IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 1 June 2022, accepted 27 July 2022, date of publication 4 August 2022, date of current version 10 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3196388

RESEARCH ARTICLE

Noise Point Detection From Airborne LiDAR Point Cloud Based on Spatial Hierarchical Directional Relationship

YUNHAO LIN, YANJUN WANG, SHUHAN WANG, SHAOCHUN LI, MENGJIE WANG, HENGFAN CAI, AND FEI TENG

Hunan Provincial Key Laboratory of Geo-Information Engineering in Surveying, Mapping and Remote Sensing, Hunan University of Science and Technology, Xiangtan 411201, China
National-Local Joint Engineering Laboratory of Geo-Spatial Information Technology, Hunan University of Science and Technology, Xiangtan 411201, China
School of Resource Environment and Safety Engineering, Hunan University of Science and Technology, Xiangtan 411201, China

Corresponding author: Yanjun Wang (wangyanjun@hnust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 41971423 and Grant 31972951, in part by the Natural Science Foundation of Hunan Province under Grant 2020JJ3020, in part by the Science and Technology Program of Hunan Province under Grant 2019RS2043 and Grant 2019GK2132, and in part by the Postgraduate Scientific Research Innovation Project of Hunan Province under Grant CX20210991.

ABSTRACT In three-dimensional (3D) airborne light detection and ranging (LiDAR) point-cloud data acquisition, noise point clusters (such as cloud, birds and incomplete scanning ground points) and isolated points are usually generated in the scanning process. Detection and elimination of these noise points directly affect the subsequent processing efficiency of the LiDAR point clouds. In this paper, a noise detection method from airborne LiDAR data based on spatial hierarchical directional relationship and region growing algorithm is proposed. First, the original airborne LiDAR points are divided into regular 3D grids, and the maximum point density unit is searched adaptively to select the initial surface seed points for region growing algorithm. Then, the spherical neighborhood is constructed with the initial seed point as the center, and fourteen main growth directions are generated based on the 3D space topology. Second, candidate seed points in each main direction are determined by the distance threshold. Finally, all LiDAR points are iteratively executed using candidate seed points as new region growing seed points. This paper selects two mountain terrain scenes with different cloud contents as the study area, and the precision, recall rates and F1-score of the proposed method reach 99.8%, 100% and 99.3%, respectively. This method can detect point-cloud clusters and isolated points, thus simplifying the LiDAR point clouds, providing basic support for the subsequent accurate data processing and analysis.

INDEX TERMS Airborne LiDAR, aerial laser scanning, point cloud noise, noise detection, hierarchical direction, region growing, spherical neighborhood, outlier point removal.

I. INTRODUCTION

Three-dimensional (3D) light detection and ranging (LiDAR) technology has gradually become one of the important data sources for earth observation and target classification and recognition because of its rapid, real-time access to environmental information. LiDAR is a high-precision remote sensing method used for measuring the position and shape of objects and forming high-quality 3D point cloud images [1]. In recent years, airborne LiDAR has been widely used in aerospace [2], marine exploration [3], 3D modeling [4], [5], tree biomass estimation [6], [7] and other fields. To enable good application prospects in various fields, researchers have proposed a series of geographical object recognition (including ground/terrain, roads, power lines [8], [9], trees [10], [11], water bodies, buildings and so on),

The associate editor coordinating the review of this manuscript and approving it for publication was Krishna Kant Singh.

classification and extraction algorithms based on LiDAR point clouds.

However, acquisition of LiDAR point-cloud data can be affected by external environmental factors [12], such as atmospheric particles, rain, snow and other adverse weather conditions, and multipath echo caused by laser diffuse reflection during instrument scanning [13], [14], resulting in noise point-cloud noise. Therefore, data preprocessing, such as noise detection and filtering, are critical for ensuring accurate classification and information extraction from point cloud data.

At present, the noise points in LiDAR data mainly include isolated points and noise clusters. Isolated points include isolated outliers, noise points near the signal [14], and isolated points in different directions away from the ground surface, which can be divided into high anomalies and low anomalies [15]. Noise clusters are aggregated noise points. These noise points will affect the 3D solid model reconstruction and target feature classification. It is therefore necessary to clean the point-cloud data by removing noise. However, quickly and effectively detecting point-cloud noise under complex scene conditions remains challenging.

The remainder of this paper is organized as follows: Section II take an overview of LiDAR point cloud method for removing noise points, Section III illustrates the methodology, Sections IV reports the experimental area and results, and Section V reports the discussion. Finally, Section VI draws the conclusions of the work.

II. RELATED WORK

The current point cloud denoising methods can be roughly divided into three categories: overall environment denoising, eliminating special noise points and ground fil-tering. The detailed explanation of these three denoising types is as follows:

A. OVERALL ENVIRONMENT DENOISING METHOD

For the point cloud data processing containing multiple types of surface objects, the noise point is defined as the discrete point or noise cluster generated by floating objects in air and diffuse reflections from instruments during data acquisition. Since a complex scene of point cloud data could be composed of diverse surface objects, and there is no obvious correlation or similarity between different objects, research methods have mainly been focused on the relation-ship between neighboring points or sets of points belonging to two different kinds of objects.

The commonly used methods for detecting noise points are: traditional filtering methods, such as median filtering [16] and mean filtering [17]. There are problems in the universality of these methods for complex point cloud data. The research idea based on point neighborhoods is currently the main means of denoising. Rusu [18] proposed the statistical outlier removal (SOR) filter, which distinguishes noise from signals by the average distance statistics of point neighborhoods. Jia-Jia *et al.* [19] proposed the spatial frequency outlier filter based on a spherical neighborhood search. These two methods have problems detecting noise point clusters. Another commonly used statistical filtering of radius neighborhood points [12] could classify the points far from LiDAR as noise point elimination, resulting in the destruction of scene features. To improve the accuracy of the results, researchers proposed improvements on the basis of the original filter, such as fast clustering SOR filtering [20] and dynamic radius neighborhood point statistical filtering [12]. However, these methods are limited to detecting isolated points and do not confirm the detection effect of noise clusters.

Feature clustering is often used to determine the relationship between sets of points. Ester *et al.* realized the wide application of density-based spatial clustering application with noise method (DBSCAN) [21], and then a method dominated by density features appeared. The DBSCAN method has been improved by using an ellipsoid [22] unit to divide the point cloud space and by using the accessibility distance [23] as a threshold, but the improved method is complex and high computational demand. The first two density clustering methods have problems in terms of the universality of steep terrain areas and are sensitive to the input parameters of the method. Before the application of the clustering method, principal component analysis (PCA) is usually used to represent the point set on the feature plane, and some noise points are removed by using the discrete nature of noise points [14], [24]–[26]. However, how to apply PCA in complex scenes is a problem.

B. SPECIAL NOISE POINTS ELIMINATION METHOD

Point cloud data obtained by LiDAR are greatly affected by adverse weather conditions and the geographical environment. The noise types are mainly meteorological feature elements (including clouds, snow and other solid liquid particles condensed in air). At present, there are some methods to detect specific snow noise points by combining the intensity information of snow in the point cloud with the application of deep learning [27] or improving the existing filter [12], [28].

C. GROUND POINT FILTERING METHOD

The denoising method of terrain data is equivalent to ground point filtering, and the denoised point set can be used to construct a digital elevation model (DEM) or even digital surface model (DSM).

Traditional ground point filtering methods can be divided into three types: slope-based, surface-based and segmented methods [29]. Slope-based methods assume that if the height difference between two points is greater than the threshold, then the points should be divided into different classes. For example, gradient-based or slope-based filtering methods can quickly and efficiently separate flat terrain ground points and non-ground points, but this method is not robust in areas with large terrain fluctuations [30]. The method based on surface and segmentation for surface fitting can be further divided into two subcategories: the method based

on morphology [31], [32] and the method based on interpolation [29], [33]. Depending on the performance of the two methods in different environments, filters that work well in forested environments may not work well in urban environments [34].

The noise detection method combining histograms with common filters [13], [35], [36] or using adaptive TIN (Triangulated Irregular Network) denoising [37] after specifying the elevation range is not highly automated and some steps need visual rules. Zhang *et al.* proposed a window adaptive ID threshold denoising method based on a quadtree [38] for the discrete point denoising of satellite LiDAR data, but the detection effect of noise clusters was not proved. Zhang *et al.* proposed a cloth simulation filter to obtain ground points [39], but it is not suitable for point cloud data with negative outliers. Therefore, Yang *et al.* proposed establishing the transfer iterative trend surface to remove negative anomalies and then applied the bidirectional cloth simulation filtering correction model to extract the convex and concave seabed terrain [40]. However, seabed terrain is a surface. For other point cloud data with large terrain fluctuations, the applicability of bidirectional cloth simulation filtering has issues.

Simulation method is suitable for noise point detection of surfaces or plane point clouds. The method based on region growing is a popular choice for roof surface noise detection [41]. Besl and Jain [42] proposed two stages of the method, including rough segmentation based on the average and Gaussian curvature of each point and its symbol, and refinement of iterative region growing based on variable-order binary surface fitting. This method was later used for 3D point cloud segmentation by others. The traditional regional growing algorithm takes the normal vector angle and curvature as the growth criteria, and is commonly used in the extraction of roof surfaces in the field of LiDAR [43]. Gorte *et al.* [44] selected a triangle from the input irregular triangulated network as the seed and applied region growing to add adjacent triangle iterations to the current line segment. Cao *et al.* [45] first selected seed points in the dimension-reduced parameter space, and then segmented planar patches in the space using region growth. Gilani *et al.* [46] proposed an improved PCA method to generate a consistent point normal. When the number of points is large, the method based on region growth is easy to implement and faster than the method based on model fitting [47], but the curvature and normal vector angle are highly sensitive to noise points, and it is difficult to detect an accurate boundary between smooth regions.

For roof surfaces, especially in the roof border cross area, the method of region growing will lead to poor segmentation of complex structure buildings, so the detection of noise points of other types of objects will judge the signal as noise point removal, resulting in Type II error. The current region growing algorithm considers the planarity and is based on the point-to-point or point-to-fit plane angle. The threshold setting standard is not uniform when the method is applied to detect noise points with spatiality. When the threshold setting is too large, all noise points cannot be detected, and if the threshold is too small, then leakage points will appear.

