APPLICATION OF A SHELLNET BASED APPROACH TO SEMANTIC SEGMENTATION IN URBAN POINT CLOUD

Deliang Chen [1], Xuan Ma [1, *], Xinliang Lu [1], Jianbo Xiao [1]

[1] School of Geographic and Biologic Information, Nanjing University of Posts and Telecommunications, 210023, Nanjing, China- (dlchen, 1020173006, b18080125) @njupt.edu.cn, 1807180656@qq.com

Commission II, WG II/3

KEY WORDS: Point cloud, Semantic segmentation, ShellNet, Urban, Mobile laser scanning, 3D objects detection.

ABSTRACT:

In recent years, the popularity of airborne, vehicle-borne, and terrestrial 3D laser scanners has driven the rapid development of 3D point cloud processing methods. The 3D laser scanning technology has the characteristics of non-contact, high density, high accuracy, and digitalization, which can achieve comprehensive and fast 3D scanning of urban point clouds. To address the current situation that it is difficult to accurately segment urban point clouds in complex scenes from 3D laser scanned point clouds, a technical process for accurate and fast semantic segmentation of urban point clouds is proposed. In this study, the point clouds are first denoised, then the samples are annotated and sample sets are created based on the point cloud features of the category targets using CloudCompare software, followed by an end-to-end trainable optimization network-ShellNet, to train the urban point cloud samples, and finally, the models are evaluated on a test set. The method achieved IoU metrics of 89.83% and 73.74% for semantic segmentation of buildings and rods-like objects respectively. From the visualization results of the test set, the algorithm is feasible and robust, providing a new idea and method for semantic segmentation of large-scale urban scenes.

1. INTRODUCTION

3D laser scanning measurement has the characteristics of fast, accurate, and non-contact, which can directly obtain the 3D dense point cloud on the surface of the object, and plays a very important role in the point cloud extraction of large-scene urban roads (Pierdicca et al., 2020). Firstly, the 3D laser scanner is used to scan the target and the 3D point cloud data is obtained which is exactly consistent with the field size, and then builds a true 3D real-world model of the physical scene through data processing software, and its touchless, high-precision and high-efficiency scanned scene data provides strong data support for the recently proposed smart city brain infrastructure (Han et al., 2016). Due to the complex and large scale of urban scenes, the acquired laser point cloud data often have the problems of large data volume, discrete type, serious noise, and loopholes, etc. Therefore, how to process urban road point cloud data quickly and at a high level is the current challenge to be solved (Duan et al., 2019).

The MLS (Mobile Laser Scanning) system is used to scan urban areas and the resulting high-density point cloud contains various types of objects such as buildings, street lights, trees, etc. (Lari et al., 2011). In the existing studies, the geometric information of the point cloud is mostly used to identify various target features in the scene (Huang et al., 2019). Fewer studies have attempted to use color point clouds for urban scene analysis.

Point cloud segmentation divides point cloud data according to certain rules, usually by labeling points with the same characteristics as the same class. 3D point cloud segmentation methods have been developed for a long time, and a large number of traditional classical segmentation algorithms have emerged, which can be mainly classified into the following categories: there are edge-based methods, region-based methods, model-based methods, graph-based methods, and attributes-based methods, etc. Edge-based (Himmelsbachc et al., 2009) segmentation algorithms filter boundary points by geometric features of the point cloud, then connect the filtered boundary points to form boundary lines and finally segment the point cloud surface area into independent point sets according to the boundary lines. Region-based (Dong et al., 2018) methods group points with similar geometrically defined properties into a plane by selecting seed points, while continuously correcting the feature parameters of the seed region fitted surface until there are no points that still satisfy the threshold condition. Model-based (Schnabel et al., 2007) approach uses the mathematical parametric model of simple geometric tuples as the most a priori information to classify the point cloud into the corresponding tuples category. Graph-based (Yang et al., 2014) segmentation approach treats the point cloud data as vertices, constructs edges using the spatial neighborhood relationship of the points, and constructs a graph by weighting the connected edges using the similarity of the neighborhood points. Attributes-based (Filin, 2002) approach is the geometric structure features or spatial distribution features exhibited by the point cloud are used to cluster the features of the point cloud to achieve segmentation. However, due to the noise points, object occlusions, and uneven acquisition density of the point cloud data we obtain, these methods are difficult to fit onto the object (Nguyen et al., 2013), which greatly affects the accuracy.

