R3MR: Region Growing Based 3D Mesh Reconstruction for Big Data Platform

HONG-AN LI, MIN ZHANG, KEPING YU, (Member, IEEE), XIN QI, QIAOZHI HUA, AND YU ZHU

1College of Computer Science and Technology, Xi’an University of Science and Technology, Xi’an 710054, China
2Global Information and Telecommunication Institute, Waseda University, Tokyo 169-8050, Japan
3Shenzhen Boyi Technology Company Ltd., Shenzhen 518125, China
4Computer School, Hubei University of Arts and Science, Xiangyang 441000, China
5School of Information Science and Technology, Northwest University, Xi’an 710127, China

Corresponding author: Keping Yu (keping.yu@aoni.waseda.jp)

This work was supported in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JM-162 and Grant 2020JM-526, and in part by the Japan Society for the Promotion of Science (JSPS) Grants-in-Aid for Scientific Research (KAKENHI) under Grant JP18K18044.

ABSTRACT Visualization is one of the most intuitive and perceptible ways for information representation in the big data era. As an essential part of the visualization, 3D mesh reconstruction is facing great challenges due to its characteristics of quantity, non-structure, and low-accuracy. The traditional 3D mesh reconstruction method has strict theoretical proof and can be used to reconstruct the surface of the complex topological structure for computer rendering and display. However, it is not suitable to handle a large number of point cloud and noise point cloud in a big data platform because the process is inefficient, low-automation and requires massive calculations. To address this issue, we propose a region growing based 3D mesh reconstruction (R3MR) in the big data platform. Firstly, we divide the large data points into three categories: flat point set, high curvature point set, and boundary point set. The errors of topological structure for 3D meshes usually occur in the place with large curvatures and noise points, so the division of high curvature point set is beneficial to solve the low-accuracy problem in 3D mesh reconstruction. Moreover, the flat points can be treated as one kind of point to avoid repetitive calculations because their features are basically the same. Hence, the division of the flat point set is beneficial to solve the problem of quantity and massive calculations. Secondly, our proposal is to start the mesh reconstruction from the flat point set progressively, because it can obtain the outline of the 3D model. In many scenarios, such as autonomous driving, only the overall outline of the model is required. Finally, during the 3D mesh reconstruction, the inner edge adjacency list and optimal selection principle are set to improve the robustness of the whole system. Simulation experiments show that the proposed 3D mesh reconstruction can naturally reflect the detailed features of objects in the big data platform, especially effective for the scattered point cloud.

INDEX TERMS Point cloud, big data, 3D mesh reconstruction, region growing.

I. INTRODUCTION In the high-tech era, big data comes into being. Just like mining coal, we use reasonable mining costs to mine valuable data in big data. Therefore, with the rapid development of big data, how to obtain high-value content and low-cost mining is more important than the amount of data. This technical revolution has impacted many methods and basic theories, and brought challenges and opportunities for the research of 3D computer mesh [1]. By using special software to carry out mathematical representation on any 3D surface of an object to realize 3D big data modeling, a 3D mesh containing more detailed information can be obtained, compared with a 2D image [2], [3]. Traditional medical imaging techniques can obtain 2D projected images (such as X-ray imaging) or transverse images (such as CT or MRI). The 2D image can only display part of the section, and cannot obtain a clear stereoscopic perception in the 3D space [4]. 3D meshes not only show the anatomy of a specific part of the human body to doctors but also reveal the function of human organs to
a certain degree. 3D meshes are widely used in diagnosis, surgical planning and simulation, virtual endoscopy, anatomy teaching, etc., [5]–[8]. With the rapid development of 3D measurement technology and hardware equipment, it has become a research hotspot to reconstruct 3D mesh from the point cloud data. Moreover, 3D mesh reconstruction of scattered point cloud has been widely used in many fields, such as surface interpolation of scattered points, finite element analysis, computer graphics, and visualization in scientific computing, robot vision, cultural heritage, archaeology, surveying, 3D city models, etc., [9]–[14]. Therefore, it is particularly important to use an appropriate method for 3D mesh reconstruction of the scattered point cloud.