In general, the above methods for removing noise points from airborne LiDAR point clouds have the following shortcomings:

- Traditional denoising methods based on geometric features and statistical features are not suitable for the simultaneous detection of noise clusters and isolated points;
- The method of ground point filtering to detect noise points is to detect ground points with curved surface characteristics, but for complex terrain scenes, a large number of environmental features will be lost;
- At present, the noise point detection method affected by environmental factors is limited to the application of specified noise points such as intensity information, while other types of noise points do not have the same characteristics. Therefore, the universality of the ROR (Radius Outlier Removal) method dependent on the LiDAR point cloud density has problems;
- At present, the region growing algorithm is less applied to overall denoising, and similar to other ground point filtering methods, the processing scene is mostly for surfaces or planes. The appropriate parameter threshold cannot be determined when the parameter settings in the surface method are used for global denoising.

Aiming at the problem that the existing denoising methods cannot detect all noise-like points and are affected by the scanning environment, based on airborne Li-DAR point cloud data, this paper proposes a regional growth algorithm based on spatial hierarchical directional relationships to detect noise points. The directional connectivity of 3D points is defined by the distance between points, and the distance is used as the threshold of region growing to detect noise points in point clouds in complex environments.

III. MATERIALS AND METHODS

A. TECHNICAL PROCESS

Based on airborne LiDAR point cloud data, this paper designs a region growing algorithm based on spatial hierarchical direction to detect cloud noise clusters and isolated points in mountain range scenes. First, the original airborne LiDAR point cloud data is applied to adaptively select the initial seed points of regional growth by certain rules. Then, the spatial connectivity of the 3D point cloud is determined, the spherical neighborhood is constructed with the initial seed point as the center, and 14 main growth directions are generated. Second, the farthest interior point in each major direction is searched in the spherical neighborhood point set of seed points. Finally, all LiDAR point sets are iteratively executed using candidate seed points as new region growing seed points. The detailed technical process of this method is shown in FIGURE 1.

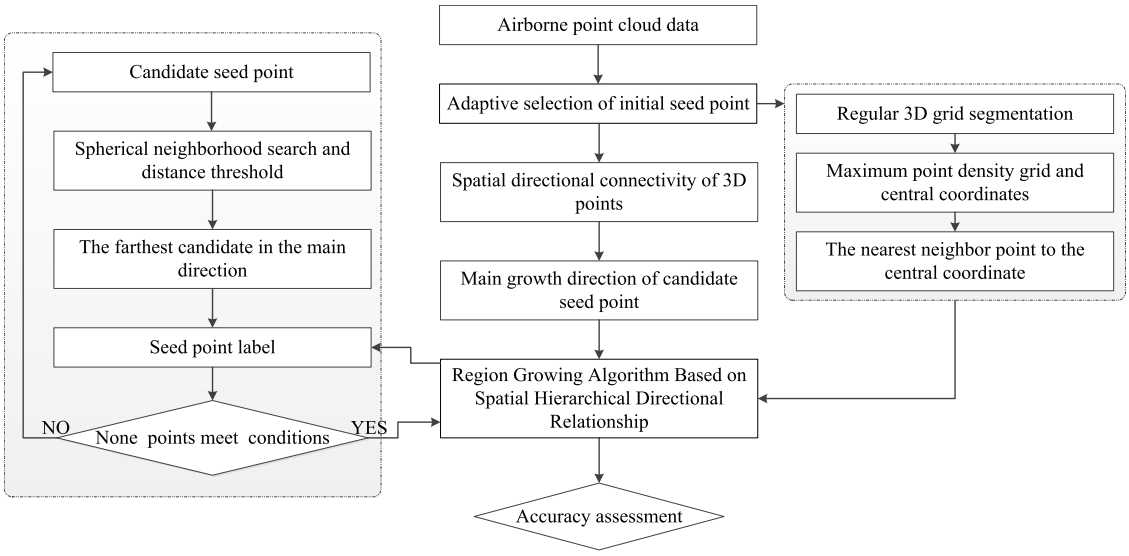


FIGURE 1. Flow chart of noise point detection by the region growing algorithm based on the spatial hierarchical directional relationship.

B. ADAPTIVE SELECTION OF THE INITIAL SEED POINT

The selection of initial seed points is a key step of the region growing algorithm, which will guide the overall direction of growth and even determine the classification of noise points. In existing region growing algorithms, the initial seed points are obtained manually, but this method is not adaptive. To improve the automation performance of the whole method, researchers have proposed algorithms to search for the initial point with certain rules. The commonly used rule is to calculate the average curvatures of all points in the LiDAR point cloud data and take the minimum curvature point as the initial seed. The curvature represents the bending degree of the curve and is solved by the normal vector. The traditional regional growing algorithm takes the normal vector angle calculated by PCA as the threshold condition, so it is relatively easy to obtain the minimum curvature. The minimum curvature is represented as the point with the best planarity in the point cloud data, so the initial seed point will be selected in the point set with a strong plane and large point density such as a road or roof surface, thereby reducing the probability of the point in the noise point cluster as the initial seed point. However, when this method is used for point cloud data with large point density in the noise cluster, the robustness of the initial point selection on the signal point is problematic.

In the airborne LiDAR data acquisition process, a large number of cloud noise clusters may be scanned due to the influence of environmental conditions. These cloud noise point sets are independent of the field scene point sets with mountains as the main body and might have a higher point density. The curvature of cloud noise points obtained by calculating the rules through neighborhood points is also small, and the position of the current minimum curvature point cannot be determined easily. If the minimum curvature point is used as the initial seed point, then it is possible that the minimum curvature point is in the cloud cluster and will result in incorrect noise point detection. Therefore, the method of selecting initial points based on strong planarity is not suitable for the above types of data. In this study, we propose a method to search for the center point of the maximum point density cell from probable candidate space regions as the initial seed point, which is applied to improve the robustness of seed point selection.

The whole point cloud space is divided into a regular 3D cube grid. The cube length, width and height are set to 20 meters and the 3D unit grid is calculated according to the minimum coordinate values of the X, Y, and Z axis in the raw LiDAR point cloud data. Suppose the minimum coordinates on the three axes are X_{min}, Y_{min}, and Z_{min}, the length, width and height of the cube grid are l, and the position of each point in the cube grids may be determined by Equation (1).

$$\begin{cases} L_i = \left\lfloor \frac{X_i - X_{min}}{l} \right\rfloor + 1 \\ W_i = \left\lfloor \frac{Y_i - Y_{min}}{l} \right\rfloor + 1 \\ H_i = \left\lfloor \frac{Z_i - Z_{min}}{l} \right\rfloor + 1 \end{cases} \tag{1}$$

where L_i, W_i and H_i represent the levels of the current point position on three coordinate axes of X, Y and Z, respectively; $\lfloor\ \rfloor$ is floor function; X_i, Y_i and Z_i represent the X, Y and Z coordinate values of the current point, respectively; and l is the edge length of a square.

Each point is marked in different grid cells after the calculation. These cube grids, which contain different numbers of points can be hierarchically divided along the Z axis.

1) The probable candidate space regions can be determined by histogram analysis of the height distribution. Specifically, the signal points are mostly distributed near the middle-level region, and the noise points are mostly distributed in the upper- or lower- level region. Therefore, the probable

candidate space regions consist of cube grids near the middle-level region along the Z axis.

2) The 3D unit with the largest point density in probable candidate space regions can be selected. The units are sorted by the numbers of points in them. The unit with the largest number of contained points, named D, is the grid cell with the largest point density.

3) The center coordinate C_D of unit D is calculated as the initial seed point. However, the center point of the point set does not necessarily correspond to the center coordinates of the unit, so the k-nearest neighbor (KNN) algorithm is used to search the point $p0$ closest to the center coordinate as the center point, and the point is used as the initial seed point of the region growing algorithm.

Point density is an important feature reflecting the difference between noise clusters and signal clusters. From the scanning trajectory of the laser range finder, whether it is vehicle or airborne point cloud data, the dense point cloud is concentrated in the elevation range of the terrain. Even if the point density of the cloud noise cluster of the above type of data is less than that of the ground point, the initial seed point is the ground point. Since the final result of point cloud denoising is to detect noise points unrelated to the overall scene and the signal point is the aggregation point set with the ground point as the connection medium, the ground point as the initial seed point is more conducive to growth.

C. DEFINITION OF SPATIAL DIRECTIONAL COONECTIVITY

Point cloud data are composed of a series of 3D point sets, and the topological relationship between points in 3D space is categorized as adjacent and overlapping. Adjacent and overlapping relations can be expressed by the definition of a neighborhood, but there are no high-level topological objects such as lines, surfaces and volumes in point cloud data, so it is impossible to construct high-level topological structure models. Moreover, the calculation process for further constructing other objects by connecting points constituting lines or rings on point cloud data with a large amount of data is very complicated, which easily causes data redundancy in most topological models and incomplete integration of topological and geometric information.

Based on the idea of neighborhood and region growing, this paper proposes a hypothesis that connectivity can be judged by computing Euclidean distance between points in space. The object is composed of point sets with a certain distance in the space. LiDAR scans the point set data representing different objects, and the point distance at the junction of connected objects is smaller than that between non-connected objects. The distance threshold between points is determined by the distance between points in the dense point area of the point cloud data. If the distance between points fluctuates in a small range of the distance between points, then the direct connectivity between these points is considered.

There is a direct or indirect connection between 3D points. The signal point is the aggregation point set with the ground point as the connection medium, and most of the noise points are isolated point sets floating in space without connection with the signal point; that is, the distance between the noise point and the signal point is greater than the defined distance threshold. However, the efficiency of searching the connected boundary of the whole point set by the distance threshold of the direct connection relationship is too low or even falls into a dead cycle. Therefore, this paper uses indirect connectivity to find the boundary, sets a distance threshold greater than a certain multiple of the sampling spacing and stipulates the main direction of boundary search to simplify the search process.

D. MAIN GROWTH DIRECTION OF THE CANDIDATE SEED POINT

After the region growing algorithm is applied to search the inner point of the cur-rent seed point through a spherical neighborhood, it takes the seed point as the center and the sphere as the range boundary to find the inner point furthest in this direction along the specified direction as the candidate seed point. After the candidate points in each direction are determined, all the candidate points are marked as seed points, and the spherical neighborhood search is performed at the same time to determine the next batch of seed points. The growth trend takes place in several specified main directions and takes the farthest point as the candidate seed point to reduce the point-by-point iterative growth process, which is equivalent to the initial seed point-centered diffusion.

