Simplification of point cloud data for large-scale ellipsoidal complex surface

Kai Huo¹, Hui Yang*, Hairong Fang¹, Shijian Shi¹ and Yufei Chen¹

¹ Department of Mechanical Engineering, Beijing Jiaotong University, Beijing, 100044, China
*Corresponding author’s e-mail: 15116342@bjtu.edu.cn

Abstract. Aiming at the issue that the current point cloud data simplification algorithm cannot accurately retain the subtle features of ellipsoidal complex surface, simplification algorithm of a point cloud data feature preservation based on principle component analysis (PCA) and adaptive mean shift (AMS) methods is proposed. First, K-means spatial clustering is used to make preliminary clustering and simplification of point cloud data. Second, the representative value of the subtle features of the point cloud data is calculated based on the PCA method. Then, according to the proposed improved adaptive mean shift method, the simplification of the point cloud data is developed. Finally, the point cloud data obtained by the denoising and smoothing of a large spherical crown workpiece collected at a certain equipment manufacturing site is used as the experimental object, and a comparative experiment is conducted between the proposed simplification algorithm and the k-means clustering algorithm, which proves the proposed algorithm can effectively maintain the subtle feature points of the ellipsoidal point cloud data.

1. Introduction

Due to the large size of large ellipsoidal components, the point cloud data obtained by the measurement system amounts to hundreds of thousands, which will greatly occupy the computer's memory space in subsequent data processing and affect the speed of computer operations. At the same time, it also causes great inconvenience to subsequent data storage, processing and numerical control processing operations. Therefore, it is necessary to reduce the collected point cloud data. According to the principle of the algorithm, the current common point cloud simplification algorithm can be divided into two types, namely, the division based on area and the feature division based on point cloud data points. The principle of the point cloud simplification algorithm based on area division is to replace each area of the point cloud data with one of the points on the premise of maintaining the features of the point cloud, such as equal spacing reduction method, vertex clustering method, chord height difference method, and so on. The principle of the simplification algorithm based on point cloud data point feature division is to describe each point in the point cloud with one or more evaluation indicators, and retain the points with higher evaluation indicators to ensure that the point cloud features are maintained and the simplification is completed.

Lin et al.[1] proposed a point cloud simplification algorithm based on coordinate incremental information, which used the coordinate incremental information of two-dimensional data points as the point evaluation index to simplify the point cloud. However, the algorithm requires the point cloud data must be discrete point cloud data obtained by scanning. Yang et al.[2] comprehensively considered the principles of region division and point feature division based on point cloud data, and
proposed a point cloud simplification algorithm based on the normal vector angle and curvature of each point of the point cloud. The algorithm used the curvature value of the point cloud to divide the area of the point cloud. In each point cloud area, the angle of the normal vector of the point is used as the basis for simplification. This algorithm can completely retain the feature of different areas of the surface, but the point cloud after the simplification is not uniformly distributed, and some information will be lost. Alexa et al.[3] proposed a simplification algorithm based on the least squares method to fit a plane, which used the distance from each point to the least squares plane fitted to its domain point as an evaluation index to simplify the point cloud. This algorithm can effectively retain the feature points in the non-flat area, but the point cloud data in the flat area is not enough. Pauly et al.[4] divided the point cloud data into several regions by hierarchical clustering method, and used the region growing method to reduce the points in each region. This algorithm can well retain the largest feature value in each region, and the number of points can be controlled. However, the algorithm relies heavily on the division of the initial area. Shi et al.[5] proposed a non-parameterized point cloud simplification algorithm based on K-means clustering, which used the maximum normal vector deviation as the evaluation index for each point. Under this evaluation index, the point cloud data was divided into clusters and the cluster center is taken as the point to be simplified and retained. This algorithm can well retain the detailed feature information at the edge of the point cloud, but the detailed feature points in the point cloud area are lost. Lee et al.[6] proposed a point cloud simplification algorithm based on the discrete morphology algorithm, which used the discrete morphology algorithm to calculate the discrete shape operator of each point in the point cloud data as an evaluation index for simplification. This algorithm can effectively retain the sharp features of the point cloud model, and the efficiency of simplification is high. However, it requires that the point cloud before the simplification should not have noisy points.

