Proceedings of the 40th Chinese Control Conference
July 26-28, 2021, Shanghai, China

Unmanned Vehicle 3D Lidar Point Cloud Segmentation

Rui Guo[1*], Zheyi Jiang[2], Rui Gao[1], Wenkun Yang[1], Yuxin Gao[2], Xiaofeng Chen[1], Yongfeng Zhi[1], Liang Guo[3]

1. School of Automation, Northwestern Polytechnical University, China, Xi'an 710072; E-mail: gr2003@nwpu.edu.cn
2. School of Electronics and Information, Northwestern Polytechnical University, China, Xi'an 710072
3. School of Physics and Optoelectronic Engineering, Xidian University, China, Xi'an 710071

Abstract: The solid state lidar is one of important tools for environment sensing of unmanned platform, and has been widely used in vehicle environment modeling. However, due to the low resolution, sensitive noise and complex scene, the effective segment of the whole scene is a key issue during unmanned platform data processing. In the paper, an improved 3D point clouds segmentation method is proposed for multi-line lidar in practice. After extraction building façade based on curvature segmentation, weighted Euclidean clustering is utilized to classify buildings and vegetation bodies. Then, experiments are performed on the real data acquired by the unmanned platform and the effectiveness of the proposed method is verified by comparing with the commonly used building growth segmentation algorithm.

Key Words: 3D point cloud segmentation, solid state lidar, curvature segmentation, weighted Euclidean distance

1 Introduction

With the development of lidar technology, 3D model reconstruction based on solid-state lidar has been widely used in digital city, autonomous driving, virtual reality and so on [1]. And the 3D lidar on the unmanned platform can collect the point cloud data in the 3D space, which is well-suited for urban environment sensing and 3D city model construction. However, due to the complex environment and the wide variety of targets, the quite effective segmentation of 3D point cloud data is the key issue such as building and vegetation segmentation in the urban environment.

Till now, two strategies are usually utilized for point cloud classification. The grid-based methods belong to one type [2, 3]. By projecting the 3D point cloud into the 2D plane for the following processing, this kind of methods are in fact 2.5D raster processing [2]. Hence, the grid-based methods are lost part of the 3D information. Another type of methods deal with the point cloud data in 3D space directly by using different segmentation algorithms, such as edge detection, region growth and feature clustering [4, 5]. Edge detection method is sensitive to noise, and region growth is more dependent on the initial seed points and growth criteria, so these algorithms are not suitable for complex environment with strong noise. While applying the feature clustering for lidar point clouds classification, different algorithms are considered, including K-means, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) and Euclidean distance [6, 7]. K-means algorithm has a low-speed convergence. The DBSCANS algorithm usually cause the curse of dimensionality for 3D data. The clustering algorithm based on Euclidean distance can quickly segment the point clouds of a large scene, but difficult to fix a threshold [8,9].

As mentioned above, the characteristics of the measured 3D lidar point cloud data should be considered while dealing with the 3D point cloud segmentation collected by the unmanned vehicle lidar. And in this paper we proposed an improved segmentation method for multi-line lidar 3D point clouds, which extracts the building facade based on curvature segmentation and adopt weighted Euclidean distance to classify the building and vegetation. Finally, the effectiveness of the method is validated by using the real data collected by Leishen C16 solid state lidar on the unmanned vehicle platform.