According to the 3D point cloud data processing method, the 3D point cloud semantic segmentation methods based on deep learning are divided into two categories, namely direct semantic segmentation methods and indirect semantic segmentation methods (Qi et al., 2017). The former is to extract feature information directly from point cloud data, and the architecture retains the intrinsic information within the original points to predict point-level semantics without transformation to voxels and multi-views (Su et al., 2015); The latter converts the original

* Corresponding author

point cloud data into a regular 3D voxel mesh or multi-view, indirectly extracting features from the 3D point cloud data by means of data transformation and completing the segmentation. The features are indirectly extracted from the 3D point cloud data by means of data transformation for semantic segmentation purposes. The significant development of the point cloud data processing algorithm, although the accuracy rate in the scene segmentation task has been achieved, the training speed is slower and the network structure is complex. For example, PointCNN weights and displaces the input features at the same time, and then applies a typical convolution, but the convergence rate is slower; Pointwise uses point-by-point convolution to obtain the local features of the points, using voxel positioning weights to make it inflexible. The ShellNet algorithm used in this paper uses efficient ShellConv convolutional operators to directly process large-scale data sets. Since the neural network has fewer parameters, it can maintain a very fast training speed, and the experimental results also ensure the effectiveness of the network.

In summary, this study proposes a technical process for segmenting target point clouds in urban scenes based on the elevation, intensity, and geometry of the point clouds, with respect to the characteristics of various target point clouds in complex urban scenes. The experimental data verified that the technical flow has a good segmentation effect and improves the automation of the segmentation of urban scenes.

2. METHODOLOGY

2.1 Point Cloud Denoising

In recent years, the availability of point cloud data has been increasing. When point cloud data is directly obtained from the MLS system, the inaccuracy of deep acquisition will cause the point cloud to be noisy and may contain many outliers (Javaheri et al., 2017). Point cloud denoising, as the first step in data preprocessing, has a relatively large impact on the follow-up and is therefore required in this study.

Based on the property that outlier points will move away from their neighbors, this study uses radius outlier removal, where each point is connected to its neighbors within the radius with a small graph (Schoenenberger et al., 2015). A threshold of the minimum number of neighbors within the neighborhood of the radius is set up to identify outliers.

2.2 ShellNet Network Structure

In recent years the field of point cloud research, it has been a research hotspot on how to perform efficient feature computation for unstructured data like point clouds (Chen et al., 2020). This study uses an algorithm for segmenting urban scenes-ShellNet (Zhang et al., 2019). To achieve an efficient point cloud neural network, a convolution that can directly use point clouds needs to be defined. ShellConv is the core part of ShellNet network to obtain features of local point sets. The main idea of ShellConv is to output a deeper sparse point set by merging point sampling into the convolution (Joshi et al., 2021). The function implemented by ShellConv is to calculate the characteristics of the sample point. The input point cloud is randomly sampled to form a set of points centered on the representative points, distributed on these spherical shells, and then the local characteristics of the layer shell are derived by maxpooling. Finally, the characteristics of the sampling point are obtained by the local characteristics of multiple shells. This is shown in Figure 1. In this method, although the number does not increase, a larger acceptance area can be obtained. A set of representative points is randomly selected from the input point set, for a particular representative point p, its neighbor q is obtained by the nearest neighbor method, then the convolution on point p is

$$F(p)^{(n)} = \sum_{q \in \Omega_p^{(n)}}^{n} w(q)^{(n)} F(q)^{(n-1)} \quad (1)$$

where F represents the input characteristics of the point set for a particular channel, W is the weight of the convolution. The superscript (n) is used to indicate the parameters of the n layer. F(p) and F(q) denote the characteristics of point p and point q.

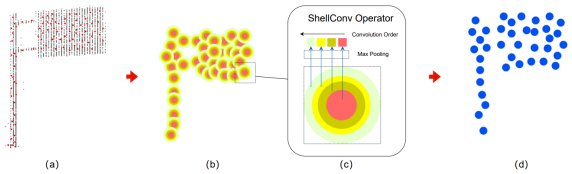

Figure 1. ShellConv. (a) The red dot is a random sampling point. (b) The set of points centered on the sampling point is distributed on the spherical shell. (c) The characteristics of these points are enhanced by maxpooling. (d) Output features.