The point cloud simplification of large-scale ellipsoidal complex surface is focused on how to filter out the feature points with strong ability to express the information of the surface area and delete some unimportant points at the same time. In an area with simple surface information, one point or a few points can be used to represent the points in the entire area. However, in areas with complex surface information, it is required to retain those feature points with strong ability to express the information of the surface area. Otherwise, the reconstructed surface may be inconsistent with the surface of the workpiece in the subsequent surface reconstruction. Since the research object of this paper is a large-scale ellipsoidal complex surface, the feature of a point cannot be directly judged by information such as vectors or curvature. Therefore, this paper proposes a point cloud feature preserving simplification algorithm based on Principal Component Analysis and Adaptive Mean Shift method, which not only reduce the point cloud data of ellipsoidal surfaces with more uniform point cloud distribution but also can maintain the subtle feature of the original point cloud data.

The content of the paper is arranged as follows: the self-adaptive point cloud data feature preservation simplification algorithm is introduced in section 2, which includes preliminary simplification of K-means algorithm, the analysis of point cloud data based on the PCA, adaptive mean shift algorithm. In section 3, the experimental is carried out verify the effectiveness of the proposed algorithm. Finally, the conclusion is given in section 4.

2. Self-adaptive point cloud data feature preservation simplification algorithm

In this section, K-means spatial clustering is used to make a preliminary simplification of point cloud data firstly. Then, the representative values of the subtle features of the point cloud data are calculated based PCA method. Finally, according to an improved adaptive mean shift method, the point cloud data is reduced.

2.1. Preliminary simplification of K-means algorithm

K-means algorithm first appeared as a mathematical method of processing vector information in signal processing. With the development of science and technology, the K-means algorithm, as a classic and mature clustering analysis method, has been widely used in image segmentation, medical image
segmentation, remote sensing image analysis and other fields for in-depth data mining analysis[7]. The basic idea of the K-means algorithm is to divide the point cloud into several subsets according to a certain rule, and calculate a cluster center in these subsets, so that the points of the cluster center are evenly distributed in space. In this section, the basic idea of the K-means algorithm will be explained in detail.

For a sample point set \( \{x_1, x_2, ..., x_n\} \), the K-means algorithm will randomly generate K data points as the initial cluster centers at the beginning, denoted as \( \{\mu_1, \mu_2, ..., \mu_k\} \). Then, the K-means algorithm calculates the Euclidean distance from each point in the point set to each cluster center, and divides each point to the nearest cluster center to form a cluster subset, denoted as \( \{C_1, C_2, ..., C_k\} \). Next, the K-means algorithm will recalculate the new cluster center according to the following formula, and iteratively update the position of the cluster center until the error square of the cluster subset and the SSE meet the accuracy requirements or no longer change. Among them, the calculation formulas of the Euclidean distance \( d \), the cluster center point \( \mu_i \), and sum of squared errors of clustered subsets (SSE) are represented as follows:

\[
d(x_i, \mu_i) = \sqrt{\sum_{j=1}^{n} (x_{ij} - \mu_{ij})^2}
\]

(1)

\[
\mu_i = \frac{1}{N_i} \sum_{x_i \in C_i} x_i
\]

(2)

\[
SSE = \sum_{i=1}^{k} \sum_{j=1}^{N_i} |x_{ij} - \mu_{ij}|
\]

(3)

The clustering results of K-means algorithm are extremely dependent on the clustering centers randomly selected in the initialization phase, so the method of selecting cluster centers in the initialization phase is the top priority of the K-means algorithm. The simplest initial clustering center is to randomly select K data points in the point cloud data, but the initial clustering center obtained by this method may lead to uneven distribution of the final cluster subset. In order to solve the above problems, this paper constructs a K-D tree for the workpiece point cloud data, and takes some nodes of the K-D tree as the initial cluster center, so that the density of each cluster obtained is large and the cluster centers are separated from each other. The K-D tree (K-Dimensional tree) is actually a binary tree, and the tree stores some multi-dimensional data. In this paper, it is three-dimensional data. Constructing a K-D tree for a set of 3D data points is actually the division of 3D data points. Each non-leaf node in the K-D tree corresponds to a rectangular 3D point cloud area. The K-D tree segmentation of the point cloud can ensure that the amount of point cloud data contained in each subset is roughly the same[8].