The scattered point cloud is a form of big data, because it contains the characteristics of big data with large quantity of data, no-structure and low-accuracy. Based on Delaunay triangulation, region growing, or implicit surface fitting, 3D mesh reconstruction methods of the scattered point cloud can be divided into several kinds of categories [15]. Delaunay triangulation can get more accurate reconstruction results for most objects. But for some objects with noise or sharp features, it cannot get more satisfactory results, or even cannot be reconstructed [16]. The reconstruction method based on implicit surface fitting can resist the influence of noise in point cloud data and the reconstruction result has a good smoothing effect. Besides, region growing based 3D mesh reconstruction has become a research hotspot because of its processing capability for large-scale point cloud data and strong representation for sharp features, which especially can be applied for big data platforms. Stelldinger [17] proved the concept of spiral edge, that is, starting from the edge of the seed triangle, the optimal point was selected from the ball bounding box with a certain radius. However, it needs to determine the normal vector of the data points, and it is relatively difficult to select the radius of the sphere. Huang et al. [18] projected the data point closed to the active edge into the triangular mesh where the edge is located. Among the visible projection points, the minimum length was used as the criterion to select the optimal point. Lin et al. [19] determined the area affecting the active edge by calculating the uniformity of sampling points. After obtaining a specific set of points, a triangular mesh is constructed by new points and strong representation for sharp features, which especially can be applied for big data platforms.

Given these, firstly, by analyzing the geometric characteristics of points and their K-nearest neighbor, the scattered point cloud data is classified into three categories: boundary point, flat point, and high curvature point. Secondly, based on the optimal point selection criteria as described in [25], we integrate the dihedral angle, “used point” and other qualification conditions into the optimal point selection. Thirdly, the inner edge adjacency list is adopted in the reconstruction process to shorten the time of mesh construction. Finally, according to the order from the flat point set to high curvature point set and then to the boundary point set, the mesh reconstruction is stepwise completed from simple to complex [26]. The simulation results show that the proposed method not only inherits the high efficiency of the region growing method but also significantly reduces the possibility of topology errors in the region with high curvature, which is usually used to describe the complex curved surface mesh. And the sharp features can be displayed naturally.

The structure of this paper is organized as follows. Section I introduces the current research background of 3D mesh reconstruction methods of the scattered point cloud. Section II and III present our novel region growing based 3D mesh reconstruction method of big data platform and how to simplify the triangular mesh. The experiment and analysis are put forwarded in Section IV, and we compare our proposal with other popular methods. Section V concludes this paper.

II. REGION GROWING BASED 3D MESH RECONSTRUCTION IN BIG DATA PLATFORM

A. METHOD-RELATED TERM AND DATA STRUCTURE

Let the scattered point cloud be $P = \{p_i | i = 1, 2, \cdots, n\}$, where $n$ represents the number of points in the point cloud, $p_i$ is a point in the point set. The purpose of the method is to construct a triangular mesh for the point cloud incrementally.

1) RELATED TERM

The terms involved in this paper are as following (As shown in Figure 1):

- **Inner edge**: an edge is shared by two adjacent triangles in a triangular mesh.
- **Interior point**: in a triangular mesh, a point is called an interior point if all its adjacent edges are inner edges.
- **Exterior point**: a point in the point cloud that does not yet connect to the mesh.
- **Active Point**: a point that is neither an interior point nor an exterior point.
- **Active edge**: an edge in a triangular mesh that connects only one adjacent triangle.
- **Search area for an active edge**: a set of K-neighbor points that belong to the two endpoints of the edge.
- **Boundary edge**: there are no exterior points or active points in the search area of this edge.
Dihedral angle: a figure consisting of two half planes starting from a straight line.

2) DATA STRUCTURE
In order to speed up the search of adjacent points, we use 3D mesh method to sort point clouds topologically. In order to effectively manage the point set, the edges and triangles of the triangular mesh have been generated and managed using the following data structure:

- **Point list**: records the data points of the scattered point cloud and sets up a marker bit T to determine whether the point is connected to a triangular mesh.
- **Edge list**: records the sequence number of the two endpoints belonging to each edge in the point list, as well as information about the adjacent triangle that first connected the edge.
- **Active edge queue**: stores edges that can be extended.
- **Triangle list**: records the normal vector information of each triangle and the ordinal number of its vertices in the point list.

In addition, in order to improve the efficiency of mesh reconstruction, a corresponding inner edge adjacency list is set up for each vertex, which is utilized to detect the extensibility of edges.

B. POINT CLASSIFICATION METHOD BASED ON GEOMETRIC DISTRIBUTION OF POINT NEIGHBORHOOD

In this paper, the data points are divided into three categories: boundary point, high curvature point and flat point. The distribution of all kinds of points is shown in Figure 2. As an important step of mesh reconstruction, the classification of data points determines the efficiency and quality of subsequent reconstruction. The data point classification method is as follows:

1) SPACE SUBDIVISION AND NEIGHBORHOOD SEARCH
Generally, the measured data points are scattered and have no topological relationship. In order to facilitate the subsequent search and calculation of points and improve the efficiency of the method, the topological relationship of data points should be established first.