2 Methods

2.1 Data and process

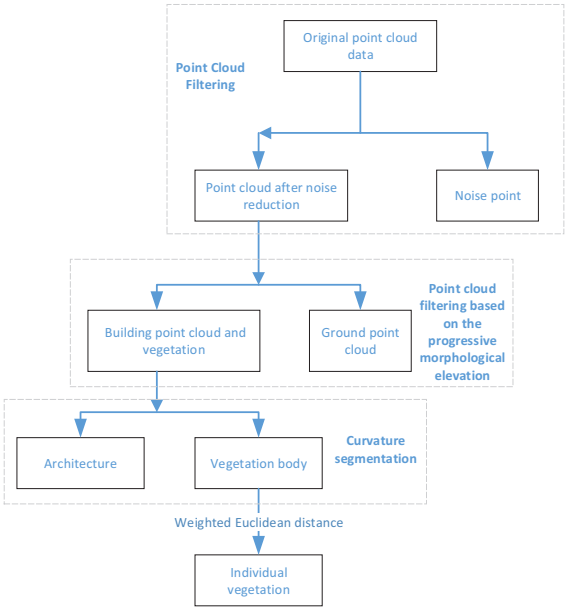


Figure 1. Flowchart of the proposed method

Generally, the 3D solid state lidar mainly adopts the multi-line scanning mode to form the laser beams in the vertical direction, while the motor rotates 360° at a fixed frequency for panoramic scanning. The collected 3D point cloud data includes the current rotation angle, the distance information, the reflection intensity, and the current frame number, etc. And finally the information of rotation angle,

* This work was funded by the National Natural Science Foundation of China (61701410, U20B2040), and by the Equipment Pre-research Foundation of China (61404130125, 61404130118).

harness angle and distance is converted to three-dimensional coordinate XYZ with other information retained. However, the increase of the measurement distance and the decrease of the resolution in the vertical direction will cause too large gaps among the points, which causes more difficulties during the point cloud segmentation while using the conventional clustering algorithms.

And here the characteristics of real 3D lidar point cloud data is considered while performing point cloud segmentation, to reduce the point interval belonging to the same target and to ensure the right segmentation of the point clouds with close spatial distance and large curvature change. The whole flowchart of the method is shown in Fig. 1, including the noise and ground point filtering, the segmentation based on the curvature method and weighted Euclidean distance.

2.2 Point Cloud Data Filtering and Segmentation

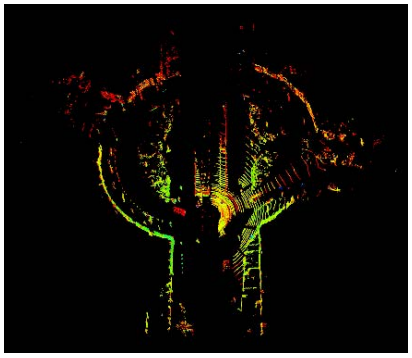
Noise and ground points filtering is a requisite preprocessing stage before point cloud segmentation, in order to better maintain the shape feature and other characteristics of the point clouds of the targets, such as buildings and vegetation. To eliminate the noise points, the detailed filtering procedure is shown as follows.

Step 1: Calculate the distances between one point and its neighborhood points in the real data.

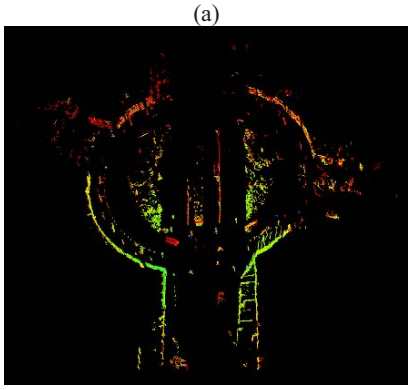
Step 2: Calculate the mean value of distances in Step 1 as the global distance of the point.

Step 3: Evaluate the statistical distribution of the global distances and then estimate the threshold.

Step 4: The neighborhood point whose distance is larger than the threshold is classified as a noise point.