In summary, the preliminary simplified process of the K-means algorithm in this paper is as follows:

1. Perform K-D tree division on the original point cloud data \( P = \{p_1, p_2, ..., p_n\} \), where \( n \) is the total number of point clouds, and select K nodes as the initial clustering centers \( N = \{\mu_1, \mu_2, ..., \mu_k\} \).

2. According to the Eq. (1), calculate the Euclidean distance between all points in cloud data and each cluster center, and select the closest cluster according to the minimum distance, so as to obtain K clusters \( C = \{C_1, C_2, ..., C_k\} \).

3. Based on the Eq. (2), calculate the cluster center point \( \mu_i \) of each cluster point cloud separately, and use it as the new cluster center.

4. Perform the step (2) on all point cloud data according to the new clustering center, and perform clustering again.

5. Calculate the value of SSE according to the Eq. (3), and repeat steps (3) and (4) until SSE meets the accuracy requirement or no longer changes. The K cluster centers obtained at this time are the initial cluster division results.
2.2. Analysis of point cloud data based on the PCA method

After the K-means clustering algorithm is used to initially simplify the point cloud data, it is necessary to calculate the characteristic value of each point of the point cloud data and then perform clustering again. At present, the feature representation methods of point cloud data can be divided into three types based on the principle, which include the calculation of feature value based on the PCA method, the surface fitting with the distance from the fitting plane to the point as the characteristics of the points and the statistical value of the field points of each point in the point cloud data based on statistical methods[9]. In 2001, Gumhold et al.[10] analyzed point cloud data for the first time based on the PCA method, and used the eigenvalues and eigenvectors of the matrix to identify the feature points in the point cloud. On top of this, Pauly et al.[11] based on the PCA method to analyze the eigenvalues and eigenvectors obtained from the point cloud data, and proposed a measurement index that characterizes the curvature of the surface near the point—the surface change $\omega$. The PCA analysis method first subtracts the coordinates of the center of gravity of the point cloud from the points in the point cloud, and constructs a covariance matrix $C_{33}$. The ratio of the smallest eigenvalue of the covariance matrix to the sum of all eigenvalues is defined as the surface change degree $\omega$. The basic process of PCA method to calculate point cloud eigenvalues is as follows:

Suppose the point cloud data set to be reduced is $P = \{ p_1, p_2, ..., p_n \}$, where $n$ is the total number of point clouds. If the nearest $k$ area of one point of $p_i$ is expressed as $Nb(p_i)$, then the calculation formula of the center of gravity $u$ of the area $Nb(p_i)$ can be obtained as follows:

$$ u = \frac{1}{k_i} \sum_{p_j \in Nb(p_i)} p_j $$

(4)

Then the covariance matrix $C$ of point $p_i$ can be defined as:

$$ C = \frac{1}{k_i} \sum_{i=1}^{k} (p_i - u)(p_i - u)^\top = \frac{1}{k_i} \begin{bmatrix} p_i - u \\ p_i - u \\ \vdots \\ p_i - u \end{bmatrix} \begin{bmatrix} p_i - u \\ p_i - u \\ \vdots \\ p_i - u \end{bmatrix}^\top, \quad p_i \in P. $$

(5)

The covariance matrix $C$ of the point $p_i$ reflects the relevant information of the local surface near the point $p_i$. Assume that the three eigenvectors of the covariance matrix $C$ are represented by $v_0, v_1$, and $v_2$, and the eigenvalues corresponding to the three eigenvectors are $\lambda_0, \lambda_1$, and $\lambda_2$. Pauly et al. found that the smallest eigenvalue $\lambda_2$ corresponding to the eigenvector $v_2$ is the normal vector of the domain surface of the point $p_i$, and $\lambda_2$ indicates the changes of the point $p_i$ along the normal vector of the domain surface. Then, the change of the minimum eigenvalue of $p_i$ can be used to estimate the curvature change degree in the area of this point. Therefore, the degree of surface change at point $p_i$ is represented as:

$$ \omega_k(p_i) = \frac{\lambda_2}{\lambda_0 + \lambda_1 + \lambda_2} $$

(6)