Let the point cloud data be \( P = \{ p_i | i = 1, 2, \cdots, n \} \). Firstly, we establish the minimum external cuboid of point cloud data for space subdivision and neighborhood search [27]. We rasterize it with a given edge length and assign data points to the corresponding mesh. Then, we complete the K-nearest neighbor search of the data points: For any point \( p_i \), we find the mesh corresponding to the point.

From the center of the mesh, expand the search in turn, taking \( k \) points nearest to \( p_i \) as the K-neighborhood of the point \( p_i \). According to the previous experiments, the value of \( K \) is set as 20 and the parameter of mesh edge length is 0.2.

2) PRINCIPLES OF BOUNDARY POINT EXTRACTION
The scattered point cloud can be divided into boundary point and interior point according to the distribution of data points and their K-nearest neighbors [28]. For any point \( p_i \), if it’s K-nearest neighbors are distributed on one edge of the point, \( p_i \) is called a boundary point, otherwise, it is called an internal point.

The method of literature [20] is adopted to divide the boundary points. For any point \( p_i \) and its K-nearest neighbor, let \( \bar{p} \) represent the center of mass of K-nearest neighbors of point \( p_i \), and \( p_j \) is the farthest point from \( p_i \) in the neighborhood. Then the point \( p_i \) as the center of the sphere, \( \| p_i p_j \| \) is as the radius of the sphere. By calculating the ratio \( a_i \) between the barycenter point \( \| p_i \bar{p} \| \) and the farthest point distance.
Thus, we can judge whether any point \( p_i \) is a boundary point. If \( a_i \) value is larger, \( p_i \) is the boundary point. Conversely, if \( a_i \) value is smaller, \( p_i \) is an internal point. Where the Euclidean distance \( d \) between two points \( p_i(x_i, y_i, z_i) \) and \( p_j(x_j, y_j, z_j) \) is:

\[
d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}
\]  

(1)

Through experimental analysis, when \( a = 0.3 \), we can get a better classification effect. This method traverses the point cloud and extracts the boundary point.

3) CLASSIFICATION OF INTERIOR POINTS

After extracting the boundary points, the residual points are further divided into flat point class and high curvature point class according to their neighborhood smoothness and geometric characteristics. The principle for determining whether any point \( p_i \) is a flat point is as follows. The k-neighborhood point of \( p_i \) is the least square plane fitting plane to determine the normal vector \( \vec{n}_i \) of \( p_i \). If the local domain of point \( p_i \) is relatively smooth, the angle between the vector \( \vec{p}_i\vec{p}_j \) and \( \vec{n}_i \) of point \( p_i \) and its adjacent point \( p_j \) should be close to \( \pi /2 \). Thus, we can judge by the sum of dot product of vector \( \vec{p}_i\vec{p}_j \) and \( \vec{n}_i \) of \( p_i \) and its k adjacent points:

\[
S(p_i) = \sum_{j=1}^{k} \left( \frac{\vec{p}_j - \vec{p}_i}{\|\vec{p}_j - \vec{p}_i\|}, \vec{n}_i \right)^2
\]  

(2)

Traverse the data points in the point cloud and define the appropriate threshold \( S_r \). If \( S(p_i) \) is greater than \( S_r \), it is a point of high curvature. The rest are flat points, recording flat point and high curvature point. According to the previous experiments, \( S_r = 0.16 \), we can distinguish flat point and high curvature point. The steps of data point classification are as following:

Step 1. The neighborhood point of \( p_i \) is calculated, and then calculate the barycenter point \( \bar{p} \) of the set of neighborhood points.

Step 2. The ratio between \( \|\vec{p}\bar{p}\| \) and \( \|\vec{p}_i\vec{p}_j\| \) is calculated as the decision value to determine whether \( p_i \) is a boundary point.

Step 3. The whole point cloud is traversed by Step 1 and Step 2, and the boundary decision value of the point is calculated. If the boundary decision value is greater than the threshold value \( a \), it is the boundary point and records it.

Step 4. The least square plane method is applied to determine the normal vector of the remaining points according to their neighborhood.

Step 5. The \( S(p_i) \) of each point is calculated according to Equation (2), if \( S(p_i) \) is less than the threshold \( S_r \), it is a flat point. Otherwise, it is a high curvature point. The classification results of interior points were recorded respectively.

C. CONSTRUCTION OF TRIANGULAR MESH

Region growing based 3D mesh reconstruction in big data platform starts with “seed” triangle when constructing the mesh, in the order from the smooth point set to the high curvature point set and finally the boundary point set. That is to say, based on the method of selecting the optimal point in literature [25] from simple to complex, the dihedral angle and “used points” are added, and the corresponding inner edge adjacency list is set up to complete the mesh reconstruction of point cloud.