ShellConv is used in ShellNet instead of the traditional 2D convolution. The segmentation network follows U-net, which is a classical full convolutional network that can combine local information and global information (Zhang et al., 2018). The deconvolution part starts from the set point of N_2 in Figure 2. Through the three-layer ShellConv operator, the output points of the deconvolution layer gradually increase, but the characteristic channels gradually decrease, until the points upsampled are the same as the number of input points N.

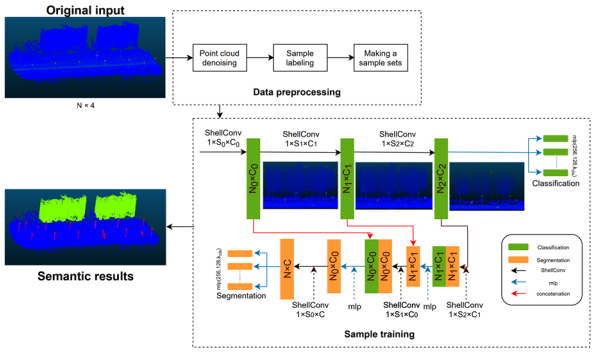

Figure 2. Technology Roadmap. For the input point cloud, pre-processing is first performed, including point cloud denoising, sample set labeling, and generation, where N is the number of raw point clouds, and XYZ coordinates and intensity are the four feature inputs for the point. Entering the ShellNet network, through three layers of ShellConv, a matrix of size $N_2 \times C_2$ is obtained, where N_2 is the number of representative points that are finally extracted from the input point cloud. Each point contains a high-dimensional feature vector of size C_2. This matrix is entered into the mlp module, size (256, 128), to generate a probability plot for object classification.

2.3 PointNet++ Network Structure

PointNet (Qi et al., 2017) is a pioneering effort that directly processes point sets. The main idea of PointNet++ is to add a multi-level feature extraction structure to PointNet, which is to divide the input point cloud into several local point sets, and

extract the global features of each point set, then make the features continuously abstracted, so as to obtain higher-level features, each set is called set abstraction. Each set abstraction consists of three parts: the sampling layer, the grouping layer, and the PointNet layer (Yao et al., 2019). In the sampling layer, FPS (farthest point sampling) is used to collect the centroids; in the grouping layer, KNN is used to find the k nearest points around the centroids to form the local area; finally, PointNet is used to extract the local features from each local area given by the grouping layer.

For the segmentation task, each point is given a corresponding class label, that is, the set of points is restored to the original data, which is done mainly by interpolation and hopping connections. The interpolation is a weighted average of the inverse of the distances of the k nearest neighbors. The jump join is the stitching of the output features obtained from each of the previous set abstraction layers with the features of the interpolated points (Ma et al., 2022). As the obtained feature dimension is too high, which will affect the training speed and training effect, it will go through unit PointNet to reduce the feature dimension and improve the robustness of the model. This process is repeated until the features are propagated to the original set of points.

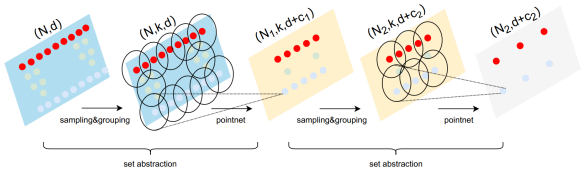

Figure 3. Set abstraction. N is the number of input points, d is the coordinate dimension of points, and c is the intensity.

3. EXPERIMENT AND DISCUSSION

In this section, the efficiency and effectiveness of our solution for segmenting urban targets from MLS point clouds are investigated and discussed. Note that all experiments are performed on the same workstation with an Intel Gold 6130 @2.7GHz CPU and an NVIDIA RTX3090 GPU. During the training process, the initial learning rate is set to 0.001, and each iteration will be 0.7 times the original.

3.1 Dataset

In order to fully verify the feasibility and robustness of the algorithm in this paper, Nanjing Olympic Sports Center(In the WGS84/UTM coordinate system, the x-coordinate of the dataset is between 661650.06 ~ 666158.03m and the y-coordinate is between 3541576.99 ~ 3545957.31m.) was used as the experimental object in this study, as shown in Figure 4. The training data required for the experiments were scanned by a Lynx SG1 vehicle-mounted scanner released by Optech of Canada, containing relatively fine details covering a wide variety of urban scenes: apartments, gymnasiums, offices, buildings under construction, street lights, utility poles, billboards, etc. As Figure 5 shows, the data set is displayed in terms of height.