The main growth direction is determined by the 3D anisotropy of all the ungrown points in the current spherical neighborhood, which represents the main representative direction of the free growth of seed points in space and reduces the redundancy calculation and judgment before the next growth. The direction is set, as shown in Figure 2. The spatial coordinate system is established with the point $p0$ as the coordinate origin of the current seed point, and the spherical neighborhood search range of a sphere representing the current seed point is drawn with the point as the center d of the sphere as the radius. If the coordinate axis is the main growth direction, then the six coordinate directions can be expressed in the form of vectors: $(\mathbf{0}, \mathbf{0}, \boldsymbol{d})$, $(\mathbf{0}, \mathbf{0}, -\boldsymbol{d})$, $(\mathbf{0}, \boldsymbol{d}, \mathbf{0})$, $(\mathbf{0}, -\boldsymbol{d}, \mathbf{0})$, $(\boldsymbol{d}, \mathbf{0}, \mathbf{0})$ and $(-\boldsymbol{d}, \mathbf{0}, \mathbf{0})$ (represented by six red vector lines in FIGURE 2), the farthest inner point in the main direction is the candidate seed point. However, the six spheres formed by the seed points in the six main directions cannot cover all the point sets that can be grown and cannot reflect the 3D heterogeneity of spatial directional relationship.

Based on the original spatial coordinate axis, an improvement was made. Taking $p0$ as the eight intervals of the spatial coordinate system of the coordinate origin as the object, a vector was added to each interval, and the angle between the vector and the coordinate axis was set to 45° as the auxiliary coordinate axis. The original coordinate axis and the auxiliary coordinate axis constitute the main direction of the growth of 14 seed points $p0$. In addition to the previous six direction vectors, eight orientation vectors (represented by eight green vector lines in FIGURE 2) were added, as shown in

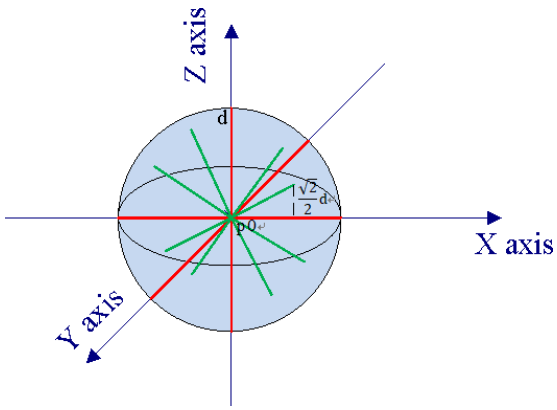


FIGURE 2. Determination of the main direction of growth.

Equation (2). Similarly, the sphere with the sphere range as the boundary and the innermost point on the vector as the central point is drawn with radius as the sphere, which can cover all points.

$$\mathrm{main_i} = \begin{bmatrix} \frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d \\ -\frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d & -\frac{\sqrt{2}}{2}d \\ -\frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d \\ \frac{\sqrt{2}}{2}d & -\frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d \\ \frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d & -\frac{\sqrt{2}}{2}d \\ -\frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d & -\frac{\sqrt{2}}{2}d \\ -\frac{\sqrt{2}}{2}d & -\frac{\sqrt{2}}{2}d & \frac{\sqrt{2}}{2}d \\ \frac{\sqrt{2}}{2}d & -\frac{\sqrt{2}}{2}d & -\frac{\sqrt{2}}{2}d \end{bmatrix} \tag{2}$$

where $\mathrm{main_i}$ represents the main growth in i direction, d represents the distance threshold of the spherical neighborhood, and each row of the matrix represents the direction vector of an auxiliary coordinate axis. There are eight direction vectors in total.

In this paper, the distance from point to point represents the indirect connectivity of points in the space, and the spherical neighborhood centered on seed points is formed to obtain the next batch of candidate seed points. Each candidate seed point forms its own connected domain (spherical neighborhood), and the direct or indirect connectivity of each connected domain reflects the characteristics of the hierarchicaldirection.

E. REGION GROWING ALGORITHM BASED ON THE SPATIAL HIERARCHICAL DIRECTIONAL RELATIONSHIP

The region growing algorithm first selects a point or region as a seed and then ex-tends it iteratively to adjacent points using appropriate rules. The growth criterion of the distance from point to plane and the angle difference between normal vectors are two widely used similarity measures [41]. This method judges whether the interior point conforms to the growth condition by calculating the similarity between the point and the neighborhood interior point, which is equivalent to each point having a judgment whether it grows and finally retains each point as a seed point. However, the efficiency of this method is too low.

Based on the assumption of spatial hierarchical directional relations between the 3D points, this paper takes the distance from point to point as the similarity measure of growth clustering and determines and marks interior points by spherical neighborhood search, as shown in Equation (3). Several points farthest from seed points (center points) in different principal directions in spherical neighborhood units are selected as candidate seed points, as shown in Equation (4).

$$\mathrm{P_i} = \begin{cases} 1, & \mathrm{dis}_{(\mathrm{P_i,p0})} \le d \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $\mathrm{P_i}$ represents the 3D point of the original point cloud data ($i = 1, 2, 3\ldots$), $\mathrm{dis}_{(\mathrm{P_i,p0})}$ represents the Euclidean distance from any original point to the seed point $p0$, d represents the distance threshold of the spherical neighborhood, 1 represents the marked inner point, and 0 represents that the point is not the inner point of the current seed point. When $\mathrm{dis}_{(\mathrm{P_i,p0})}$ is less than the distance threshold, the point is marked as 1; otherwise, it is not marked.

$$\mathrm{p_i} = \mathrm{main_i}\,[\mathrm{nearest}\,(S, \mathrm{P_i}\,(1))] \quad (0 \le j \le 14) \tag{4}$$

where $\mathrm{p_i}$ represents the candidate seed points of seed point $p0$ in the i derection, $main_i$ represents the i growth direction, nearest(S, $\mathrm{P_i}(1)$) is used to calculate the nearest neighbor interior point of the sphere, S represents the surface of the current spherical neighborhood, and $\mathrm{P_i}(1)$ represents the original point marked as the interior point.

This method selects several points that meet the conditions as candidate seed points and classifies different point sets to avoid each point cycle judgment, which reduces the time complexity of the algorithm. The purpose of the distance radius setting is to find the segmentation area between the signal point and the noise point. The scanning area of the point cloud data in the minimum segmentation area is determined, and there is no clear range. Considering that there is a large distance area between the signal point sets, the optimal radius threshold is obtained by taking different distance thresholds into the experimental test of the algorithm.

The method in this paper is shown in Figure 3. The black point is the original point cloud of the current region. $p0$ is the initial seed point obtained by the nearest neighbor search of the center point of the current grid unit ($p0$ can also represent the seed point in the growth process). A spherical neighborhood is created with $p0$ as the center and d as the radius (such as the neighborhood with a pale purple sphere as the boundary in FIGURE 3), and the points in the sphere are marked as the

TABLE 1. Statistical results of point cloud data information.

Study Area	Area Coverage (m×m)	Average Point Density (point/m^2)	Scanning Points (point)	Signal Points (point)	Discrete Noise Points (point)	Noise Cluster Points (point)
S_1	1,500×1,500	~2.7	5,979,463	5,700,746	114	278,603
S_2	1,520×1,450	~3.6	7,986,516	3,289,727	99	4,696,690

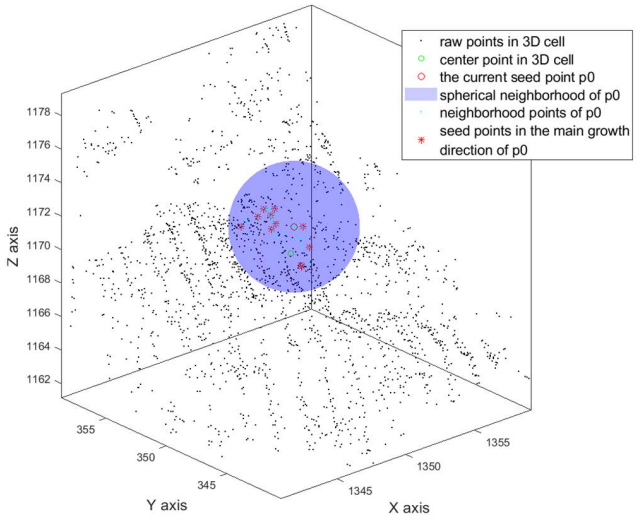


FIGURE 3. Region growth algorithm based on spatial hierarchical direction.

inner points (pale blue points). In the inner points, the points closest to the surface of the sphere in the main direction are marked as candidate seed points (such as 10 points with red '$*$' symbols in FIGURE 3). Subsequently, these 14 candidate seed points are the center points of the next batch of spherical neighborhoods to continue to grow.

When the point set between the seed point and the candidate seed point is judged whether it is an inner point, the next growth is carried out. During the growth process, all LiDAR point sets are iteratively executed, and finally, the growth ends when all points in all directions are marked.

IV. RESULTS

A. DATA SOURSES

The study site in this paper is a mountain area, located in Honolulu, Hawaii, with a small number of trees and roof surfaces of buildings. The floating clouds and some isolated points in the air are the main components of the noise points. The airborne LiDAR data were acquired in summer 2013 using an Optech ALTM GEMINI laser system (scan rate: 37 Hz; laser pulse rate: 70,000 Hz; multipulse in air mode enabled with up to five echoes) mounted on a twin-engine Piper PA-31 Navajo airplane (aboveground flight height: ~800-1400 m). To verify the denoising effect of our proposed method in different terrain scenarios, we select two types of terrain scenarios as experimental data: S_1 and S_2, one with fewer holes in the signal area and more floating clouds or mist (FIGURE 4(a)) and another with a large number of holes in the terrain due to nearer cloud occlusion (FIGURE 4(b)). The missing degree of signal areas and types of noise points in these two study areas are comparatively different.

Since the airborne LiDAR system obtains point cloud data from the top of the terrain, the rangefinder can only obtain surface information when the laser is emitted from the top to the bottom on the solid surface, while other structural lasers blocked by the uppermost solid surface cannot penetrate, resulting in holes. As shown in FIGURE 5(a), there are a large number of trees and buildings in the study area S_1, but only the crowns and roofs of buildings are scanned (red ellipse areas), so that a small number of voids can be seen in those areas. As shown in FIGURE 5(b), the red ellipse areas exhibit discrete points and the scanned cloud noise cluster is far away from the mountain. The cavities caused by buildings and trees are small, and the continuity of the terrain is not destroyed. The cloud in S_2 is close to the mountain, and the airborne LiDAR system flight path is above the cloud. If the clouds are scanned first, causing the point clouds of other objects under the cloud to be occluded (FIGURE 5(c)), then a wide range of voids are present within the signal points (red ellipse areas). The statistical information of the experimental area is shown in TABLE 1.