1) SELECTION OF INITIAL TRIANGLES

The initial triangle, the seed triangle, is selected in the scattered point cloud, and the three edges of the triangle are added to the active edge list as the initial edge list. The method of constructing “seed” triangle is as follows:

Step 1. Select a point \( p_1 \) within the flat point of the point cloud.

Step 2. The point \( p_2 \) nearest to \( p_1 \) is selected, and its \( p_1 \) and \( p_2 \) are regarded as the first edge of the initial triangle.

Step 3. In order to find the point \( p_3 \) in the K-nearest neighbor of \( p_1 \) and \( p_2 \), the triangle \( \Delta p_1 p_2 p_3 \) conforms to the principle of minimum-maximum inner angle.

Step 4. Update point list, active edge list and triangle list.

At this point, triangle \( \Delta p_1 p_2 p_3 \) is the seed triangle, which is the foundation of mesh reconstruction.

2) DETERMINATION OF INNER EDGE ADJACENCY LIST

In the process of mesh reconstruction, it is necessary to detect the extensibility of edges. If an edge has connected with two triangles, that is, the edge belongs to the inner edge, and then a new triangle expansion is carried out, which does not conform to the streamline principle of mesh. Hence, before expanding the edge of the active edge queue, it is necessary to decide whether the edge is an inner edge. If so, the edge is added from the active edge queue to the edge queue, and the next edge of the active edge queue is reconstructed accordingly; Otherwise, the active area of the edge is searched so that the current mesh continue to expand outward.

In order to test the expansibility of edges, the following strategies are adopted in this paper: If \( \vec{p}_i\vec{p}_j \) is an inner edge, or if \( \vec{p}_i\vec{p}_j \) connects two triangles at the same time, \( p_i, p_j \) are inner edge adjacent points to each other. Based on the above characteristics, the inner edge adjacency list is established for each data point. By establishing the inner edge adjacency list, we can easily learn the extensibility of the edge. For example, if point \( p_i \) is in the inner edge adjacency list of point \( p_j \), it means that \( p_i p_j \) is an inner edge and no longer operates on that edge.

3) THE PRINCIPLE OF SELECTING THE OPTIMAL POINT

In the triangular mesh reconstruction based on region growing, if the optimal point can be reasonably selected, it will have an important impact on the subsequent accurate display of the topological relationship of point cloud data and the improvement of the efficiency of the method. In literature [25], when selecting new data points to join the mesh, it is considered that the optimal point is that the sum of the distances between the two endpoints of the boundary should be small, and the minimum inner angle of the new triangle
is larger than $\pi/6$ and the maximum inner angle is less than $\pi/2$. In order to narrow the range of the optimal selection, the bounding box technique is applied in this method: The longest edge of the current triangle is selected 2.5 times as the edge length of the bounding box, and constructs two bounding boxes centering on the two endpoints of the boundary edge. The range of optimal selection is greatly reduced through the bounding box, thus the partition speed is improved.

The method in literature [25] can quickly reconstruct the mesh of the object model to a certain extent, but its mesh will have the following problems: (1) The dihedral angle between the newly constructed triangle and the existing triangle is no constraint. In some cases, the angle between the two points is so small that the sharp dihedral angle does not conform to the principle of local flatness when the surface is reconstructed. (2) In some cases, the newly constructed triangle will overlap with the existing mesh and lose the meaning of reconstruction.

In order to solve this kind of problem, this paper refers to the method in literature [25] in the selection of the optimal point, and adds the judgment of angle threshold $\beta$ of dihedral angle: When the dihedral angle between the original triangle and the newly constructed triangle is greater than $\beta$, this point can only be introduced.

In the process of mesh reconstruction, the expansion efficiency will be reduced by adding too many new edges. Therefore, when searching for the optimal point for the current active edge, this paper gives priority to the data points that already belong to the triangular mesh, also known as “used point”. To determine whether any point is a “used point” by looking up the inner edge adjacency list of the data point: if the inner edge adjacency list of a point is empty, it is indicated that the point is an exterior point; conversely, it is a “used point”. At the same time, in order not to overlap the newly generated mesh with the existing mesh, the three edges of the triangle composed of “used point” and the current active edge should not be inner edges.

In summary, the principle of “used point” is as follows: For any active edge, the points in the search area of that edge are divided into “used point set” and “residual point set”; The expansion points are first selected in “used point set”; and if none of them meet the conditions of the extension points, then “residual point set” are selected. As shown in Figure 3. The search area for active edge $l$ is $\{v_0, v_1, v_2, v_3, v_4\}$. At this point $\{v_0, v_2, v_4\}$ are “used point” and should be extended as a matter of priority.