Since the covariance matrix $C$ is a symmetric matrix, $\lambda_0, \lambda_1, \lambda_2$ are all positive values and the value range of $\omega_k(p_i)$ is $[0, 1/3]$. The size of $\omega_k(p_i)$ quantitatively reflects the extent to which the area surface of the point leaves the tangent plane in the direction of the normal vector of the point, that is the unevenness of the area surface of the point. When the domain surface of the point is a plane, the value of $\omega_k(p_i)$ is 0. When the domain surface of the point is a standard spherical surface, $\omega_k(p_i)$ is 1/3. The larger the value of $\omega_k(p_i)$, the sharper the curved surface near the point $p_i$. 
However, the surface change degree $\omega_k$ calculated by the PCA method can only identify the feature points of a flat area. Since the curvature of each point on the surface is not much different, $\omega_k$ cannot be used to identify the feature points of the point cloud data on the large-scale ellipsoidal complex surface. In order to more accurately reflect the features of different points on the surface, Nie et al.[12] proposed that the subtle feature points on the surface can be characterized according to the surface variation based local outlier factor (SVLOF). The SVLOF value can more accurately reflect the features of each point on a large ellipsoidal complex surface, and then perform clustering based on the degree of feature. Therefore, this paper uses the SVLOF parameter values of discrete points to characterize the features of data points on a large ellipsoidal complex surface. Here, SVLOF is defined as follows:

$$SVLOF_k(p) = \frac{\omega_{k+1}(p_i)}{\omega_k(p_i)}$$  \hspace{1cm} (7)

where $\omega_{k+1}(p_i)$ is the degree of surface change calculated from the $k+1$ neighborhood of the $p_i$ point, and $\omega_k(p_i)$ is the degree of surface change calculated from the $k$ neighborhood point of the $p_i$ point. As shown in figure 1, the value of $\omega_{k+1}(p_i)$ and $\omega_k(p_i)$ will not be very different when a point on the surface is a non-feature point, and the value of SVLOF is close to 1 at this time. However, the value of $\omega_{k+1}(p_i)$ and $\omega_k(p_i)$ will be very different when a point on the surface is a feature point, and the SVLOF value at this point will also be larger.

![Figure 1. Difference in surface variability between feature points and non-features point surfaces on a surface.](image)

2.3. Adaptive mean shift algorithm

After K-means clustering is performed on the point cloud data of large-scale ellipsoidal components, the point cloud data is evenly divided into several clusters, and the center of each cluster is the preliminary simplified result. However, this simplified method does not retain the feature points in the point cloud data, and will cause larger errors in the subsequent surface reconstruction. Therefore, for the initially obtained clusters and their cluster centers, this paper uses an improved mean shift algorithm to calculate the drift of the cluster centers according to the SVLOF feature expression value of each point in the cluster to obtain a new cluster center and ensure that the feature points in the cluster are retained.
The mean shift algorithm is a typical parameter-free adaptive clustering algorithm. It continuously searches the area with the highest density in the sampling point field through iterative calculations. Then, through vector superposition, the sampling points continue to converge in the direction of increasing density to the vicinity of the point with the highest density in the point cloud, so as to realize automatic clustering of point cloud data[13]. The basic idea of the mean shift algorithm is as follows.

Assuming that the point cloud set of sample points to be processed is any \( n \) sample points in the \( d \)-dimensional Euclidean space \( \mathbb{R}^d \), the mean shift vector of any point \( x \) is given as follows:

\[
M_h(x) = \frac{1}{k_x} \sum_{x \in S_k} (x - x) \tag{8}
\]

where \( x \) represents the sampling point, \( x \) represents the center of the sampling point field, and \( k \) refers to the number of points that fall into the sampling point field among the \( n \) sample points. \( S_k \) is the area of sampling points, the simplest way is to take it as a spherical range with radius \( h \). The point in \( S_k \) is the set of \( y \) points that satisfy the following relationship:

\[
S_k(x) = \left\{ y : (y - x)^2 \leq h^2 \right\} \tag{9}
\]

In 1995, Cheng[14] first proposed the concept of kernel function. Because the distance between different sample points and sampling points has very different effects on the mean shift vector \( M_h(x) \), the points closer to the sampling point should have greater weight. Therefore, the importance of each sample point to \( M_h(x) \) in the iterative calculation is not the same, and a weight coefficient \( w(x) \) needs to be introduced to reflect the distance between the point and the sampling point. The mean drift vector can be extended as follows:

\[
M_h(x) = \frac{\sum_{i=1}^{n} \frac{G(x - x)}{h} w(x) x_i}{\sum_{i=1}^{n} \frac{G(x - x)}{h} w(x)} - x \tag{10}
\]

where \( G(x) \) represents the kernel function, which is a convenient method for calculating the inner product of low-dimensional space data mapped to high-dimensional space.