B. EXPERIMENTAL RESULTS

Combined with the study areas $\boldsymbol{S_1}$ and $\boldsymbol{S_2}$, the method proposed in this paper is used for experiments. Based on the original airborne LiDAR data, the noise points separated by artificial eyes are used as the real reference data. At the same time, the accuracies **Pre**, **Rec** and **F1 score** are selected to quantitatively evaluate the noise point detection results. The calculation method is shown in Equations (5-7).

$$\mathrm{Pre} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{5}$$

$$\mathrm{Rec} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{6}$$

$$\mathrm{F1} = \frac{2 * \mathrm{Pre} * \mathrm{Rec}}{\mathrm{Pre} + \mathrm{Rec}} \tag{7}$$

where **TP** represents the number of points correctly detected as noise points, **FP** represents the number of points incorrectly detected as noise points, and **FN** represents the number of points missing as noise points.

To further verify the performance of the proposed method for detecting noise points from the study area, we compare the proposed method with the following: the commonly used SOR filtering, filter based on the point-to-fit plane distance, DBSCAN, Euclidean cluster extraction and region growing algorithm based on normal vector. The quantitative evaluation results are shown in TABLE 2, all the parameters of

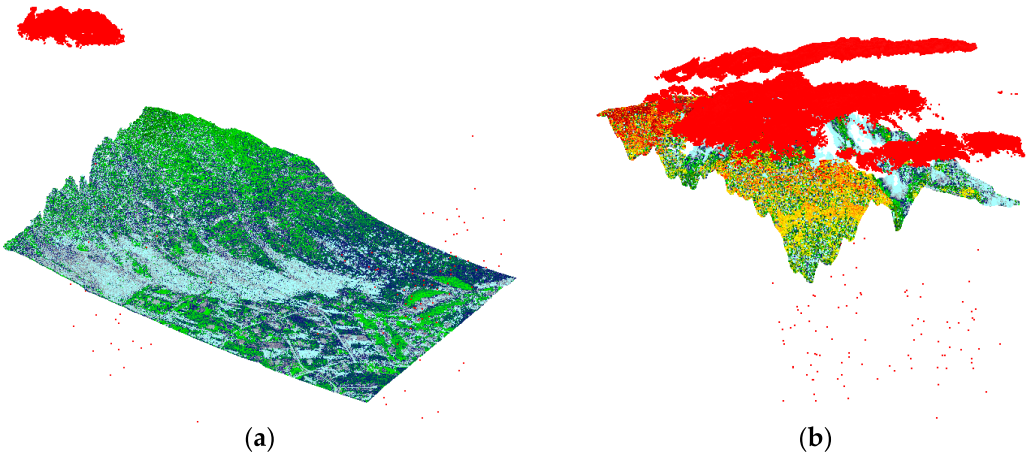


FIGURE 4. Experimental data (red points represent noise points and the colors of signal points are displayed with high contrast intensity values): (a) study area of S_1 and (b) study area of S_2.

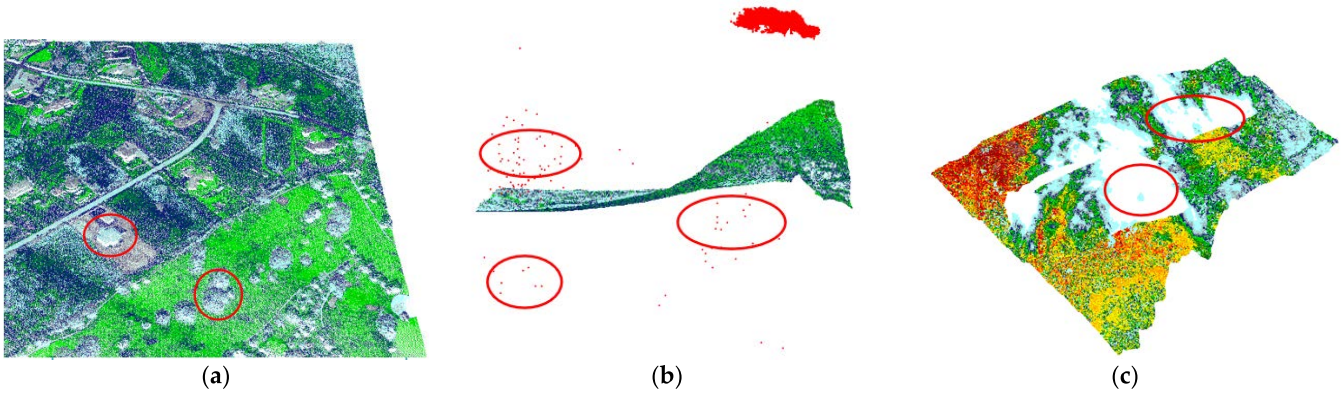


FIGURE 5. Characteristics of the study area: (a) The hollowness of trees and buildings in S_1; (b) isolated points in S_1; and (c) large voids caused by cloud occlusion in S_2.

TABLE 2. Accuracy evaluation of the experimental results.

Study area	Noise point detection method	Pre (%)	Rec (%)	$F1$ (%)
S_1	Radius outlier removal (SOR) filtering	73.7	0.3	0.6
	Filter based on point-to-fitting plane distance	18.3	6.5	9.6
	Density-based spatial clustering application with noise (DBSCAN)	100.0	47.9	64.8
	Euclidean cluster extraction [1]	91.3	100.0	95.4
	Region growing algorithm based on normal vector	68.6	100.0	81.4
	Our method	**92.4**	**100.0**	**96.0**
S_2	Radius outlier removal (SOR) filtering	96.9	0.8	1.6
	Filter based on point-to-fitting plane distance	77.5	3.9	7.4
	Density-based spatial clustering application with noise (DBSCAN)	99.3	97.7	99.2
	Euclidean cluster extraction [1]	98.8	98.8	98.8
	Region growing algorithm based on normal vector	98.8	94.1	96.4
	Our method	**99.8**	**98.8**	**99.3**

[1] The direct results of the Euclidean cluster extraction method for S_1 and S_2 are many 3D clusters. The Pre, Rec and $F1-score$ are computed after simple manual post-processing, such as grouping some clusters together.

comparison methods are the optimal setting results obtained after repeated debugging. The results of noise points detected by the proposed method are shown in FIGURE 6. The results of noise points detected by the comparison methods in the study area are shown in FIGURE 7.

From TABLE 2, the results indicate that the recall rates of the SOR filtering and filter based on point-to-fitting plane distance are very low, and the precision rates are clearly high. Combined with the analysis of the denoising results, it can be seen that the two methods remove isolated points and a small number of cloud noise clusters, but the proportion of scattered points in the noise points in the experimental area is very small, resulting in a very low recall rate. Because the denoising algorithm removes some signal points, very few noise points are removed, resulting in a decrease in precision. The method, filter based on point-to-fitting plane distance,

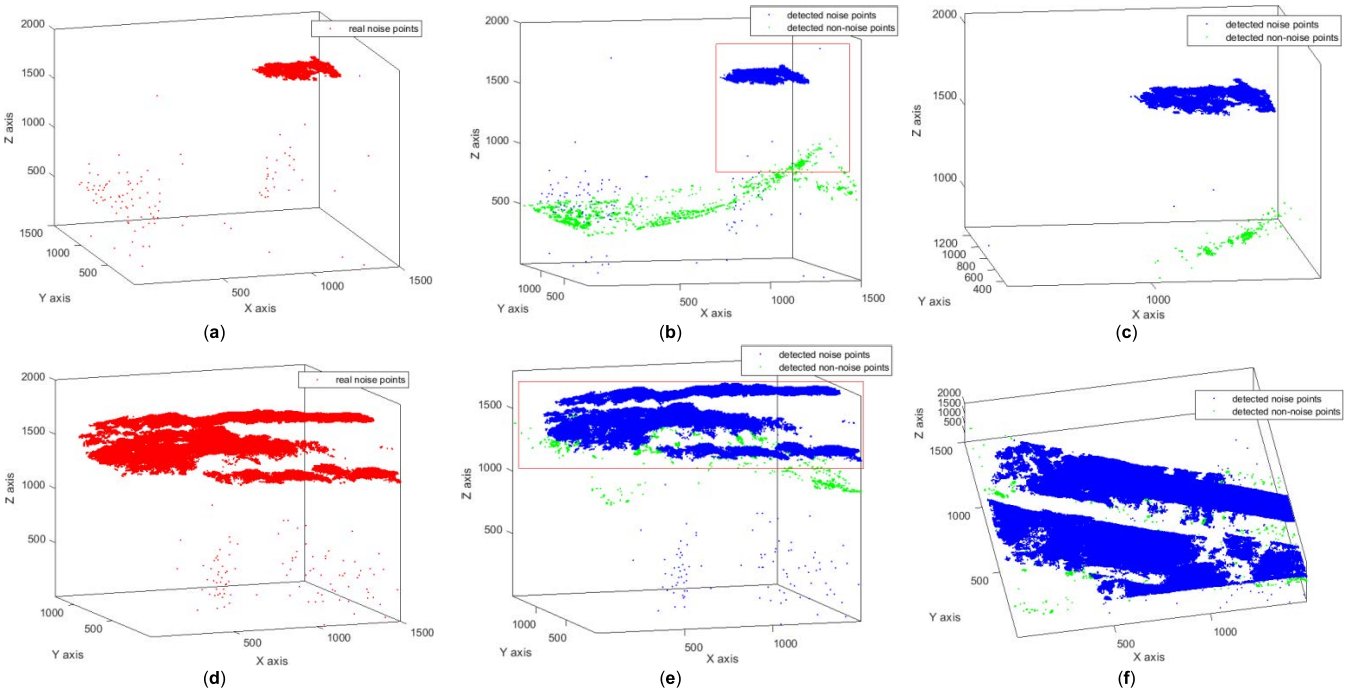


FIGURE 6. Results of noise point detection by the region growing algorithm based on the spatial hierarchical directional relationship (red points represent real noise points, blue ones represent detected noise points and green ones represent signal points that were incorrectly detected as noise points): (a) The original noise point data of S_1. (b) Detected noise points by our method from S_1. (c) Local magnification of red rectangle area in (b). (d) The original noise point data of S_2. (e) Detected noise points by our method from S_2. (f) Local magnification of red rectangle area in (e).

also removes a large number of signal points in the denoising process, so the accuracy is lower than that of the SOR filtering. The middle signal point and cloud noise point have similar point density characteristics. Additionally, the cavity in the signal region leads the algorithm to eliminate noise points and filter the small and medium density point sets in the signal region. This means that the number of signal points are greater than the number of noise points being filtered, so the precision rates are low.