In total, the dataset consists of more than 60 million 3D points and contains 32 labeled urban scenes. Each scene has up to 10^8 points with XYZ coordinates and intensity information. This research proposes a set of technical processes for semantic segmentation, which is practically applicable in urban scenes. Whether it is a smart city or a modern industrial application, the point cloud data acquired is massive. In order to be able to realistically reflect the accuracy of this research method, we, therefore, chose to use this dataset. In addition, the small amount of data can lead to overfitting, which in turn affects the training results.

The data are manually labeled into three semantic categories, including buildings, rods-like objects, and others.

(1) Buildings: apartments, gymnasiums, offices, buildings under construction, etc.
(2) Rods-like objects: street lights, utility poles, billboards, street signs, etc.
(3) Others: objects that do not belong to the previously mentioned classes.

To verify the segmentation performance, 26 of these 32 scenes are randomly selected as the training set and the remaining 6 as the validation set in this paper.

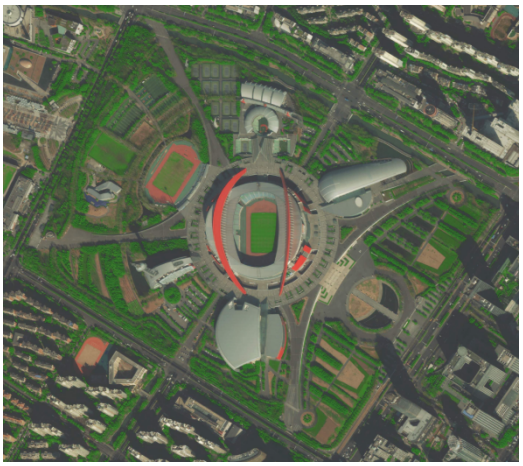

Figure 4. Top view of Nanjing Olympic Sports Center.

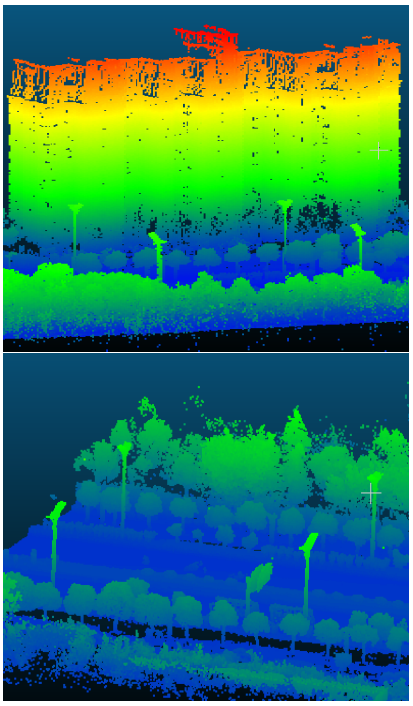

Figure 5. Point cloud colorized by height.

3.2 Evaluation

In order to evaluate the segmentation results, after the point clouds for each category have been extracted, the accuracy is assessed using an accuracy evaluation method. Overall Accuracy is a commonly used metric in multi-category segmentation problems but can be affected by uneven sample distribution. In order to more scientifically assess the effectiveness of this paper's method for each category of segmentation, specifically precision, recall, F-score, and IoU metrics will be used as evaluation metrics for comparison. In this study, we designate an object such as a building as a positive sample and denote it as TP if it is segmented correctly, or FN if it is segmented as other objects or rods-like. Rods-like are denoted as FP if they are segmented as buildings, and rods-like are denoted as TN if they are segmented correctly. Based on the above metrics, the required accuracy evaluation value can be calculated as follows.

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F} - \text{Score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{4}$$

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{5}$$

For precision and recall, the two are not necessarily correlated. However, in the real world, these two metrics can exhibit mutual constraints due to overly large data sets. In this study, we need to weigh these two metrics together and therefore include the F-score as an evaluation metric. IoU is generally calculated based on categories, that is, the IoU of each category is calculated and then accumulated and averaged to obtain a global-based evaluation, which has been used as a standard metric in semantic segmentation.