(a)



(b)

Figure 2. Filtering based on (a) elevation without considering the slope (b) the progressive morphological elevation

In addition to noise points, the ground points may be mistaken for the neighbouring building or vegetation points. The common used ground points filtering based on the elevation will misjudge the building points as ground points if there was slopes in the measured environment, resulting in overfiltering. Therefore, the ground point filtering method based on progressive morphology is proposed, where the Monte Carlo method is used to set the initial elevation and slope during implement the filtering of ground points with the variable elevation threshold. Take the public KITTI point cloud data set as an example, the constant elevation and progressive elevation for ground point filtering are compared as shown in Fig.2. The KITTI data set is supplied by Karlsruhe Institute of Technology and Toyota Research Institute in America, which is the world's largest computer vision algorithm evaluation data set in the automatic driving scene. In Fig.2 (a), the point elevation value are in [-1.5m, 20m], and some ground points are still left. Setting the initial ground elevation as -2.0m and the slope value as 0.05, the ground point filtering result is illustrated in Fig.2 (b) by using the progressive morphological filter. It can be concluded that the ground points are mostly removed.

2.3 Curvature calculation

For any point in the lidar point clouds, there exists neighboring point clouds whose curved surface approaches the point. The curvature at this point can be characterized by the curvature of the local curved surface fitted by the point and its neighborhood points.

The normal vector and curvature of the point clouds are usually computed by using the differential geometry theory of discrete curved surfaces. The curvature can be estimated based on scatters or grids. The result of fitted curved surface based on grids is poor in geometric information representation, so the point cloud geometric information is estimated based on scatters, which is used to fit the curved surface by using moving least squares algorithm. After pre-filtering, the edge points and noise points has less effect on the curved surface fitting. And the detailed curvature segmentation procedure is given in Fig. 3.

Algorithm: Curvature Segmentation Algorithm

```
Input:  Point cloud samples A = (p_1, p_2, ... p_n),
        Threshold E, Neighborhood r and R
Output: Curvature division cluster D
Begin
    While 1 to n do
        For fit point p_i on the surface of the
        neighborhood r, compute the normal vector
        of r.
        For fit point p_i on the surface of the
        neighborhood R, compute the normal vector
        of R.
        Compute the curvature C_i of p_i
        if C_i<E
            Save the curvature of p_i
            Add p_i to the cluster D_1
            Update cluster D
        else
            Save the curvature of p_i
            Add p_i to the cluster D_2
            Update cluster D
        end
    end
    Return D
End
```

Figure 3. The scheme of curvature segmentation.

For any point $p_i(x_i, y_i, z_i)$ in the point clouds, select a neighborhood radius r, and perform least squares fitting on scattered points in the specified neighborhood r to obtain a fitted surface:

$$F(x,y,z) = Ax^2 + By^2 + Cz^2 + Dxy + Eyz + Gxz + Hx + Jy + Kz + m \quad (1)$$

The normal vector $\overrightarrow{n_r} = (F_{xi}, F_{yi}, F_{zi})$ of the point $p_i(x_i, y_i, z_i)$ in the neighborhood r is obtained by surface fitting, and another area of p_i with a specified radius R is selected. After surface fitting, the normal vector $\overrightarrow{n_R} = (F_{xi}, F_{yi}, F_{zi})$ of the point P_i in the area R is obtained. And by difference, the curvature tensor of p_i is obtained:

$$C = \left| \overrightarrow{n_R} - \overrightarrow{n_r} \right| \quad (2)$$

A fixed segmentation threshold E is set according to the obtained curvature tensor to segment the building point cloud cluster D_1 and the vegetation point cloud cluster D_2.

In the real experiments, the selection of the curved surface radius relates to the range between the target and lidar.

2.4 Weighted Euclidean distance calculation

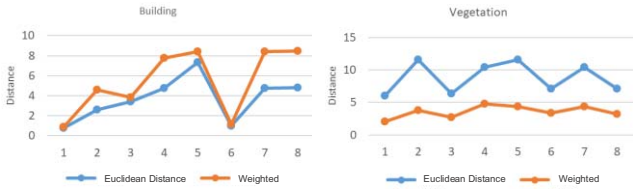
The multi-layer solid-state lidar adopts the laser pulse ranging mode, with a maximum range of 150 meters and 320,000 points per second. To improve the efficiency of such massive point cloud data processing, the octree nearest neighborhood search is employed to speed up data traversal during the traditional Euclidean clustering. However, the point clouds have the characteristics of dense near and sparse far, the constant weight cannot guarantee a complete and effective segmentation of the entire large-scale scene.