In the application of DBSCAN, only the points on some clouds can be removed and the isolated points cannot be removed, so the precision is 100% and the recall is only 47.9%. The accuracy of the evaluation results of the entire study area using the Euclidean cluster extraction method is second only to our method. Euclidean clustering extraction is widely used in point cloud clustering and segmentation, but lacks of direct semantic information and automaticity for noise points identification and extraction.

The region growing algorithm is generally higher than other types of denoising algorithms. The processing precision reaches 68.6%, and the recall reaches 100.0% in S_1. This is because the algorithm takes the signal region as the starting point to grow until it grows to the boundary between the signal region and the noise region. Therefore, all the noise points are filtered, and the recall reaches 100%. However, some signal points that do not meet the threshold conditions are also filtered. The precision and recall of the study area reached 98.8% and 94.1% in S_2, respectively.

Compared with the region growing algorithm based on the normal vector and curvature, our method based on distance has better detection effect and higher overall precision. Notably, the time used to detect noise points in the region growing algorithm based on the spatial hierarchical direction is half of the running time of the region growing algorithm based on a normal vector, which has higher efficiency. The distribution of noise points in $\boldsymbol{S_1}$ is relatively simple, but the detection precision is less than that in $\boldsymbol{S_2}$ because the top of the building and the tree point cloud lacking in another area in $\boldsymbol{S_1}$ are detected as noise points, so some signal points in the noise point class decrease the precision.

V. DISCUSSION

A. COMPARATIVE ANALYSIS

In the introduction section, current methods applied to eliminate noise points of point cloud data are presented. The characteristics of various types of noise, methods involving features, denoising methods and relevant literature indicators are shown in TABLE 3. Several algorithms that have some applicability to eliminate the experimental area in this paper are used for comparative experiments.

Visual qualitative analysis of the experimental shows that other noise detection algorithms used for comparison have certain issues when processing the experimental data in this paper. In the study area S_1, since the cloud noise is far from the ground point, isolated points and noise clusters can be detected. However, in S_2, the distance between the cloud and the ground point is close, and some cloud noise is mistakenly identified as signal points. Moreover, the three methods applied to the study area S_2

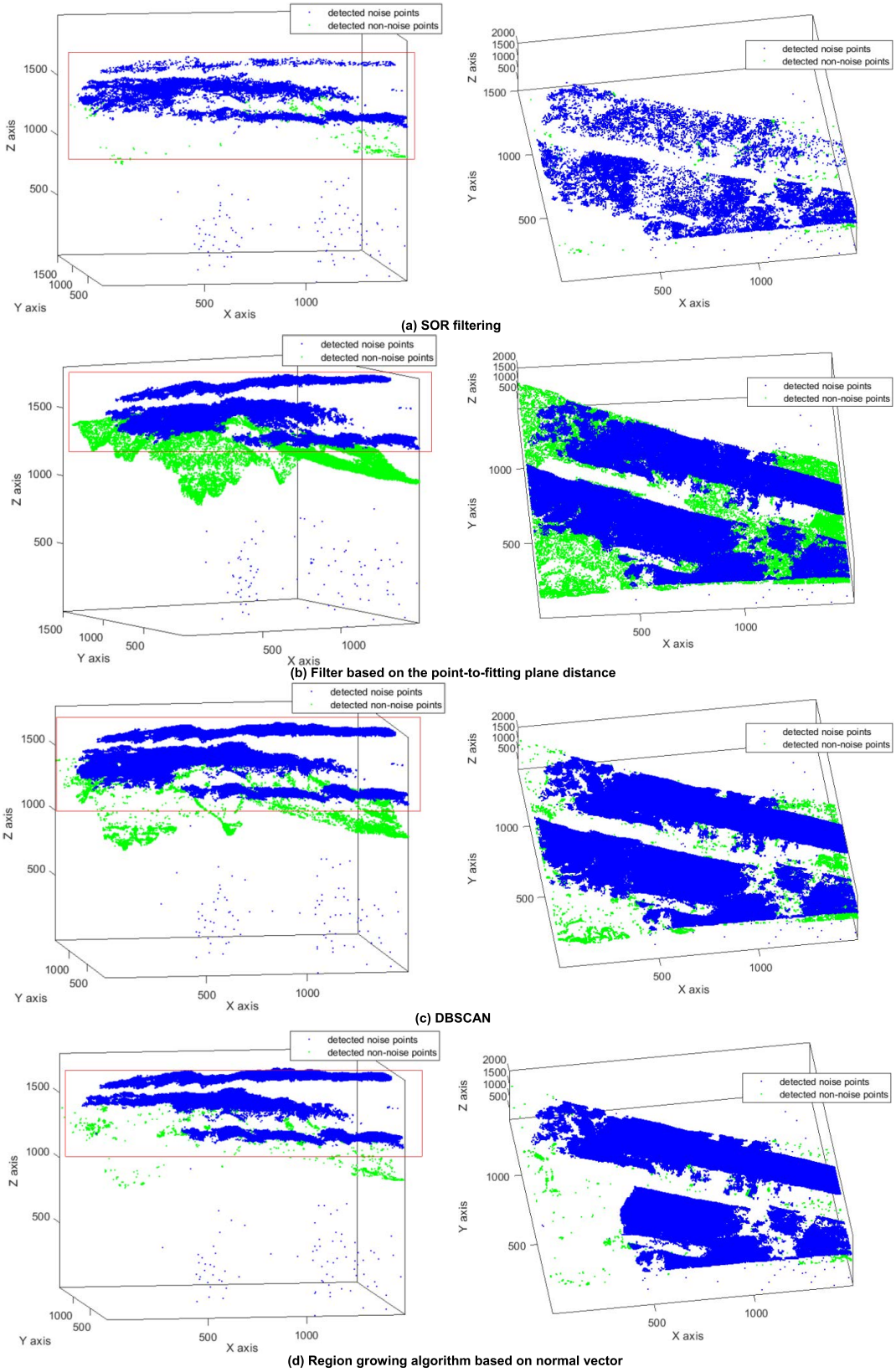
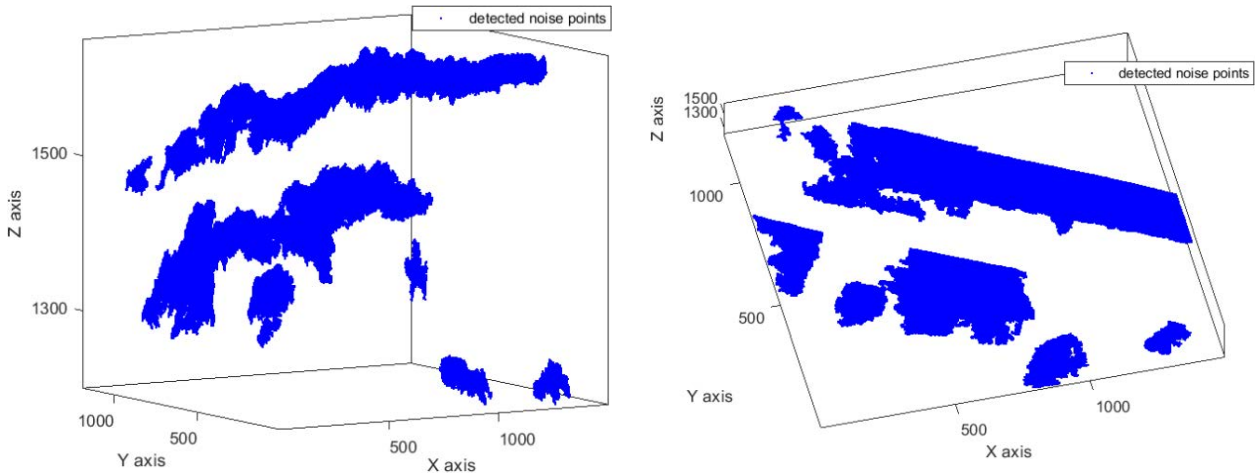


FIGURE 7. Results of noise point detection by comparison methods in S_2 (blue points represent detected noise points and green ones represent signal points that were incorrectly detected as noise points, the red rectangle represent the 3D areas needed to be enlarged for the processing results visualization): The figures on the left are the results of noise points detection by each method, and the corresponding local magnification are on the right column.



(e) Euclidean cluster extraction

FIGURE 7. *(Continued.)* **Results of noise point detection by comparison methods in S_2 (blue points represent detected noise points and green ones represent signal points that were incorrectly detected as noise points, the red rectangle represent the 3D areas needed to be enlarged for the processing results visualization): The figures on the left are the results of noise points detection by each method, and the corresponding local magnification are on the right column.**

TABLE 3. **Summary of point cloud denoising methods.**

Denoising Type	Noise Characteristic	Methods Involved Features	Method Name	Related Bibliographical Indices
Overall environment denoising	Objects floating in the air and scattering points	The relationship between points or between point sets and point sets	Statistical outlier removal (SOR) filter	[18], [20]
			Spatial frequency outlier filter	[19]
			Radius outlier removal filtering	[12]
			Density-based spatial clustering application with noise (DBSCAN)	[21-23]
			Principal component analysis	[14], [24-26]
Special noise points eliminating	Meteorological factors (solid liquid particles condensed in clouds, snow and other air)	Characteristics of meteorological factors and the farther the distance from the rangefinder is, the lower the density	Snow intensity information	[27-28]
			Dynamic radius outlier removal filtering	[12]
Ground point filtering	Noise unrelated to terrain	Surface or planarity of the terrain	Elevation histogram	[13], [35-36]
			Data structure (adaptive TIN, Quadtree)	[37-38]
			Surface fitting (morphology and interpolation)	[29-33]
			Cloth simulation filtering	[39-40]
			Region growing algorithm	[41-47]

in a complex environment all have the problem of filtering partial isolated signal points. The algorithm demonstrates different effects in detecting isolated points and noise clusters.