3.3 Results and Discussion

In this study, semantic segmentation experiments were conducted on the constructed outdoor point cloud data using the ShellNet network, and the urban scene of Nanjing Olympic Sports Center was mainly selected and the segmentation results of this scene were visualized. In order to demonstrate the superior performance of the target segmentation algorithm proposed in this paper in terms of category segmentation, a PointNet++ network was used for comparison experiments, and the segmentation results are shown in Figure 6. The experimental results show that PointNet++ achieves 99.39%, 99.48% and 99.42% in terms of recall, precision and F-Score, respectively. However, the results of ShellNet are slightly higher with 99.53%, 99.58%, and 99.54%. When comparing the IoU indices, ShellNet and PointNet++ achieve 73.74% and 68.39% for rods-like objects and 89.83% and 83.87% for building objects, respectively. In summary, it can be seen from Table 2 that ShellNet outperforms PointNet++ in general, obtaining accurate segmentation, which illustrates its superior performance in segmenting urban scenes.

In order to get a better impression of the effectiveness of ShellNet on large scale data sets, this paper has selected scenes from six validation sets with good results and compared the segmentation results of ShellNet, PointNet++ and real ground scenes. As shown in Figure 6 for building 1, building 2 and building 3, it is clear that PointNet++ has a large error in segmenting buildings. ShellNet's segmentation results do not have this large error, except for building 2, where there is a significant mis-segmentation, but are basically the same as the real ground scene. As shown in Figure 6 for rods-like 1, rods-like 2, rods-like 3, PointNet++ also has many misclassifications when segmenting rods-like, misclassifying part of the point cloud on a rods-like as a building, in rods-like 2 ShellNet misclassifies the upper part of the point cloud on the rods-like as a building, in rods-like 3 classifies the rods-like into other classes. The specific accuracy evaluation values are shown in Table 1. In the fifth and sixth scenes, the accuracy evaluation values for ShellNet rods-like are lower than those of Pointnet++, which we suspect may be due to the small number of samples of rods-like in these two scenes, but this problem does not occur in the other scenes. Overall, the segmentation of PointNet++ is numerically good, but its visualisation results show a lot of errors, which is particularly evident in the comparison with ShellNet.

Method	Validation set	The evaluation index					
		Precision(%)	Recall(%)	F-Score(%)	IoU_a(%)	IoU_b(%)	IoU_c(%)
PointNet++	1	99.44	99.40	99.41	99.35	95.23	72.98
	2	99.11	99.12	99.11	99.10	76.96	54.03
	3	99.24	99.24	99.24	99.11	97.51	62.62
	4	99.58	99.56	99.56	99.55	87.11	80.51
	5	99.74	99.61	99.61	99.61	62.52	63.22
	6	99.76	99.38	99.38	99.41	0.00	77.00
ShellNet	1	99.80	99.79	99.79	99.77	98.12	90.77
	2	98.96	98.97	98.93	98.94	70.60	66.57
	3	99.50	99.50	99.50	99.50	98.83	68.20
	4	99.76	99.74	99.75	99.73	92.50	86.88
	5	99.75	99.59	99.64	99.59	78.90	55.21
	6	99.73	99.59	99.66	99.62	0.00	74.79

Table 1. In the six scenes divided into our data set, PointNet++ and ShellNet were used for segmentation experiments respectively, and the accuracy of their results was evaluated and compared with specific values