To deal with this issue, the weighted Euclidean distance is again carried on the curvature segmentation result. Considering that the scan line point cloud has the characteristics of dense near and sparse far and large vertical spacing, weighted Euclidean segmentation is proposed, and then the relative distance is applied instead of the absolute distance. The weighted Euclidean distance is implemented on all dimensions to balance the influence of each dimension on the result. It satisfies the standard normal distribution, which can be expressed as:

$$d(i,j) = \sqrt{\omega_1(x_i - x_j)^2 + \omega_2(y_i - y_j)^2 + \omega_3(z_i - z_j)^2} \quad (3)$$

where x_i and x_j are the coordinates of the point in the X dimension of the ground coordinate system, and ω_1 is the reciprocal of the variance of the X dimension. The y_i and y_j are the coordinates of the point in the Y dimension of the ground coordinate system, and ω_2 is the reciprocal of the variance of the Y dimension. The z_i and z_j are the points in the Z dimension of the ground coordinate system, and ω_3 is the reciprocal of the variance of the Z dimension.

Part of the point cloud data belonging to buildings and vegetation in the scene is selected to achieve the distance measurement between these points and the referrence vegetation point cloud. The Euclidean distance and weighted Euclidean distance are obtained as shown in Fig. 4. It can be noticed that the distance between similar point clouds become closer, and the distance between different types of point clouds increases while using weighted Euclidean distance.

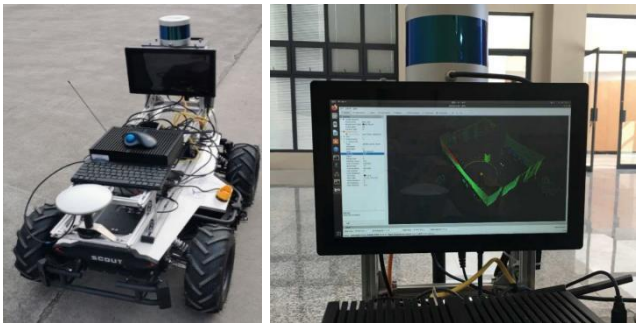


(a) (b)

Figure 4. Comparison of Euclidean distance and weighted Euclidean distance between (a) the building points (b) the vegetation points and the referred vegetation point

3 Experiments

3.1 Experiment platform and real data sets



(a) (b)

Figure 5. (a) Unmanned vehicle Lidar experimental platform (b) Lidar and data acquisition display interface

The experimental platform of the unmanned vehicle lidar is shown in Fig. 5(a), which uses the Leishen C16 3D solid-state lidar. The display interface is shown as Fig.5 (b). The Leishen C16 lidar has 16-line laser beams in the vertical direction, emitting pulses with a wavelength of 905 nm. The internal motor rotates at a frequency of 10Hz for 360° panoramic scanning. The vertical viewing angle is within ±15°. The angle between adjacent beam lines is 2°, and the maximum distance is 120m. In the collected data there are large gaps both in the horizontal and vertical directions. The measured data in the experiment has 39395 points including points of roads, buildings, vegetation and etc. The original scene is shown in Figure 6. It can be observed that as the distance from the lidar platform extends, the reflection intensity becomes lower, while the scan line interval increases.

