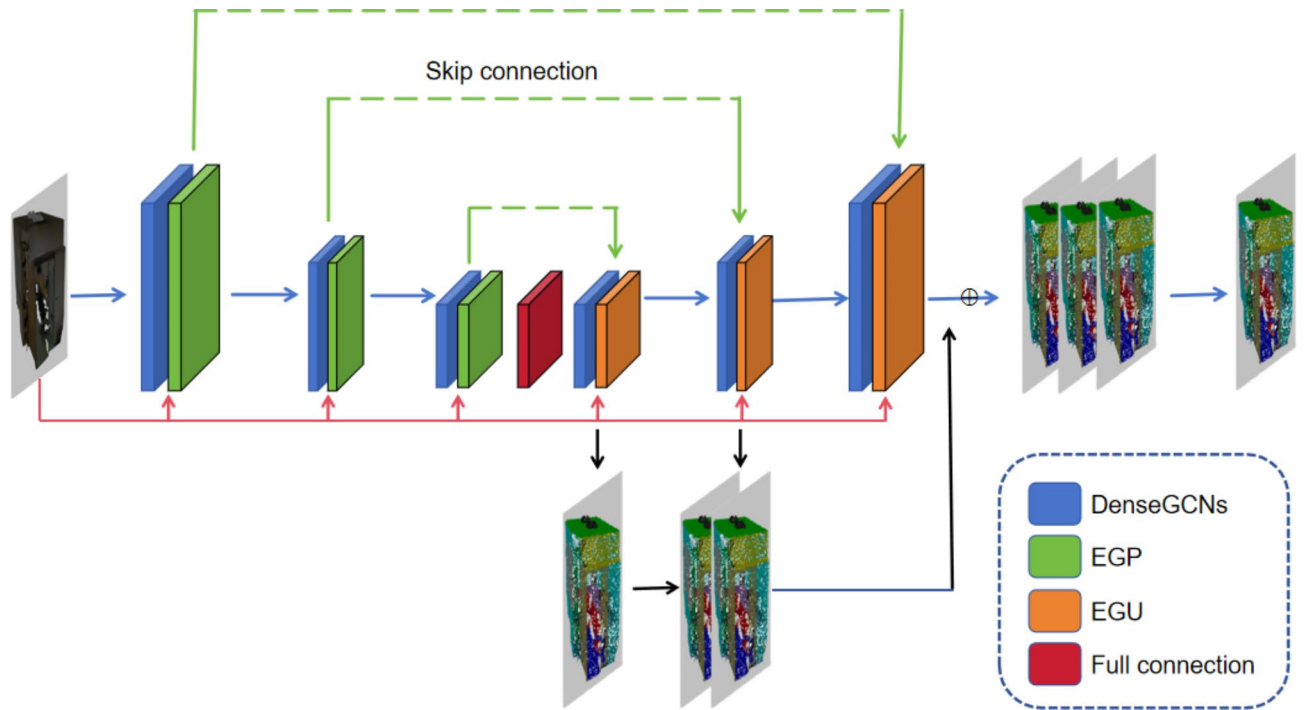


Fig. 7. Illustration of the EIDU-Net modules.

Edge-preserved Graph UnPooling

In the encoder-decoder structure, the low-layer 3D point features need to be restored to the corresponding feature density. Corresponding to the improvement of EGP pooling, Edge-preserved Graph Unpooling (EGU) also sets the nodes in the feature graph as central nodes during the graph unpooling operation, and also constructs their local neighborhood graphs with the original point cloud data. After randomly reducing the edge nodes in the neighborhood graphs through random downsampling, the feature information of the central nodes and the remaining edge nodes are aggregated. In the process of gradual upsampling, the central nodes and edge nodes are concatenated to enhance the features through locally detailed features, which improves the expressive ability of the model. Our EGU module is designed as:

$$X'_q = X_q^f \oplus \omega \left(X_{qj}^1, \dots, X_{qj}^i \right) \quad (5)$$

where X_q^f is the feature from corresponding layers of the encoder by skip connection and $\omega (\cdot)$ is the inverse distance weighted average operation.

Edge-preserved Inception DenseGcn U-Net

Inspired by the encoder-decoder structure, our model contains downsampling part as encoder and an upsampling part as decoder. As shown in Fig. 7, the encoder-decoder structure combining EGP and EGU is the outer nested structure in EIDU-Net. The encoder part consists of three layers, which gradually extract point features while performing pooling operations through EGP. Correspondingly, the decoder part consists of similar structure. Each layer of the decoder part can better aggregate local feature information through skip connections with the corresponding encoding layer. In addition, the black line shows that the three modules of the encoder separately recover feature density layer-by-layer through the EGU module and aggregate information at the end. The red lines indicate that the original point cloud data provides neighborhood graph information of the selected central points to the EGP module and EGU module.