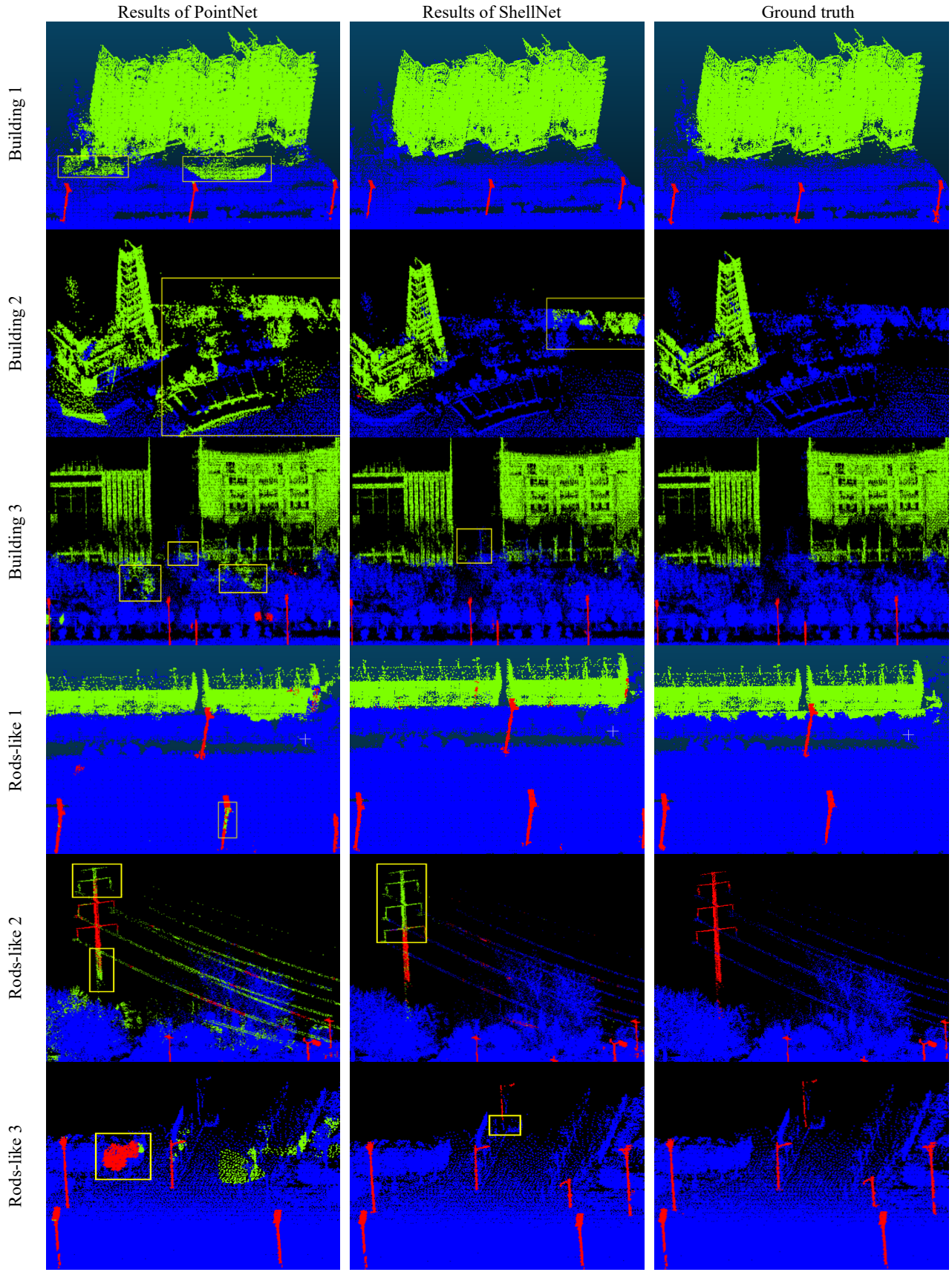

Figure 6. Test results for buildings, rods-like, and other objects. For obvious contrast, ground truth values and results are rendered in different colors, with buildings rendered in green, poles in red, and other objects in blue. Yellow rectangles show misclassified cases.

Method	Recall(%)	Precision(%)	F(%)	IoU_a(%)	IoU_b(%)	IoU_c(%)
PointNet++	99.39	99.48	99.42	99.36	83.87	68.39
ShellNet	**99.53**	**99.58**	**99.54**	**99.51**	**89.83**	**73.74**

IoU_a of other objects
IoU_b of buildings
IoU_c of rods-like

Table 2. Evaluation of the overall segmentation effect of PointNet++ and ShellNet on our dataset.

4. CONCLUSION

In order to minimize human intervention in the current situation where automatic semantic segmentation of complex urban scenes is difficult, this paper employs the ShellNet deep learning network for automatic semantic segmentation. The network is an end-to-end deep neural network for the point-by-point classification of outdoor large-scale point clouds, effectively segmenting the entire urban scene into three categories on our own dataset. Through semantic segmentation experiments on buildings, poles, and other objects, the results show that the research method in this paper is feasible and robust, and the accuracy of its test data meets the requirements in production activities. Compared to traditional methods, ShellNet and PointNet++, two deep learning methods, appropriately address the disorderly nature of point clouds and exploit the spatial relationships between points to aggregate information in a tandem fashion between local and global features. Compared to PointNet++, ShellNet's network model is a little more accurate, avoids misclassification, and can also correctly classify edge areas of buildings. The ShellNet model has fewer parameters and also outperforms PointNet++ in time, enabling fast classification of point clouds with large data volumes and is more suitable for point cloud classification in large-scale outdoor scenes.