For the workpiece point cloud data to be simplified in this paper, a certain clustering point set obtained after preliminary clustering and simplification is expressed as \( Q = \{ q_1, q_2, ..., q_m \} \), the clustering center of point cloud \( Q \) is \( q_m \), \( SVLOF(q_i) \) is the local outlier coefficient of the surface change of \( q_i \) in the point set \( Q \). Then, the mean shift vector at point \( q_m \) is defined as:

\[
M_{g,h}(q_m) = \frac{\sum_{i=1}^{m} \frac{G(q_i - q_m)}{h} w(q_i) q_i}{\sum_{i=1}^{m} \frac{G(q_i - q_m)}{h} w(q_i)} - q_m \tag{11}
\]

where \( G(x) \) is the kernel function, \( h \) is the bandwidth. In this paper, an adaptive kernel window \( h \) is used, so that each point \( q_i \) corresponds to a \( h_i \). The research of Mayer et al.[15] showed that the bandwidth \( h_i \) is taken as \( |q_i - q_j| \) is a simple and effective method, which can realize the automatic change of the bandwidth without parameters. Here, \( q_j \) is the neighboring points of \( q_i \), and \( |q_i - q_j| \) represents the Euclidean distance from point \( q_i \) to other points in its surrounding area. Therefore, the weight coefficient \( w(q_i) \) in Eq. (11) is defined as follows:

\[
w(q_i) = \begin{cases} 
1 & \text{if } |SVLOF_{k}(q_i) - SVLOF_{kave}(q)| > 3T_v \\
0 & \text{if } |SVLOF_{k}(q_i) - SVLOF_{kave}(q)| < 3T_v 
\end{cases}
\tag{12}
\]
Here, $SVLOF_{kave}(q)$ and $T_i$ are defined as follows:

$$SVLOF_{kave}(q) = \frac{1}{m} \sum_{i=1}^{m} SVLOF_i(q_i)$$  \hspace{1cm} (13)

$$T_i = \sqrt{\frac{1}{m-1} \sum_{i=1}^{m} (SVLOF_i(q_i) - SVLOF_{kave}(q))^2}$$ \hspace{1cm} (14)

In summary, the point cloud feature preservation simplification algorithm based on PCA analysis method and adaptive mean shift method proposed in this paper is as follows:

Start

Input the point cloud data after denoising and smoothing

Perform K-means clustering on the input point cloud data to get the initial cluster center

Calculate the SVLOF value of each point through the PCA algorithm

Perform adaptive drift iterative calculation in the initially obtained clusters to get the final cluster centers

End

Figure 2. Flowchart of the point cloud simplification algorithm in this paper.

3. Experiment of the proposed algorithm

In this section, the point cloud data obtained after the denoising and smoothing process in the previous research[16] is the experimental object to verify the effect of the self-adaptive point cloud feature preservation simplification algorithm. The point cloud data processed by the denoising smoothing algorithm is shown in figure 3, and the point cloud data processed by the adaptive point cloud feature preservation and simplification algorithm in this paper is shown in figure 4. After the denoising and smoothing process in reference[16], the amount of point cloud data on the surface of the large-scale ellipsoidal components is 68057, and the amount of point cloud data after simplification is 708. It can be seen that the algorithm proposed in this paper can effectively simplify the point cloud data of the large-scale ellipsoidal complex surfaces.