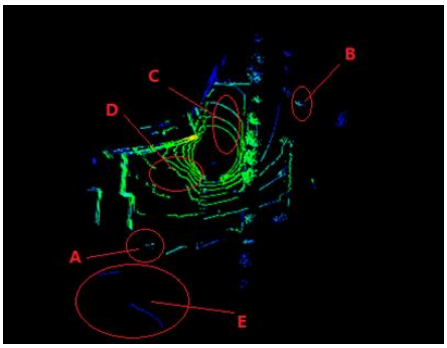
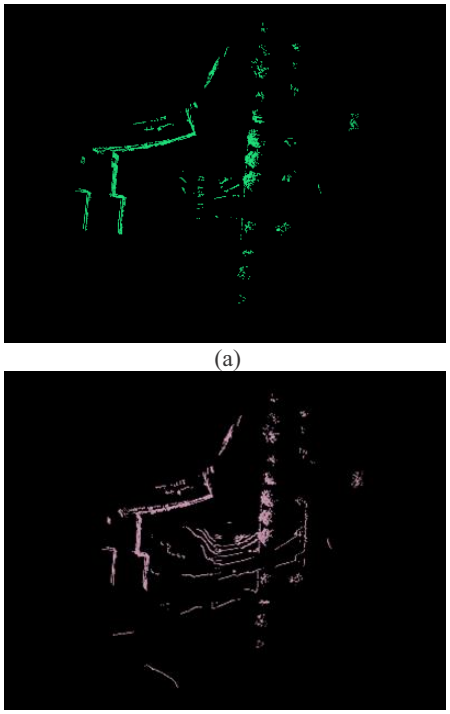


Figure 6. Original point clouds

3.2 Results of point cloud pre-filtering

The preprocessing results of noise point and ground point

filtering are illustrated in Fig. 7(a). It can be noticed that the noise points (in the marked areas A, B) and ground points (in the marked areas C, D, E) in Fig. 6 are mostly removed by using the progressive slope filter. Fig 7(b) is the result of ground points filtering directly based on elevation information without considering slopes. Setting a fixed threshold of 0.6m (take the ground as the referred horizontal plane), in Fig.7(b) the ground point clouds near the building are completely removed, however, the ground point clouds near the vegetation still exists. If filtering the ground point clouds near the vegetation, part of the building point clouds will inevitably be lost. Taking the slope into consideration in Fig 7(a), the threshold changes according to the slope. And the maximum elevation of the removed ground points is 1.64m, which verifies the necessity of using progressive morphology in the ground point clouds filtering in consideration of the slope.



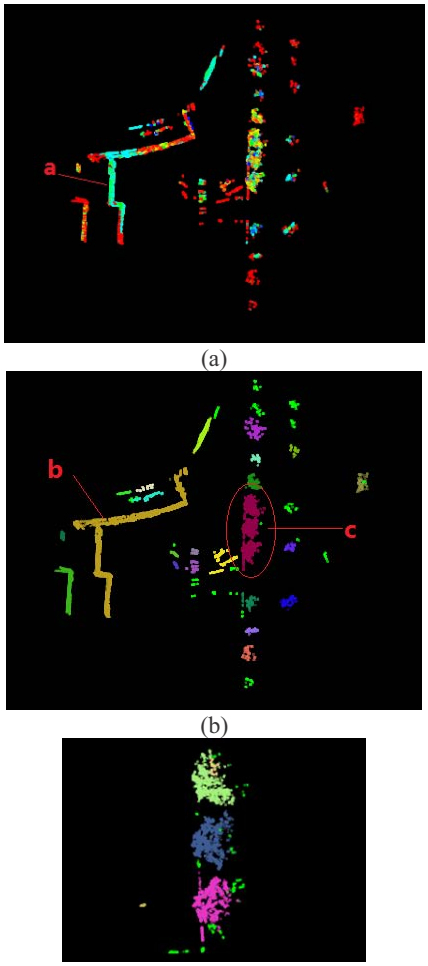
(a)

(b)

Figure 7. Noise and ground point filtering comparison (a) the proposed filtering result (b) Filtering result based on elevation without considering the slope

3.3 Building and vegetation segmentation results

The curvature segmentation result after removing noise and ground points is shown in Fig. 8(a). According to the curvature threshold, the entire scene is divided into plane and surface. The neighborhood radius is set to be 10m and 15m, respectively. In Fig. 8(a), part of the building facades is misjudged as curved cluster (marked by "a") due to the rapid change of the wall surface. Then the weighted Euclidean segmentation is implemented and achieve the result of curvature merge is revealed in Fig. 8(b), in which the building (such as marked "b") and vegetation (such as marked "c") are effectively segmented. Performing the weighted Euclidean clustering aiming at the vegetation, and the results of extracting different vegetation individuals are shown in Figure 8(c).