4) METHOD STEPS

The steps of mesh reconstruction method based on geometric distribution of point neighborhood are as follows:

Step 1. According to 2.2, the point cloud is divided into flat point set, high curvature point set and boundary point set, and the attribute value $T$ of each point is 0, which indicates that all points are involved in mesh reconstruction.

Step 2. In the flat point set, select the seed triangle $S_0$, add the three edges of $S_0$ to the active edge queue, and set the $T$ value of the three vertices of the triangle to one.

Step 3. Take the head element of the active edge queue to determine whether the active edge is expandable. If it is expandable, go to Step 4; If not, delete the queue head element and judge the next element until the element is expandable or the queue with active edges terminates when empty.

Step 4. According to the optimal selection rule, search the expansion point of the current active edge. If not found, it indicates that the edge is a border edge, go to Step 6. Otherwise, go to Step 5.

Step 5. A new triangle $S$ and two “new” edges are formed by connecting the two endpoints of the current active edge with the selected expansion point. The $T$ value of the expansion point is set to 1, and two “new” edges are detected at the same time: If it is an existing edge, it is identified as an inner edge; Otherwise, an active edge is added to the active edge queue. Update the list of inner edge adjacency points for the two endpoints of this edge.

Step 6. In the flat point, if the active queue is not empty, delete the element of the queue head, take the new pair of head elements, and go to Step 3; Otherwise, enter the high curvature point set for mesh reconstruction.

Step 7. According to the above method, the high curvature point set and the boundary point set are traversed to complete the mesh reconstruction.

III. SIMPLIFICATION METHOD BASED ON HAUSDORFF DISTANCE

If all point cloud data is used for 3D mesh reconstruction for big data platform, it will not only occupy a large number of computer resources, reduce the operation efficiency, but also bring a lot of inconvenience to the subsequent storage, display, and transmission of huge data. Thus, it is very important to simplify the data effectively under the premise of ensuring accuracy.

Due to geometric features always being lost excessively in Kim’s simplification process of the scattered point cloud, we adopt an improved simplification method based on Hausdorff distance [29]. At first, the principal curvature of points in the point cloud is estimated by the least square parabolic fitting. Then an error metric based on Hausdorff distance of principal curvature is used to keep and extract the feature points. Finally, through testing and analyzing some measured data with different characteristics, the results show that the
presented method achieves the more reasonable effect of simplification.

A. HAUSDORFF DISTANCE

The Hausdorff distance describes the similarity between two sets of points, as a definition of the distance between two sets of points. If A and B are two finite sets of points that are not identical, hence the Hausdorff distance between A and B are defined as:

\[
D(A, B) = \max(d(A, B), d(B, A))
\]

(3)

where, \(d(A, B)\) and \(d(B, A)\) represent the one-way Hausdorff distance from set A to set B and set B to set A respectively, which are defined as:

\[
d(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|,
\]

\[
d(B, A) = \max_{b \in B} \min_{a \in A} \|b - a\|
\]

(4)

B. GEOMETRIC FEATURES OF POINTS

As the most basic element to describe geometric features, the feature points play an important role in the quality of the reconstructed surface. Therefore, before simplifying the point cloud, it is necessary to judge whether the data points are represented by geometric feature points. If the data point \(p\) between its \(k\) nearest neighbor points respectively, therefore after the establishment of the neighborhood relationship of points, for any data point \(p\), if the geometric features of the surface at point \(p\) are obvious, \(H_p\) is relatively large; or \(p\) is the non-feature point, \(H_p\) is smaller.

By determining the Hausdorff distance value of the data point, the curvature change of any point can be accurately reflected. Based on this, a considerable number of points are reserved to highlight the geometric features of the model in the region with large curvature variation. In the area with small curvature change, the redundant data points should be deleted as much as possible to effectively reduce the amount of data while ensuring that the surface will not be distorted. The simulation results show that the Hausdorff distance can be used as the standard to effectively reduce the amount of data while maintaining the geometric feature information in the object model.

C. METHOD STEPS

For the set \(\{q_i, i = 1, 2, \ldots, n\}\) of data points rasterized and topologically related, take any point \(p\), and the set of K neighborhood points is \(\{q_i, i = 1, 2, \ldots, k\}\). The method steps to calculate the Hausdorff distance of each sampling point and simplify the data points are as follows:

Step 1. The method of quadratic parabolic surface fitting is used to estimate the principal curvature of all points.

Step 2. Calculate the Hausdorff distance between point \(p\) and its domain point, and take the maximum value as the Hausdorff value of that point.

Step 3. Step 2. Calculate the Hausdorff distance between point \(p\) and its domain point, and take the maximum value as the Hausdorff value of that point.