In order to verify the accuracy of the simplified algorithm in this paper, combined with the subsequent CNC machining requirements of the large-scale ellipsoidal components, this paper uses the color difference diagrams of the curved surface before and after the reduction generated by the Polyworks software to judge the effect of the point cloud simplification algorithm. Figure 5 shows the color difference between the original point cloud fitting surface and the point cloud fitting surface obtained by the K-means clustering and simplification algorithm[5]. Figure 6 shows the color difference diagram of the original point cloud fitting surface and the point cloud fitting surface obtained by the simplification of the self-adaptive point cloud data feature preservation and
simplification algorithm in this paper. It can be seen from the chromatic aberration diagrams of the fitted surfaces after the two algorithms are simplified that the adaptive point cloud data feature preservation and simplification algorithm proposed in this paper can better maintain the subtle features of the ellipsoidal complex surface, and the reconstructed surface before and after streamlining will not have too big difference, which has no effect on the subsequent CNC machining.

4. Conclusions
Aiming at the problem that the current point cloud simplification algorithm cannot accurately retain the subtle features of the ellipsoidal surface, this paper proposes an adaptive point cloud feature retention simplification algorithm based on the principal component analysis method and the adaptive mean shift method. The K-means spatial clustering method is used to perform preliminary clustering and simplification of the point cloud. Then, the representative value of the point cloud data's subtle features is calculated based on the PCA method. According to the improved adaptive mean shift method, the point cloud data is simplified. Finally, the point cloud data of a large spherical crown workpiece collected on an equipment manufacturing site is used as the experimental object, which proves that the simplified point cloud data can better retain the subtle features of the original surface. It also ensures that the error between the reconstructed surface and the real surface of the workpiece is smaller.
Acknowledgments
The authors gratefully acknowledge the financial support of the Fundamental Research Funds for the Central Universities under Grant No.2018JBZ007, and Huo Kai Scientific Pre-research Fund under Grant No. KYYLM338.

References
[1] Lin, X.H. (2017) Research on Laser Point Cloud Data Compression and Surface Modeling. Xi’an University of Science and Technology, Xi’an.
[2] Yang, L.J. (2014) Research on Compression Algorithm of Point Cloud Data-Take Digital Geological Museum as an Example. Central South University, Changsha.
[3] Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T. (2003) Computing and rendering point set surfaces. IEEE Transactions on Visualization and Computer Graphics, 9(1): 3-15.
[4] Pauly, M., Gross, M., Kobbelt, L. (2002) Efficient Simplification of Point-Sampled Surfaces.
[5] Shi, B.Q., Liang, J., Liu, Q. (2011) Adaptive simplification of point cloud using k-means clustering. Computer-Aided Design, 43(8): 910-922.
[6] Lee, P.F., Huang, C.P. (2011) The DSO Feature Based Point Cloud Simplification. In: 2011 Eighth International Conference on Computer Graphics, Imaging and Visualization. Singapore. pp. 1-6.
[7] Deng, B.Y. (2020) A survey on advanced k-means algorithm. Computer Engineering and Software, 41(02): 188-192.
[8] Tian, Z.H. (2019) Research on 3D Point Cloud Data Simplification and Mesh Reconstruction Algorithm. Yanshan University, Qinhuangdao.
[9] Nie, J.H., Liu, Y., Gao, H., Wang, B.Y., Ge, Y.Q. (2015) Feature line detection from point cloud based on signed surface variation and region segmentation. Journal of Computer-Aided Design and Computer Graphics, 27(12): 2332-2339.
[10] Gumhold, S., Wang, X.L., Macleod R. (2001) Feature Extraction from Point Clouds. In: 10th International Meshing Roundtable. California. pp. 293-305.
[11] Pauly, M., Keiser, R., Gross, M. (2003) Multi-scale feature extraction on point-sampled surfaces. Computer Graphic Forum, 22(3): 281-289.
[12] Nie, J.H., Hu, Y., Ma, Z. (2011) Outlier detection of scattered point cloud by classification. Journal of Computer-Aided Design and Computer Graphics, 23(09): 1526-1532.
[13] Wang, H.Y. (2014) Research on Image Target Segmentation Methods Based on Clustering. Xidian University, Xi’an.
[14] Cheng, Y.Z. (1995) Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence. 17: 790-799.
[15] Mayer, A., Greenspan, H. (2009) An adaptive mean-shift framework for MRI brain segmentation. IEEE Transactions on Medical Imaging, 28: 1238-1250.
[16] Fang, H.R., Shi, S.J., Yang, Y. (2020) Denoising and smoothing algorithm of point cloud data for profiling of large ellipsoid complex curved surface. Modern Manufacturing Engineering, 76-80.