(a)

(b)

(c)

Figure 8. (a) Curvature segmentation result (b) segmentation result by using weighted Euclidean distance (c) individual vegetation segmentation result

The search radius of the weighted Euclidean segmentation directly determines the segmentation effectiveness. Here different neighborhood radius values are used and compared as in Fig.9. While the neighborhood radius is set to be 20m, the most accurate.segmentation result is achieved.

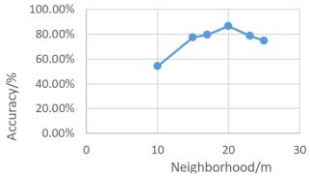


Figure 9. Accuracy of the segmentation results with different neighborhood radius values

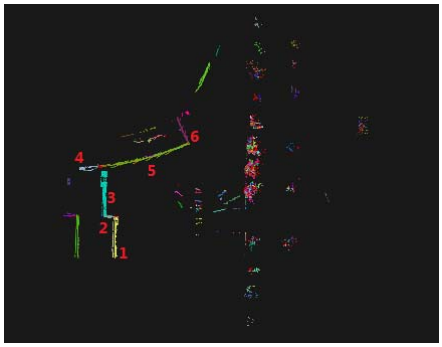


Figure 10. Results of the region growth segmentation

The region growth algorithm which based on curvature,

usually produces good results in building façade detection. In this paper the algorithm based on region growth is taken for comparison. The region growth segmentation result is displayed in Fig 10. Although the six facades of the building are completely detected as shown by marker 1-6, there are obvious over-segmentation of the vegetation.

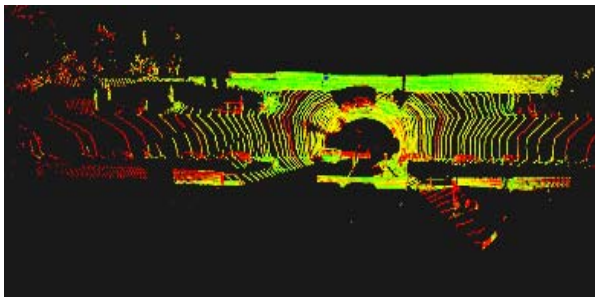
The comparison of the segmentation results are given in Table 1. Based on the real outdoor environment of the unmanned platform during data collection, the proposed method is compared with the region growth segmentation algorithm. The proposed method shows higher accuracy of building and vegetation segmentation. Moreover, the results of over-segmentation rate and under-segmentation rate are also better than the traditional region growth algorithm.

Table 1: Comparison of segmentation results

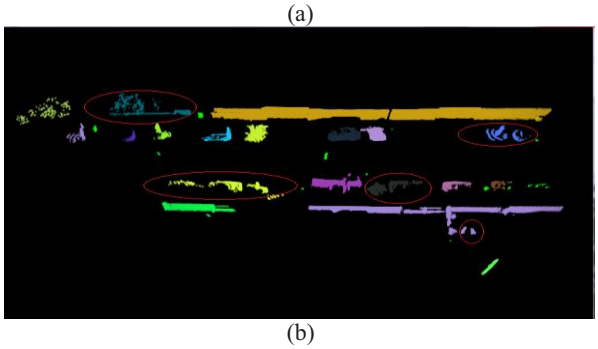
	Accuracy	Over-Segmentation	under-segmentation
Our method	87.3%	7.6%	5.1%
Region growth	72.8%	21.1%	6.1%