In recent years, smart cities have become a strategic choice for promoting global urbanization, improving urban governance, and developing the digital economy. As a part of the smart city, architecture is one of the important carriers, and it is becoming more and more intelligent from the perspective of the building itself. Therefore, this study focuses on the division of buildings and rods-like. Due to the different characteristics of the distribution of point clouds and the intensity of various objects in the city, the next step is to select more algorithms for more complex urban scenes.

ACKNOWLEDGMENTS

This work was supported by the Natural Science Research Projects of Jiangsu Higher Education Institutions (No. 21KJB420004).

REFERENCES

Chen, S., Zhang, Z., Zhong, R., Zhang, L., Ma, H., Liu, L., 2020. A dense feature pyramid network-based deep learning model for road marking instance segmentation using MLS point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 59(1), 784-800.

Dong, Z., Yang, B., Hu, P., Scherer, S., 2018. An efficient global energy optimization approach for robust 3D plane segmentation of point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 137, 112-133.

Duan, C., Chen, S., Kovacevic, J., 2019. 3D point cloud denoising via deep neural network based local surface estimation. *In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8553-8557.

Filin, S., 2002. Surface clustering from airborne laser scanning data. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/A), 119-124.

Han, Y., Yang, B., Zhen, Y., 2016. Mobile 3D laser scanning technology application in the surveying of urban underground rail transit. *Earth and Environmental Science*, 46(1), 012057.

Himmelsbach, M., Luettel, T., Wuensche, H. J., 2009. Real-time object classification in 3D point clouds using point feature histograms. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009: 994-1000.

Huang, J., Zhang, X., Xin, Q., Sun, Y., Zhang, P., 2019. Automatic building extraction from high-resolution aerial images and LiDAR data using gated residual refinement network. *ISPRS journal of photogrammetry and remote sensing*, 151, 91-105.

Javaheri, A., Brites, C., Pereira, F., Ascenso, J., 2017. Subjective and objective quality evaluation of 3D point cloud denoising algorithms. *In 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 1-6.

Joshi, G., Walambe, R., Kotecha, K., 2021. A Review on Explainability in Multimodal Deep Neural Nets. *IEEE Access*.

Lari, Z., Habib, A., Kwak, E., 2011. An adaptive approach for segmentation of 3D laser point cloud. *In ISPRS Workshop laser scanning*, 29-31.

Ma, H., Ma, H., Zhang, L., Liu, K., Luo, W., 2022. Extracting Urban Road Footprints from Airborne LiDAR Point Clouds with PointNet++ and Two-Step Post-Processing. *Remote Sensing*, 14(3), 789.

Nguyen, A., Le, B., 2013. 3D point cloud segmentation: A survey. *In 2013 6th IEEE conference on robotics, automation and mechatronics*, 225-230.

Pierdicca, R., Paolanti, M., Matrone, F., Martini, M., Morbidoni, C., Malinverni, E. S., Frontoni E., Lingua, A. M., 2020. Point cloud semantic segmentation using a deep learning framework for cultural heritage. *Remote Sensing*, 12(6), 1005.

Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, 652-660.

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv*:1706.02413.

Schnabel, R., Wahl, R., Klein, R., 2007. Efficient RANSAC for point-cloud shape detection. Computer graphics forum. Oxford, UK: Blackwell Publishing Ltd, 2007, 26(2): 214-226.

Schoenenberger, Y., Paratte, J., Vandergheynst, P., 2015. Graph-based denoising for time-varying point clouds. In 2015 3DTV-Conference: *The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, 1-4.

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015: Multi-view convolutional neural networks for 3d shape recognition. *In Proceedings of the IEEE international conference on computer vision*, 945-953.

Yang, J., Gan, Z., Li, K., Hou, C., 2014. Graph-based segmentation for RGB-D data using 3-D geometry enhanced superpixels. *IEEE transactions on cybernetics*, 45(5), 927-940.

Yao, X., Guo, J., Hu, J., Cao, Q., 2019. Using deep learning in semantic classification for point cloud data. *IEEE Access*, 7, 37121-37130.

Zhang, Z., Hua, B. S., Yeung, S. K., 2019. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. *In Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1607-1616.

Zhang, Z., Liu, Q., Wang, Y., 2018. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, 15(5), 749-753.