1) TWO METHODS, INCLUDING SOR FILTERING AND FILTERING BASED ON THE POINT-TO-FITTING PLANE DISTANCE, ARE MORE APPLICABLE FOR ISOLATED DISCRETE POINT DETECTION

When the SOR filter method is applied to the points in the discrete point set, the average distance of the nearest neighbor is much larger than the average distance of the points under the Gaussian distribution, so it is recognized as a noise point. The cloud over the study area belongs to the noise cluster. It can be seen from the graph that the point distribution of the cloud center is dense and only the edge points are sparse, so the average distance of most points in the cloud is not detected in the Gaussian distribution confidence interval of the dataset, which leads to the poor overall cloud points removal effect of this algorithm. The point in the cloud has a similar average distance range as the signal point. If the cloud cluster is removed by debugging the parameters, then a large range of signal points will be mistakenly identified as noise points. In addition, some isolated signal points are formed due to cloud occlusion in the study area S_2. According to the principle of the algorithm, the local average distance of isolated signal points is larger than the threshold and is eliminated.

The method of filtering based on distance from the point to the fitting plane is similar to the SOR filter, which is equivalent to a low-pass filter. The nearest neighbor search method can also be used to search through the spherical neighborhood, which is also applied to calculate the distance of local neighborhood points. However, this method locally

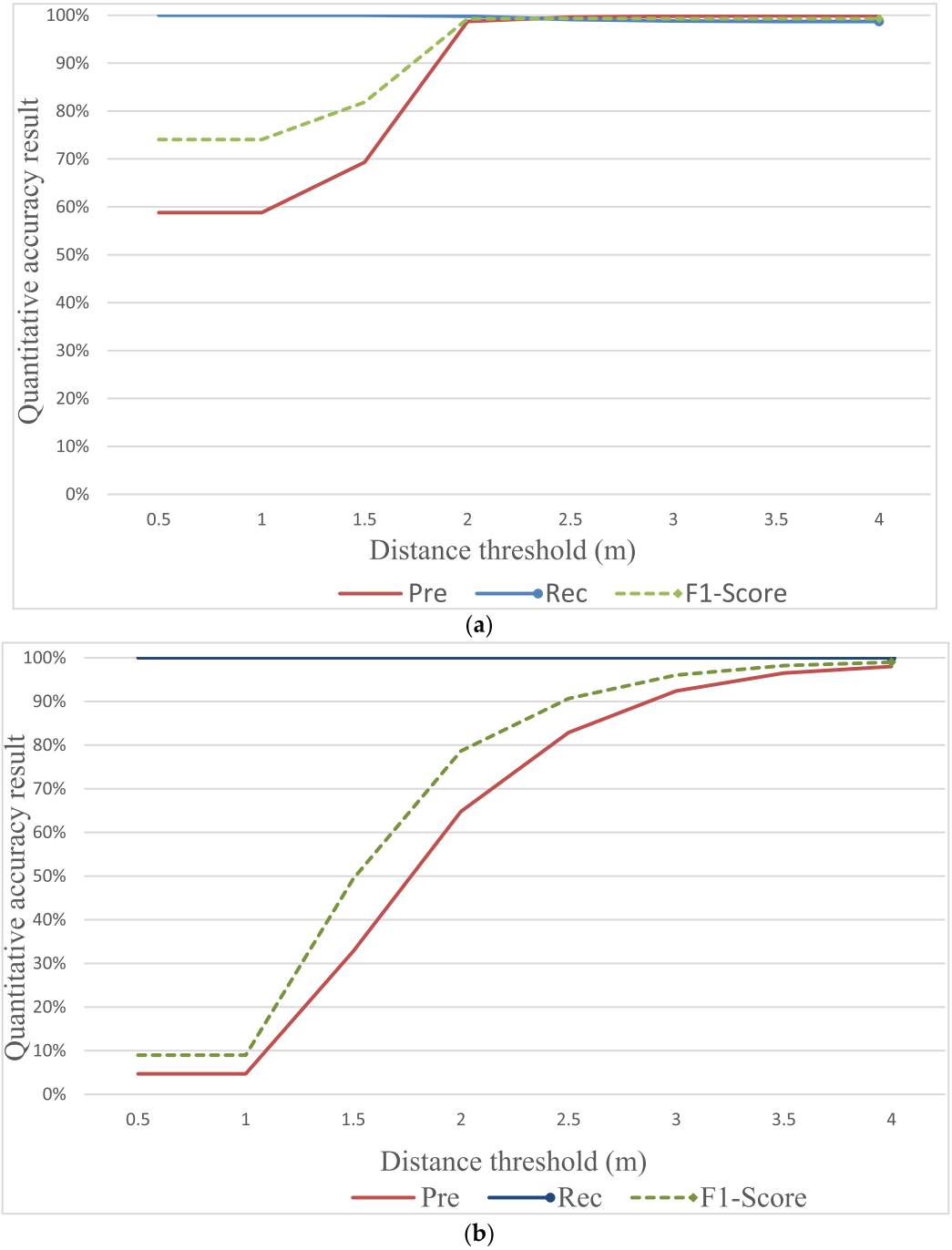


FIGURE 8. Parameter sensitivity analysis: (a) sensitivity analysis of the distance parameter in S_1 and (b) sensitivity analysis of the distance parameter in S_2.

fits a plane through the target point and its neighborhood points. If the distance between the target point and the plane is too far, then it is removed. There are two kinds of distance thresholds: relative distance and absolute distance. The relative distance is a 2D distance from the point projection to the plane where the fitting plane is located, which is usually used to mitigate noise on the same horizontal plane. Absolute distance is the Euclidean distance in space. The noise points in this study area are concentrated in 3D space, and the absolute distance threshold is mainly used. In the spherical search centered on isolated points, the points in the signal point region fit the plane, so the distance from the isolated discrete point to the plane is greater than the threshold. However, the noise cluster is similar to the reasons described in the above SOR filter. The dense cloud midpoint leads to the close distance between the fitting plane and the cloud point, and most points are retained, which cannot achieve good denoising effect.

2) DBSCAN AND EUCLIDEAN CLUSTER EXTRACTION ARE MORE APPLICABLE FOR CLOUD NOISE CLUSTER DETECTION

In DBSCAN, the spherical neighborhood with radius threshold is used to search the point density in the sphere, and the minimum neighborhood point number threshold is set to determine whether it is a noise point. In the study area S_1, some isolated points are not correctly classified because they belong to the boundary points in the point set cluster with signal points as the core, but some cloud noise belonging to the number of low point sets is detected after setting the point density threshold. The distance between the discrete point and the signal area in the study area S_2 is greater than the direct distance of the density of the core point, so it will not be classified as the boundary point. The point density of the cloud noise cluster in S_2 is similar, and it is divided into different categories from the signal point area. Two types of noise points can be detected by setting the point density threshold.

Euclidean cluster extraction has a similar principle to DBSCAN, which also classifies points with the same neighborhood into one class through a spherical neighborhood search, and divides point clouds into point sets of different categories. However, the difference is that DBSCAN can group isolated points into one class, while Euclidean cluster extraction divides LiDAR points into different clusters.

First, the direct segmentation results of Euclidean clustering extraction are several clusters, which lack of semantic information. The prior knowledge or manual visual interpretation is required to identify the meaning of each segmented cluster block. The results in Table 2 were computed after simple manual post-processing, such as grouping some clusters together. Second, this method involves many parameters, such as according to the setting threshold of distance parameter, the minimum and maximum points in each cluster. These parameters need to be determined by professional experience, and also depends on the characteristics of the point cloud data itself. The determination of parameters is also a time-consuming and trial-and-error process. The manual post-processing may introduce subjective error. Finally, the Euclidean cluster extraction needs to iterate over all the points, which can be a bit inefficient. Therefore, the Euclidean cluster extraction performed slightly less efficiency as lacking of semantic information and automaticity for noise point detection in this study.

3) THE PROPOSED REGION GROWING ALGORITHM BASED ON SPATIAL HIERARCHICAL DIRECTIONAL RELATIONSHIPS IS MORE SUITABLE FOR BOTH TYPES OF NOISE POINT DETECTION THAN OTHER METHODS

The region growing algorithm based on the normal vector also has a good detection effect for two kinds of noise points, but only in the study area S_1 is this noise far from the signal area above and below. In the study area S_2, cloud noise surrounded the mountains, and the curvature of the cloud point at the junction of the signal area and cloud noise was less than the threshold, which led to the seed point growing to some clouds, so some noise points in the second area were not detected.

There is also a case of dividing a small number of signal points into noise points in this method, that is, the error detection of some building top surfaces and tree signal points. These two types of point clouds have a certain distance from the ground point cloud due to the scanning angle of the airborne LiDAR system and do not have spatial connectivity. When the radius threshold of the algorithm is less than the distance between the ground point and the building top surface or the tree point cloud, the seed point cannot grow to these two types, and the top surface of the building and the tree are marked as noise points.

However, with the processing algorithm of point cloud data with a large range and large amount of data in this paper, the signal points detected by this method are far less than the signal points removed by the region growing algorithm based on the normal vector, which has the best processing effect on S_1 in the comparative experiment, and the reserved signal point characteristics are also the most complete. In the study area S_2, there are two algorithms that have the similar detection effect in this paper: the region growing algorithm based on normal vector and the DBSCAN method, but the computational time of our proposed method in dealing with S_1 and S_2 is approximately $1/3$ and $1/10$ of these two algorithms, respectively. Compared with the Euclidean clustering extraction, our method just determined by 14 main directions within one point's neighborhood sphere instead of the whole neighbor points. Furthermore, our method can be further expended to Hausdorff distance, direction, connectivity threshold etc., as representation of 3D spatial heterogeneity for special tasks. Therefore, the proposed method has the best detection results for both isolated points and noise clusters, making it applicable to noise point detection from airborne LiDAR point clouds.

It is worth noting that both the method proposed in this paper and the method proposed in comparative analysis have error detection, because when noise points and signal points are near each other, they cannot be distinguished by geometric features.

B. PARAMETER SENSITIVITY ANALYSIS

In this paper, the spatial hierarchical directional relationship is reflected by the distance threshold and growth direction when using the regional growth algorithm based on the spatial hierarchical directional relationship. The definition of growth direction avoids finding candidate points irregularly by seed points and grows with the spatial directional relationship between seed points and candidate points. Each point is assigned to different point sets by a distance threshold (search radius of the spherical neighborhood). At the point set level, each point set is transformed into a point unit, and the unit that meets the threshold condition is connected to the whole point set by the distance between the points.