3.4 Experiment on the public KITTI data set

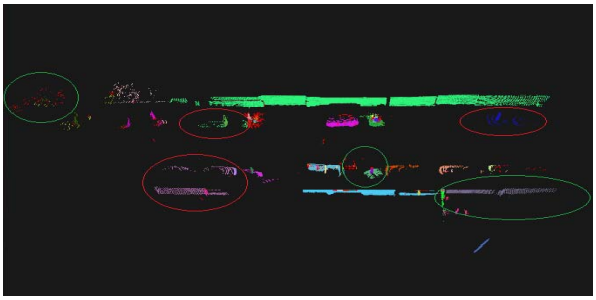
To verify the applicability of this proposed segmentation method, another data set from the public KITTI data is used to implement the experiment. Figure 11 (a) shows the original point cloud data set. Figure 11(b) and (c) are the segmentation results by using the proposed method and the region growth algorithm, respectively. The data is collected by 64 line lidar. Compared with 16 line lidar, the accuracy of the segmentation results by using the both two methods have been greatly improved to 92.1% and 88.3%, respectively. Obviously, the proposed algorithm performs better, which proves the applicability of this algorithm.



(a)



(b)



(c)

Figure 11. (a) Original point cloud data set from KITTI; (b) segmentation result by using the proposed method; (c) segmentation result by using the region growth algorithm.

4 Conclusion

Aiming at the unmanned vehicle 3D lidar point cloud segmentation, this paper start with the noise and ground point removing based on the measured data. And then considering the real data characteristics and data processing efficiency, the curvature segmentation and weighted Euclidean clustering step by step are introduced for the large-scale point clouds processing, which finally achieve the segmentation of individual building and vegetation. However, the over-segmentation of building should be noticed. On the one hand, the point clouds at the junction of the building body are relatively sparse. On the other hand, further research should be done in the future to improve the method while applying it to the real environment.

References

[1] P. Musialski, P. Wonka, D. G. Aliaga, and et al. A survey of urban reconstruction. Computer Graphics Forum, 2013, 32(6):146-177.

[2] S. Lu. Point cloud clustering algorithm for autonomous vehicle based on 2.5D Gaussian Map. Proceedings of the 2nd International Conference on Inventive Systems and Control, ICISC 2018, June 27, 2018: 754-759.

[3] A. Borcs, B. Nagy, C. Benedek. Instant Object Detection in Lidar Point Clouds. IEEE Geoscience&Remote Sensing Letters, 2017, 14(7):992-996.

[4] C. Mineo, S. G. Pierce, R. Summan. Novel algorithms for 3D surface point cloud boundary detection and edge reconstruction. Journal of Computational Design and Engineering, 2018:81-91.

[5] H. Hu, Z. Li, X. Jin, and et al. Curve skeleton extraction from 3D point clouds through hybrid feature point shifting and clustering. Computer Graphics Forum, 2020, 39(6):111-132.

[6] I. Khan, Z. Luo, J. Huang, W. Shahzad. Variable weighting in fuzzy k-means clustering to determine the number of clusters. IEEE Transactions on Knowledge and Data Engineering, 2020, 32(9):1-1.

[7] D. Deng. Application of DBSCAN Algorithm in Data Sampling. Journal of Physics: Conference Series, 2020, 1617(1):012088.

[8] X. Ruan, B. Liu. Review of 3D Point Cloud Data Segmentation Methods. International Journal of Advanced Network, Monitoring and Controls, 2020, 5(1):66-71.

[9] Z. Cheng, G. Ren, Y. Zhang. Ground Segmentation Algorithm Based on 3D Lidar Point Cloud. Proceedings of the 2018 International Conference on Mechanical, Electrical, Electronic Engineering & Science (MEEES), 2018:27-32.