Conclusion

The EIDU-Net model proposed in this paper plays an important role in promoting the research of point cloud segmentation that I know of, and provides new ideas for point cloud segmentation methods. By using deepGCN to extract the topological feature of points, it not only obtains the information of individual points and the relationship between points, but also expresses the feature information of point clouds more accurately and meticulously; in each pooling (EGP) and unpooling (EGU) operation, the top-k strategy is used to obtain k center points and random neighborhood information aggregation, which not only ensures the learning efficiency of the model, but also reduces the loss of geometric information; the U-shaped structure is utilized to fuse low-level and high-level features to further improve the segmentation accuracy. The experimental results verify the superiority and rationality of the proposed model. In the future, we can try to utilize the U2Net model to further enhance

the accuracy of the segmentation results and apply the optimized model to more fields, such as the field of autonomous driving. In addition, we can also explore joint training methods for point cloud completion (ASFM-Net for instance) and segmentation tasks, so that the completion and segmentation processes can promote each other and improve the overall performance for the more common scenes with occlusion. For example, a joint loss function can be designed to optimize both the completion and segmentation tasks, achieving more efficient model training.

Data availability

The datasets analyzed during the current study available from the corresponding authors on reasonable request.

Received: 8 January 2024; Accepted: 27 September 2024

Published online: 19 October 2024

References

- Ouyang, Z., Liu, Y., Zhang, C. & Niu, J. A cgans-based scene reconstruction model using lidar point cloud. In *IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)*, 1107–1114 (2017). (2017).
- Ji, H. & Luo, X. 3D scene reconstruction of landslide topography based on data fusion between laser point cloud and UAV image. *Environ. Earth Sci.* **78**, 1–12 (2019).
- Zhou, T., Zhu, Q. & Du, J. Intuitive robot teleoperation for civil engineering operations with virtual reality and deep learning scene reconstruction. *Adv. Eng. Inform.* **46**, 101170 (2020).
- Yue, X., Wu, B., Seshia, S. A., Keutzer, K. & Sangiovanni-Vincentelli A. L. A lidar point cloud generator: from a virtual world to autonomous driving. In *Proceedings of the ACM on International Conference on Multimedia Retrieval*. 458–464 (2018). (2018).
- Li, N., Liu, C. & Pfeifer, N. Improving LiDAR classification accuracy by contextual label smoothing in post-processing. *ISPRS J. Photogrammetry Remote Sens.* 13–31 (2019).
- Niemeyer, J., Rottensteiner, F. & Soergel, U. Contextual classification of lidar data and building object detection in urban areas. *Isprs J. Photogrammetry Remote Sens.* 152–165 (2014).
- Weinmann, M., Jutzi, B. & Mallet, C. Semantic 3D scene interpretation: a framework combining optimal neighborhood size selection with relevant features. *ISPRS Annals Photogrammetry Remote Sens. Spat. Inform. Sci.* 181–188 (2014).
- Maturana, D., Scherer, S. & VoxNet: A 3D Convolutional Neural Network for real-time object recognition. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 922–928 (2015).
- Le, T., Duan, Y. & Pointgrid A deep network for 3d shape understanding. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9204–9214 (2018).
- Zhu, X. et al. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9939–9948 (2021).
- Ji, J. et al. Encoder-decoder with cascaded CRFs for semantic segmentation. *IEEE transactions on circuits and systems for Video Technology*. 1926–1938 (2020).
- Chen, Q. et al. Spatialflow: bridging all tasks for panoptic segmentation. *IEEE Trans. Circuits Syst. Video Technol.* 2288–2300 (2020).
- Sun, X. et al. Gaussian dynamic convolution for efficient single-image segmentation. *IEEE Trans. Circuits Syst. Video Technol.* 2937–2948 (2021).
- Lawin, F. J. et al. Deep projective 3D semantic segmentation. *Computer Analysis of Images and Patterns: 17th International Conference, CAIP 95–107* (2017). (2017).
- Boulch, A., Le Saux, B. & Audebert, N. Unstructured point cloud semantic labeling using deep segmentation networks. *3dor@Eurographics 17–24* (2017).
- Qi, C. R. et al. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 652–660 (2017).
- Qi, C. R. et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*. 30 (2017).
- Zhao, H. et al. Pointweb: Enhancing local neighborhood features for point cloud processing. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5565–5573 (2019).
- Li, Y. et al. Pointcnn: Convolution on x-transformed points. *Adv. Neural. Inf. Process. Syst.* 31 (2018).
- Lai, X. et al. Stratified transformer for 3d point cloud segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8500–8509 (2022).
- Zhao, H. et al. Point transformer. *Proceedings of the IEEE/CVF international conference on computer vision*. 16259–16268 (2021).
- Wu, W., Qi, Z., Fuxin, L. & Pointconv Deep convolutional networks on 3d point clouds. *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*. 9621–9630 (2019).
- Qian, G. et al. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in neural information processing systems*. 23192–23204 (2022).
- Zhang, R. et al. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *Advances in neural information processing systems*. 27061–27074 (2022).
- Xia, Y. et al. Casspr: Cross attention single scan place recognition. *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8461–8472 (2023).
- Xia, Y. et al. SOE-Net: A self-attention and orientation encoding network for point cloud based place recognition. *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*. 11348–11357 (2021).
- Huang, W. et al. OPOCA: one point one class annotation for LiDAR Point Cloud Semantic Segmentation. *IEEE Trans. Geosci. Remote Sens.* 1–10 (2024).
- Shen, S. et al. SegTrans: semantic segmentation with transfer learning for MLS point clouds. *IEEE Geosci. Remote Sens. Lett.* (2023).
- Wang, Y. et al. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graphics (tog)* 1–12 (2019).
- Li, G. et al. Deepcgns: Can cgns go as deep as cnns? *Proceedings of the IEEE/CVF international conference on computer vision*. 9267–9276 (2019).
- Gao, H. & Ji, S. Graph u-nets. *International conference on machine learning*. 2083–2092 (2019).
- Armeni, I. et al. 3D semantic parsing of large-scale indoor spaces. *Comput. Vis. Pattern Recognit.* 1534–1543 (2016).
- Tchapmi, L. et al. Segcloud: Semantic segmentation of 3d point clouds. *2017 international conference on 3D vision (3DV)*. 537–547 (2017).
- Landrieu, L. & Simonovsky, M. Large-scale point cloud semantic segmentation with superpoint graphs. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4558–4567 (2018).
- Jiang, L. et al. Hierarchical point-edge interaction network for point cloud semantic segmentation. *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10433–10441 (2019).