The setting of the distance threshold determines whether there is a spatial connectivity relationship between the noise point and the signal point. If the distance threshold is too large, then the point set belonging to the noise point intersects with the signal point set to generate directional connectivity, which makes the method in this paper unable to detect the noise point. If the distance threshold is too small, then some signal point sets are not connected, resulting in incomplete growth of seed points. Therefore, the setting of the distance threshold needs to be discussed when the main growth direction is determined.

In this paper, the best distance threshold is set to 3 m. To search for the most suitable distance threshold for the study area to achieve the best detection effect, a sensitivity analysis of the parameters in S_1 and S_2 is carried out. The sensitivity trend diagram of the parameters in S_1 and S_2 is shown in FGURE 8.

The precision rate and $F1-score$ in the study areas S_1 and S_2 increase with increasing radius threshold. When reaching a certain threshold, the precision rate and $F1-score$ tend to 100%, and the recall rate has been tending to 100%, indicating that this method can completely detect noise points. The precision increases with the radius, indicating that the radius parameter will affect the number of signal points misclassified into noise points. However, theoretically, when the radius reaches a certain value, the seed point will grow on the noise cluster, and then the noise point will not be detected.

It can be seen from FIGURE 8 that in S_1, when the radius reaches 2 m, the growth trend of the precision and $F1-score$ is close to flat, and when the radius is 3 m, the accuracy is the highest. In S_2, when the radius reaches 3 m, the growth trend is gentle, and the precision reaches the highest when the radius is 4 m. In the process of the algorithm experiment, we found that with increasing radius, the running memory required by this algorithm is larger, and the running time is longer. Considering the efficiency of the algorithm, the optimal radius threshold of the two research areas is 3 m. When the radius is 3 m, the precision of S_1 is the highest. Although the precision of S_2 is not the highest under this parameter, the running time is low. Compared with other radius thresholds, 3 m is the most efficient distance threshold.

For the study areas in this study, it is still necessary to discuss further the detection errors beside the almost perfect accuracy results. The numbers of noise points and signal points in the study areas are large, objective evaluation results cannot be obtained only from the missed detection rates and error rates.

Combined with the qualitative analysis, the main error in S_1 lies in detecting the points of canopy and building top as noise points. Because the distances between these points and other signal points are probably greater than 3 m, our method marked them as noise points. This error is small and can be reduced by increasing the distance threshold. It can be seen from FIGURE 8(a) that even if the distance threshold is increased, the accuracy is not significantly improved and the processing time is longer. Therefore, we chose the more balanced threshold of 3 m. Of course, it is also feasible to preserve characteristic points such as tree canopy and building top by increasing the distance threshold in S_1, in which noise points and signal points are distinct. However, this increase will cause obvious errors in S_2, in which the noise cluster is closer to the signal region, and the threshold should be carefully considered. Because the seed points will grow towards the cloud noise cluster and the reserved signal points will also contain more cloud noise points with the increasing of distance threshold. Therefore, the distance threshold of 3 m is also the optimal parameter in S_2, while increasing the threshold further may result in more missed noise points.

S_1 and S_2 results of the recall rate is near 100% because the recall rate calculation principle is detected detected noise point accounted for the proportion of real noise, our approach to signal region growing, extracted when d threshold hours points in the region of the signal, instead of, all the noise points and part of the signal points are detected, Therefore, the algorithm determines that all noise points have been detected and the calculated recall rate is 100%. It can be seen from FIGURE 8(a) that as threshold d increases, it gradually grows to the area of noise points, and the recall rate decreases accordingly.

VI. CONCLUSION

In this paper, we propose a method to automatically detect noise points from airborne point cloud data in mountainous landscapes. The region growing algorithm with distance as the growth threshold is used to detect cloud noise clusters and discrete point noise in mountain range point cloud data, and the main direction of seed point growth is determined to improve the efficiency of detecting noise points and avoid growing into a dead cycle. The main contributions of this method are as follows: The region growing algorithm based on spatial hierarchical direction has high computational efficiency and considerable quality, and the precision, recall rate and $F1$ score of the proposed method for detecting noise points (including cloud point clusters and isolated points) reached 99.8%, 100% and 99.3%, respectively. The region growing algorithm with distance as the growth threshold can clearly separate the noise region and the signal region. However, the setting of the distance threshold may eliminate some signal points, such as the top of the building and trees. To balance the accurate detection and calculation efficiency of noise points caused by the distance threshold, the robustness and efficiency of this method need further research.

During this study, we noticed that the method for detecting noise points based on the distance feature is applicable in mountain scenes because the noise points are distributed far away from the scene. However, when the method in this paper is applied to urban scenes, the noise inside the urban scenes is detected by the distance feature, such as the noise point near the signal (the noise point produced by snow or fog), which often leads to an incorrect prediction. Therefore, finding the deep-seated characteristics of noise points will be our follow-up research direction. Considering that the

characteristics of single point cloud data are not significant, the fusion of multisource data is a valuable means of extracting features in the future.

REFERENCES

[1] L. Giorgi, G. Bruno, J. Nijhof, P. J. Urban, G. Vall-llosera, F. Ponzini, and J. Ladvanszky, "Subcarrier multiplexing RF plans for analog radio over fiber in heterogeneous networks," *J. Lightw. Technol.*, vol. 34, no. 16, pp. 3859–3866, Aug. 15, 2016, doi: 10.1109/JLT.2016.2581883.

[2] P. Klápště, M. Fogl, V. Barták, K. Gdulová, R. Urban, and V. Moudrý, "Sensitivity analysis of parameters and contrasting performance of ground filtering algorithms with UAV photogrammetry-based and LiDAR point clouds," *Int. J. Digit. Earth*, vol. 13, no. 12, pp. 1672–1694, Dec. 2020, doi: 10.1080/17538947.2020.1791267.

[3] C. Wiseman, M. O'Leary, J. Hacker, F. Stankiewicz, J. McCarthy, E. Beckett, J. Leach, P. Baggaley, C. Collins, S. Ulm, J. McDonald, and J. Benjamin, "A multi-scalar approach to marine survey and underwater archaeological site prospection in murujuga, western Australia," *Quaternary Int.*, vol. 584, pp. 152–170, May 2021, doi: 10.1016/j.quaint.2020.09.005.

[4] I. Sarakinou, K. Papadimitriou, O. Georgoula, and P. Patias, "Underwater 3D modeling: Image enhancement and point cloud filtering," *ISPRS Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. B2, pp. 441–447, Jun. 2016, doi: 10.5194/isprsarchives-XLI-B2-441-2016.

[5] C. Yu, F. Ji, and J. Xue, "Cutting plane based cylinder fitting method with incomplete point cloud data for digital fringe projection," *IEEE Access*, vol. 8, pp. 149385–149401, 2020, doi: 10.1109/ACCESS.2020.3016424.

[6] D. Xu, H. Wang, W. Xu, Z. Luan, and X. Xu, "LiDAR applications to estimate forest biomass at individual tree scale: Opportunities, challenges and future perspectives," *Forests*, vol. 12, no. 5, p. 550, Apr. 2021, doi: 10.3390/f12050550.

[7] J. Lu, H. Wang, S. Qin, L. Cao, R. Pu, G. Li, and J. Sun, "Estimation of aboveground biomass of robinia pseudoacacia forest in the yellow river delta based on UAV and backpack LiDAR point clouds," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 86, Apr. 2020, Art. no. 102014, doi: 10.1016/j.jag.2019.102014.

[8] H. Guan, X. Sun, Y. Su, T. Hu, H. Wang, H. Wang, C. Peng, and Q. Guo, "UAV-LiDAR aids automatic intelligent powerline inspection," *Int. J. Electr. Power Energy Syst.*, vol. 130, Sep. 2021, Art. no. 106987, doi: 10.1016/j.ijepes.2021.106987.

[9] Y. He, L. Dong, F .Zeng, C. Dong, and J. Yao, "Power lines extraction using UVA LiDAR point clouds in complex terrains and geological structures," in *Proc. IOP Conf. Earth Environ. Sci.*, vol. 804, Jul. 2021, Art. no. 032053, doi: 10.1088/1755-1315/804/3/032053.

[10] J. A. Issa, "Processing time analysis and estimation for 3-D applications," *IEEE Access*, vol. 2, pp. 1177–1186, 2014, doi: 10.1109/ACCESS.2014.2363113.

[11] T. Yun, K. Jiang, G. Li, M. P. Eichhorn, J. Fan, F. Liu, B. Chen, F. An, and L. Cao, "Individual tree crown segmentation from airborne LiDAR data using a novel Gaussian filter and energy function minimization-based approach," *Remote Sens. Environ.*, vol. 256, Apr. 2021, Art. no. 112307, doi: 10.1016/j.rse.2021.112307.

[12] N. Charron, S. Phillips, and S. L. Waslander, "De-noising of LiDAR point clouds corrupted by snowfall," in *Proc. 15th Conf. Comput. Robot Vis. (CRV)*, May 2018, pp. 254–261.

[13] A. C. Carrilho, M. Galo, and R. C. Santos, "Statistical outlier detection method for airborne LiDAR data," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 1, pp. 87–92, Sep. 2018, doi: 10.5194/isprs-archives-XLII-1-87-2018.

[14] Y. Duan, C. Yang, H. Chen, W. Yan, and H. Li, "Low-complexity point cloud denoising for LiDAR by PCA-based dimension reduction," *Opt. Commun.*, vol. 482, Mar. 2021, Art. no. 126567, doi: 10.1016/j.optcom.2020.126567.

[15] G. Sithole and G. Vosselman, "Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 59, nos. 1–2, pp. 85–101, Aug. 2004, doi: 10.1016/j.isprsjprs.2004.05.004.

[16] X. Zewei, W. Jie, and C. Xianqiao, "Vehicle recognition and classification method based on laser scanning point cloud data," in *Proc. Int. Conf. Transp. Inf. Saf. (ICTIS)*, Jun. 2015, pp. 44–49.

[17] S. D. Deshpande, M. H. Er, R. Venkateswarlu, and P. Chan, "Max-mean and max-median filters for detection of small targets," *Proc. SPIE*, vol. 3809, pp. 74–83, Oct. 1999.

[18] R. B. Rusu, "Semantic 3D object maps for everyday manipulation in human living environments," *Künstliche Intelligenz*, vol. 24, no. 4, pp. 345–348, Nov. 2010, doi: 10.1007/s13218-010-0059-6.

[19] J. Jia-Jia, W. Xian-Quan, and D. Fa-Jie, "An effective frequency-spatial filter method to restrain the interferences for active sensors gain and phase errors calibration," *IEEE Sensors J.*, vol. 16, no. 21, pp. 7713–7719, Nov. 2016, doi: 10.1109/JSEN.2016.2602887.