Step 4. Traverse all points with Step 2 and calculate the Hausdorff values of all data point.

Step 5. Traverse all points and complete simplification.

IV. EXPERIMENT AND ANALYSIS

In order to verify the feasibility and correctness of our method, the following groups of point cloud data are tested. The hardware environment of the lab is: Intel (R) Xeon.
H.-A. Li et al.: R3MR: R3MR for Big Data Platform

FIGURE 4. Dragon model reconstruction.

(a) Original point cloud  (b) Reconstruction result in [25]  (c) Our reconstruction result

FIGURE 5. Rabbit model reconstruction.

(a) Original point cloud  (b) Reconstruction result in [25]  (c) Our reconstruction result

(R) 2.13GHz CPU, 4GB memory and the programming environment is OpenGL2.0. The unit of time is s. All the test data come from PLY files downloaded from the Stanford 3D scanning repository of the Stanford University website [30] and some of them are processed by our lab. In the real situation, such as autonomous driving, the quantity of data is much larger than our test 3D mesh, and many 3D meshes are operated at the same time. The quantity of the 3D meshes used in our experiments is enough to show the effectiveness of our proposed method and the experimental operation is also simple.

A. RECONSTRUCTION OF THE ORIGINAL POINT CLOUD

Experiment 1 is conducted to test the dragon model, which contains 41842 data points and has a high sampling density, as shown in Figure 4(a). Figure 4(b) is the result of triangular mesh reconstruction of literature [25], and Figure 4(c) is the reconstruction result of our method. Through comparison, it can be found that it uses the method of classification of data points, so the sharp features can be generated naturally in the tail, horn, claw and other boundary parts.

Experiment 2 is a comparison between our method and the mesh reconstruction method in literature [25], Figure 5(a) is the scattered point cloud of the rabbit model, due to the uneven distribution of data points in some parts of the model (e.g., the body part of the rabbit). There are many data points in the region of high curvature and sparse distribution in the flat region. If we use the method of literature [25], the effect of triangle mesh reconstruction is shown in Figure 5(b); The reconstruction results of our method are shown in Figure 5(c). Through comparison, it can be found that the reconstruction effect of our proposed method is better, and the detailed features of the physical model can be clearly displayed.

B. SIMPLIFICATION OF THE ORIGINAL POINT CLOUD

For experiment 3, the selection of the parameter in Equation (6) and (7) directly affects the time cost of the method and the result of data simplification. If $\varepsilon$ is larger, the time cost is relatively small, but it is difficult to achieve ideal results when data is simplified. On the contrary, if $\varepsilon$ is taken a smaller value, the simplification accuracy is improved, but the consumption of time is also relatively increased. In view of this, $\varepsilon$ are $10^{-2}$, $10^{-3}$, $10^{-4}$, $10^{-5}$ and $10^{-6}$ in this section, which are used for simulation experiments. The simplified dragon model (containing 41842 points data) is simplified using the method given above. And points are 3776, 4538, 5297, 6485 and 7192, respectively. The original point cloud and experimental results are shown in Figure 6, and the time consumption is 497 ms, 436 ms, 356 ms, 268 ms and 207 ms. Therefore, taking the simplification effect and time cost into consideration, the mean value of $\varepsilon$ is $10^{-4}$ in the comparison experiment.

Experiment 4. In order to verify the feasibility and correctness of our method, Kim’s method [31] and the method in literature [32] are used as a comparison to carry out the simulation experiment on the 3D mesh. The time unit is (ms). Figure 7 shows the complete surface scan data of the dragon
model, and the original point cloud (Figure 7(a)) contains 41842 points. The simplified result of Kim’s method contains 5303 points, as shown in Figure 7(b); After using the simplified method in the literature [32], 5469 points are included, as shown in Figure 7(c); Our method is simplified to include 5297 points, as shown in Figure 7(d), the remaining points after simplification are relatively close. From the perspective of simplification, our method contains more feature points in the tail, abdomen and neck of the dragon model, which better preserves the features of the model and makes the contour clearer.

Figure 8 shows the surface scan data of the rabbit. The original point cloud (Figure 8(a)) contains 35337 data points, the simplified result of Kim’s method contains 8829 data points, as shown in Figure 8(b). After simplification using the method of literature [32], it contains 9365 points, as shown in Figure 8(c). Simplified results of this paper in this paper include 8304 points, as shown in Figure 8(d). From the experimental results, it can be seen from the comparison of Figure 8(b)-(d) that the point cloud distribution of the effect of our method is relatively uniform with no cavitation. However, the detail line between Figure 8(b) and Figure 8(c) is not obvious. And Figure 8(b) appears cavitation. There’s a little bit of cavitation, but the details are not obvious at the same time. Therefore, under the same strategy, our method can well preserve the points in the feature region and delete many non-feature points, and the simplified results are relatively uniform.