36. Choy, C., Gwak, J. & Savarese, S. 4d spatio-temporal convnets: Minkowski convolutional neural networks. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3075–3084 (2019).
37. Pan, L., Chew, C. M., Lee, G. H. & PointAtrousGraph Deep Hierarchical Encoder-Decoder with Point Atrous Convolution for Unorganized 3D Points. *International Conference on Robotics and Automation*. 1113–1120 (2020).
38. Guo, M.-H. et al. Pct: point cloud transformer. *Comput. Visual Media* 187–199 (2021).
39. Lei, H., Akhtar, N., Mian, A. & Seggcn Efficient 3d point cloud segmentation with fuzzy spherical kernel. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11611–11620 (2020).
40. Thomas, H. et al. Kpconv: Flexible and deformable convolution for point clouds. *Proceedings of the IEEE/CVF international conference on computer vision*. 6411–6420 (2019).
41. Hu, Q. et al. Randa-net: Efficient semantic segmentation of large-scale point clouds. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11108–11117 (2020).
42. Chen, J., Kakillioglu, B. & Velipasalar, S. Background-aware 3-D point cloud segmentation with dynamic point feature aggregation. *IEEE Trans. Geosci. Remote Sens.* 1–12 (2022).
43. Zhao, L., Tao, W. & Jsnet++ Dynamic filters and pointwise correlation for 3d point cloud instance and semantic segmentation. *IEEE Trans. Circuits Syst. Video Technol.* 1854–1867 (2022).
44. Xia, Y. et al. Asfm-net: Asymmetrical siamese feature matching network for point completion. In *Proceedings of the 29th ACM international conference on multimedia*. 1938–1947 (2021).
45. Qian, G. et al. Pu-gcn: Point cloud upsampling using graph convolutional networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11683–11692 (2021).
46. Qin, X. et al. U2-Net: going deeper with nested U-structure for salient object detection. *Pattern Recogn.* **106**, 107404 (2020).

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (61701403, 61806164, 62271393); Key Research and Development Program of Shaanxi Province (2019GY215, 2021ZDLSF06-04); China Postdoctoral Science Foundation (2018M643717); Graduate Innovation Program at Northwest University (CX2023185).

Author contributions

Xueli Xu, Jingyu Wang and Qiuquan Zhu wrote the main manuscript text and carried out the experiments. Guohua Geng and Ping Zhou pre-process the experimental data. Linzhi Su and Xueli Xu designed the research. Xin Cao and Kang Li oversight and leadership responsibility for the research activity planning and execution. All authors reviewed the manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to K.L. or X.C.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2024