[20] H. Balta, J. Velagic, W. Bosschaerts, G. De Cubber, and B. Siciliano, "Fast statistical outlier removal based method for large 3D point clouds of outdoor environments," *IFAC-PapersOnLine*, vol. 51, no. 22, pp. 348–353, 2018, doi: 10.1016/j.ifacol.2018.11.566.

[21] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. AAAI Press*, 1996, pp. 226–231.

[22] J. Zhang and J. Kerekes, "An adaptive density-based model for extracting surface returns from photon-counting laser altimeter data," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 4, pp. 726–730, Apr. 2015, doi: 10.1109/LGRS.2014.2360367.

[23] X. Zhu, S. Nie, C. Wang, X. Xi, J. Wang, D. Li, and H. Zhou, "A noise removal algorithm based on OPTICS for photon-counting LiDAR data," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 8, pp. 1471–1475, Aug. 2021, doi: 10.1109/LGRS.2020.3003191.

[24] D. Cheng, D. Zhao, J. Zhang, C. Wei, and D. Tian, "PCA-based denoising algorithm for outdoor lidar point cloud data," *Sensors*, vol. 21, no. 11, p. 3703, May 2021, doi: 10.3390/s21113703.

[25] X. Wang, C. Glennie, and Z. Pan, "An adaptive ellipsoid searching filter for airborne single-photon lidar," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 8, pp. 1258–1262, Aug. 2017, doi: 10.1109/LGRS.2017.2704917.

[26] E. Mattei and A. Castrodad, "Point cloud denoising via moving RPCA," *Comput. Graph. Forum*, vol. 36, no. 8, pp. 123–137, Dec. 2017, doi: 10.1111/cgf.13068.

[27] R. Heinzler, F. Piewak, P. Schindler, and W. Stork, "CNN-based LiDAR point cloud de-noising in adverse weather," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2514–2521, Apr. 2020, doi: 10.1109/LRA.2020.2972865.

[28] J. I. Park, J. Park, and K. S. Kim, "Fast and accurate desnowing algorithm for LiDAR point clouds," *IEEE Access*, vol. 8, pp. 160202–160212, 2020, doi: 10.1109/access.2020.3020266.

[29] X. Zhao, Q. Guo, Y. Su, and B. Xue, "Improved progressive TIN densification filtering algorithm for airborne LiDAR data in forested areas," *ISPRS J. Photogramm. Remote Sens.*, vol. 117, pp. 79–91, Jul. 2016, doi: 10.1016/j.isprsjprs.2016.03.016.

[30] X. Meng, N. Currit, and K. Zhao, "Ground filtering algorithms for airborne LiDAR data: A review of critical issues," *Remote Sens.*, vol. 2, no. 3, pp. 833–860, Mar. 2010, doi: 10.3390/rs2030833.

[31] Q. Chen, P. Gong, D. Baldocchi, and G. Xie, "Filtering airborne laser scanning data with morphological methods," *Photogramm. Eng. Remote Sens.*, vol. 73, no. 2, pp. 175–185, Feb. 2007, doi: 10.14358/PERS.73.2.175.

[32] K. Zhang, S.-C. Chen, D. Whitman, M.-L. Shyu, J. Yan, and C. Zhang, "A progressive morphological filter for removing nonground measurements from airborne LiDAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 4, pp. 872–882, Apr. 2003, doi: 10.1109/tgrs.2003.810682.

[33] X. Shi, H. Ma, Y. Chen, L. Zhang, and W. Zhou, "A parameter-free progressive TIN densification filtering algorithm for LiDAR point clouds," *Int. J. Remote Sens.*, vol. 39, no. 20, pp. 6969–6982, Oct. 2018, doi: 10.1080/01431161.2018.1468109.

[34] Z. Chen, B. Gao, and B. Devereux, "State-of-the-art: DTM generation using airborne LiDAR data," *Sensors*, vol. 17, no. 12, p. 150, Jan. 2017, doi: 10.3390/s17010150.

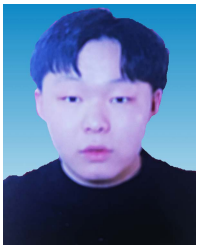
[35] F. Li, Z. Yu, B. Wang, and Q. Dong, "Filtering algorithm for LiDAR outliers based on histogram and KD tree," in *Proc. 4th Int. Congr. Image Signal Process.*, Oct. 2011, pp. 2741–2745.

[36] J. Shen, J. Liu, R. Zhao, and X. Lin, "A Kd-tree-based outlier detection method for airborne LiDAR point clouds," in *Proc. Int. Symp. Image Data Fusion*, Aug. 2011, pp. 1–4.

[37] N. Baş, H. G. Coşkun, and H. Çelik, "Outlier detection analysis from LiDAR data for mapping," in *Proc. 7th Int. Conf. Cartography (GIS)*, 2018, pp. 18–23.

[38] G. Zhang, Q. Xu, S. Xing, P. Li, X. Zhang, D. Wang, and M. Dai, "A noise-removal algorithm without input parameters based on quadtree isolation for photon-counting LiDAR," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/lgrs.2021.3081721.

[39] W. Zhang, J. Qi, P. Wan, H. Wang, D. Xie, X. Wang, and G. Yan, "An easy-to-use airborne LiDAR data filtering method based on cloth simulation," *Remote Sens.*, vol. 8, no. 6, p. 501, Jun. 2016, doi: 10.3390/rs8060501.
[40] A. Yang, Z. Wu, F. Yang, D. Su, Y. Ma, D. Zhao, and C. Qi, "Filtering of airborne LiDAR bathymetry based on bidirectional cloth simulation," *ISPRS J. Photogramm. Remote Sens.*, vol. 163, pp. 49–61, May 2020, doi: 10.1016/j.isprsjprs.2020.03.004.
[41] L. Li, J. Yao, J. Tu, X. Liu, Y. Li, and L. Guo, "Roof plane segmentation from airborne LiDAR data using hierarchical clustering and boundary relabeling," *Remote Sens.*, vol. 12, no. 9, p. 1363, Apr. 2020, doi: 10.3390/rs12091363.
[42] P. J. Besl and R. C. Jain, "Segmentation through variable-order surface fitting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 2, pp. 167–192, Mar. 1988, doi: 10.1109/34.3881.
[43] J. Shao, W. Zhang, A. Shen, N. Mellado, S. Cai, L. Luo, N. Wang, G. Yan, and G. Zhou, "Seed point set-based building roof extraction from airborne LiDAR point clouds using a top-down strategy," *Autom. Construction*, vol. 126, Jun. 2021, Art. no. 103660, doi: 10.1016/j.autcon.2021.103660.
[44] B. Gorte, "Segmentation of TIN-structured surface models," in *Proc. Joint Int. Symp. Geospatial Theory Process. Appl. CD-ROM*, vol. 34, no. 4, 2002, pp. 465–469.
[45] R. Cao, Y. Zhang, X. Liu, and Z. Zhao, "Roof plane extraction from airborne LiDAR point clouds," *Int. J. Remote Sens.*, vol. 38, no. 12, pp. 3684–3703, Jun. 2017, doi: 10.1080/01431161.2017.1302112.
[46] S. A. N. Gilani, M. Awrangjeb, and G. Lu, "Segmentation of airborne point cloud data for automatic building roof extraction," *GISci. Remote Sens.*, vol. 55, no. 1, pp. 63–89, Jan. 2018, doi: 10.1080/15481603.2017.1361509.
[47] T. Wu, X. Hu, and L. Ye, "Fast and accurate plane segmentation of airborne LiDAR point cloud using cross-line elements," *Remote Sens.*, vol. 8, no. 5, p. 383, May 2016, doi: 10.3390/rs8050383.



YUNHAO LIN received the B.S. degree in geographic information science from the China University of Mining and Technology, Xuzhou, Jiangsu, in 2020. He is currently pursuing the master's degree with the Hunan Provincial Key Laboratory of Geo-Information Engineering in Surveying, Mapping and Remote Sensing, Hunan University of Science and Technology. His research interests include objects classification, segmentation using point clouds, with a focus on road environment feature extraction, and object classification and recognition based on remote sensing image.

YANJUN WANG was born in Xiangyang, Hubei, China, in 1984. He received the B.S. and M.S. degrees in geographical information science from Wuhan University, in 2007, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, in 2012.

From 2012 to 2016, he was a Lecturer with the Hunan University of Science and Technology, where he was an Associate Professor, from 2017 to 2021. Since 2022, he has been a Professor with the Hunan University of Science and Technology. He is the author of two books and more than 30 articles. His research interests include remote sensing image, LiDAR point cloud data processing, and environmental modeling.

SHUHAN WANG received the Ph.D. degree in forest management from the College of Forestry, Beijing Forestry University, in 2016.

From 2014 to 2015, he was co-trained at Michigan State University. Since 2017, he has been a Lecturer with the Hunan University of Science and Technology. He has published more than ten papers. His research interests include the airborne LiDAR combined with high-resolution remote sensing in forest biomass and urban environment.

SHAOCHUN LI received the B.S. degree in geographical science from Huanggang Normal University, Hubei, China, in 2019, and the M.S. degree in surveying and mapping from the Hunan University of Science and Technology, Hunan, China, in 2022. His research interests include remote sensing image classification, change detection, and intelligent computing of geological information.

MENGJIE WANG received the B.S. degree in land resource management from the Guangdong University of Finance and Economics, Guangzhou, China. He is currently pursuing the master's degree in surveying and mapping science and technology with the Hunan University of Science and Technology, Xiangtan, China.

His research interests include remote sensing data processing and geographic modeling, socio-economic, and urban ecological environment monitoring.

HENGFAN CAI received the B.S. degree in survey and mapping engineering from the China University of Geosciences, Wuhan, in 2019. He is currently pursuing the M.S. degree with the Hunan Provincial Key Laboratory of Geo-Information Engineering in Surveying, Mapping and Remote Sensing, Hunan University of Science and Technology, Xiangtan. His research interests include objects classification and extraction from LiDAR point clouds and crop extraction from remote sensing imagery.

FEI TENG received the B.S. degree in surveying and mapping engineering from Shangqiu Normal University, Shangqiu, China. She is currently pursuing the master's degree in surveying and mapping science and technology with the Hunan University of Science and Technology, Xiangtan, China.

Her research interests include remote sensing data processing and geographic modeling, socio-economic, and urban ecological environment monitoring.