Simulation experiments are compared between the methods in the literature [31], [32] and our method. Simplified results and running time are shown in Table 1 and Table 2. It is obvious that the method in this paper is 118 ms faster than that in literature [31] and 139 ms faster than that in literature [32] when reconstructing the dragon model. In the reconstruction of rabbit model, the method in this paper is 18 ms faster than that in literature [31] and 45 ms faster than that in literature [32]. By comparison, it can be found that the speed of point cloud simplification method based on Hausdorff distance is better than the other two methods, which improves the efficiency of the method on the premise of ensuring the quality of simplification.

C. RECONSTRUCTION AFTER SIMPLIFICATION

Experiment 5. The surface scan data of the rabbit model are reconstructed after simplification. The original cloud points are shown in Figure 9(a), the reconstruction result is shown in Figure 9(b) and the render effect is shown in Figure 9(c).
The triangular mesh reconstruction result after simplification is inferior to the original cloud points, but the features of the model are still retained. The data quantity is decreased, the reconstruction time is relatively reduced, but the reconstruction efficiency is improved accordingly.

Table 3 is a comparison of the time consumption of point cloud mesh reconstruction methods. As can be see from the table, the refactoring time increases with the increase of the number of data points. Compared with the method in literature [25], although it has a slight advantage in computing...
speed ostensibly, because of the classification of data, this method can reconstruct point cloud triangle mesh efficiently and accurately. When the vertex number is below ten thousand, our method also could work with a short time while the literature [25] cannot. Therefore, our method has some engineering significance.

### V. CONCLUSION

In this paper, we propose a novel region growing based 3D mesh reconstruction method for big data platform, which reconstructs the classified data points from simple to complex by the rational principle of optimal selection and the use of the inner edge adjacency list. Experimental results show that our method can accurately reconstruct the surface shape of the point cloud model and could reflect the detailed features of the model more naturally. Therefore, it can be applied to the surface reconstruction of scanning 3D point cloud in big data platform.

### REFERENCES

[1] K. Yu, S. Eum, T. Kurita, Q. Hua, T. Sato, H. Nakazato, T. Asami, and V. P. Kafle, “Information-centric networking: Research and standardization status,” IEEE Access, vol. 7, pp. 126164–126176, 2019, doi: 10.1109/ACCESS.2019.2938586.

[2] T. C. Zhang, J. Zhang, and J. P. Zhang, “Visualization analysis for 3D big data modeling,” in Proc. Int. Conf. Youm. Comput. Sci. Berlin, Germany: Springer, 2015, pp. 220–227, doi: 10.1007/978-3-662-46246-5_27.

[3] S. Otoum, B. Kantarci, and H. Moufah, “Empowering reinforcement learning on big sensed data for intrusion detection,” in Proc. IEEE Int. Conf. Commun. (ICC), May 2019, pp. 1–7, doi: 10.1109/ICC.2019.8761575.

[4] S. A. Melli, K. A. Wahid, P. Babyn, D. M. L. Cooper, and V. P. Gopi, “A multiscale constraints method learning on big sensed data for intrusion detection,” in Proc. Int. Conf. Youn. Comput. Sci., May 2019, pp. 1–7, doi: 10.1109/ACCESS.2019.2938586.

[5] H.-A. Li, Y. Zhang, Z. Li, and H. Li, “A multiscale constraints method learning on big sensed data for intrusion detection,” in Proc. Int. Conf. Youn. Comput. Sci., May 2019, pp. 1–7, doi: 10.1109/ACCESS.2019.2938586.

[6] H.-A. Li, Y. Zhang, Z. Li, and H. Li, “A multiscale constraints method learning on big sensed data for intrusion detection,” in Proc. Int. Conf. Youn. Comput. Sci., May 2019, pp. 1–7, doi: 10.1109/ACCESS.2019.2938586.

[7] B. D. Deebak, F. Al-Turjman, M. Aloqaily, and O. Alfandi, “IoT-BSFCAN: A smart context-aware system in IoT-cloud using mobile-fogging,” Future Gener. Comput. Syst., vol. 109, pp. 368–381, Aug. 2020, doi: 10.1016/j.future.2020.01.035 (050).

[8] R. G. Elston, “Selection, planning, and convenience in lithic procurement,” North Amer. Archaeologist, vol. 34, no. 4, pp. 433–450, Oct. 2013, doi: 10.2190/NA.34.4h.

[9] P. Martinez-Carricondo, F. Carvajal-Ramírez, L. Yero-Paneque, and F. Ágarcia-Vega, “Combination of nadiral and oblique UAV photogrammetry and HBIIM for the virtual reconstruction of cultural heritage. Case study of cortijo del fraile in Nijar, Almería (Spain),” Building Res. Inf., vol. 48, no. 2, pp. 140–159, Feb. 2020, doi: 10.1080/09613218.2019.1626213.

[10] B. D. Deebak, F. Al-Turjman, M. Aloqaily, and O. Alfandi, "IoT-BSFCAN: A smart context-aware system in IoT-cloud using mobile-fogging," Future Gener. Comput. Syst., vol. 109, pp. 368–381, Aug. 2020, doi: 10.1016/j.future.2020.01.035 (050).

[11] R. G. Elston, “Selection, planning, and convenience in lithic procurement,” North Amer. Archaeologist, vol. 34, no. 4, pp. 433–450, Oct. 2013, doi: 10.2190/NA.34.4h.

[12] P. Martinez-Carricondo, F. Carvajal-Ramírez, L. Yero-Paneque, and F. Ágarcia-Vega, “Combination of nadiral and oblique UAV photogrammetry and HBIIM for the virtual reconstruction of cultural heritage. Case study of cortijo del fraile in Nijar, Almería (Spain),” Building Res. Inf., vol. 48, no. 2, pp. 140–159, Feb. 2020, doi: 10.1080/09613218.2019.1626213.
[32] T. Liu, Z. Xu, and C. M. Sha, “Curvature estimation of scattered point cloud data based on bounding box method,” Sci. Tech. Eng., vol. 9, no. 12, pp. 3333–3336, 2009.

**HONG-AN LI** received the M.S. and Ph.D. degrees in computer science and technology from Northwest University, Shaanxi, China, in 2009 and 2014, respectively.

From 2014 to 2019, he was an Associate Professor with the College of Computer Science and Technology, Xi’an University of Science and Technology. His research interests include computer graphics and computer-aided geometric design, computer animation, and virtual reality.

**MIN ZHANG** received the bachelor’s degree in information and computing science from the College of Computer Science and Technology, Xi’an University of Science and Technology, in 2019. She is currently pursuing the master’s degree with the College of Computer Science and Technology, Xi’an University of Science and Technology.

Her research interests include the use of image processing and computer vision, and a small project, she completed was the research and implementation of image classification based on the GoogleNet model.

**KEPING YU** (Member, IEEE) received the M.E. and Ph.D. degrees from the Graduate School of Global Information and Telecommunication Studies, Waseda University, Tokyo, Japan, in 2012 and 2016, respectively.

He was a Research Associate and a Junior Researcher with the Global Information and Telecommunication Institute (GITI), Waseda University, from 2015 to 2019 and from 2019 to 2020, respectively, where he is currently a Researcher. He has hosted and participated in more than ten projects and involved in many standardization activities organized by ITU-T and ICNRG of IRTF and contributed to the ITU-T Standards Y.3071 and Supplement 35. His research interests include smart grids, information-centric networking, the Internet of Things, blockchain, and information security. He has been the General Co-Chair and the Publicity Co-Chair of the IEEE VTC2020-Spring EBTSRA workshop, the TPC Co-Chair of SCML2020, the Local Chair of MONAMI 2020, the Session Co-chair of CcS2020, and the Session Chair of ITU Kaleidoscope 2016. He has served as TPC Member of more than ten international conferences, including the ITU Kaleidoscope, the IEEE Vehicular Technology Conference (VTC), the IEEE Consumer Communications and Networking Conference (CCNC), and the IEEE Wireless Communications and Networking Conference (WCNC). He has been a Lead Guest Editor of *Sensors, Peer-to-Peer Networking and Applications, and Energies*. He is an Editorial Board Member of the IEEE *Journal of Vehicular Technology*.

**QIAOZHI HUA** received the B.E. degree in electrical communication from the Wuhan University of Science and Technology, in 2011, and the M.S. and Ph.D. degrees from Waseda University, in 2015 and 2019, respectively. He is currently a Lecturer with the Computer School, Hubei University of Arts and Science, Hubei, China. His research interests include game theory and wireless communication.

**YU ZHU** received the M.S. degree in computer science and technology from Northwest University, Shaanxi, China, in 2012.

His research interests include computer graphics and computer-aided geometric design, computer animation, virtual reality, and image processing.

---

**XIN QI** received the M.E. and Ph.D. degrees from Waseda University, Tokyo, Japan, in 2016 and 2019, respectively. He is currently a Research Associate with the Global Information and Telecommunication Institute, Waseda University. His research interests include ICN and 3N for next generation communication systems.