# A study and implementation on the data reduction based on the curvature of point clouds

**Hong Jiang[1,3], Wenlei Sun[1*], Yongfang Shi[2], Yanhua Huang[1]**

[1] *1The School of Mechanical Engineering, Xinjiang University, Urumqi, China, 830047*

[2]*The School of Medical Engineering, Xinjiang Medical University, Urumqi, China, 830011*

[3] *The School of Software, Xinjiang University, Urumqi, China, 830008*

*Received 6 February 2014, www.tsi.lv*

**Abstract**

In the process of Reverse Engineering (RE), higher density of measured data from all kinds of parts with complex curved surface will not only lead to lower efficiency in computing, storing and data processing, but also affect the fairness of reconstructed surface. According to the advantages and disadvantages of common algorithms, an algorithm for data reduction is proposed in this paper, in which the neighbourhood search method based on the point cloud's curvature is used. With the utilization of proposed algorithm, high precision and the desired effect can be ensured. Finally, a roller bit's data cloud, as an example, is reduced efficiently and validly by the algorithm in this paper.

*Keywords:* Reverse Engineering, Data Reduction, Point Cloud, Curvature

## 1 Introduction

As an advanced manufacturing method, RE employs the digital technology for measurement and Three-Dimensional (3D) model reconstruction. Based on the 3D reconstruction model, the products with defects and deficiencies can be redesigned and remanufactured according to such analyses as mechanics, dynamics and so on. In the RE process, a crucial work is to obtain and process the data of objects surface.

The digital data obtained through the 3D laser or camera scanner are regarded as the point cloud. When the measurement is required by a complex curved surface, especially for some quadratic surface and cubic surface, lots of sufficient data are needed to be measured. However, due to too much sampled points on the reconstructing surface, the computer will be made too slow to implement the storage and calculation and the surface will be made less smooth. So it is necessary to reduce the point cloud data according to the characteristics and requirements of reconstruction parts [1-4, 14].

## 2 The common algorithm of data reduction

Chen *et al* [5] put forward a method through which the measured data can reduce data point through reducing triangular mesh generated directly by itself in 1999. Lee *et al* presented a data reduction method for laser scanning and measuring, including the minimum distance, angle deviation method and uniform grid method. The Preceding methods have the common shortcoming, that is, difficult to determine the boundary point cloud data. Shang et al. [6] put forward the adaptive minimum distance method based on the above methods. By means of this method and based on the principle that the smaller minimum distance is chosen in mutation and transitional regions and the larger minimum distance is chosen in flat regions, the minimum distance is firstly chosen for data points in each zone according to the accuracy and curvature change. Thereafter, the method simplifies the data for different regions separately, which is better to keep the detailed feature of original data. However, it can be found that the effect of data processing using this method is more preferable for the smaller amount of point cloud data than for the larger one, and the efficiency and accuracy are unable to meet the design requirements. Shi in Xi'an Jiaotong University [7] presented a method of data reduction based on the remaining features, which can not only simplify point cloud data effectively but remain the features of original data. This method chooses one representative point as the original class and then classifies other data into the original class. Afterwards it traverses each class as well as replaces the class using the obtained local modal points so as to accomplish data reduction. The method is able to remain the geometrical shape of the original model for the surfaces better with larger curvature change and more additional features.

The common algorithms of data reduction include bounding box, random sampling and uniform grid method.

Bounding box algorithm starts at one point of cloud data. A certain size of rectangular bounding box is firstly established and then divided into some small cubes of

[*] ***Corresponding author*** e-mail: onlyxjjh@126.com

uniform size by given ruler. In each cube, the centre point of this cube is selected to replace all points in this cube. As a result, all points of this cloud point are divided and replaced.

However, the edge length of the bounding box in this method is given by user arbitrarily and all cubes are of the same size, therefore, it is impossible to ensure the reduction accuracy. As for the density data, a centre point replacing all the points in the cube will lose more important and key points with detailed characters. Therefore this method will fail when the complicated surface curvature change obviously.

Random sampling algorithm is relatively simple. It is easy to be realized as following. A random function that can represent all cloud data points needs to be found first. Meanwhile this function can generate random numbers at the range of point cloud data. Then, the number of points that the point cloud will be left is assumed. Certain random numbers produced by the random function which represent some points are deleted next. Repeat this step until the number of the left points is reduced to a given value. This algorithm is simple and easy to be implemented. The efficiency of the algorithm is higher. However, the shortcoming of this algorithm is more obvious. When massive amount of data are reduced, the reduction results are close to a uniform simplification because of its high randomness. The precision of reduction results is uncontrollable and the uniformity of the reduced data will be reflected. In the meantime, it is difficult to reconstruct a 3D mode due to the possibility of presented hole.

For uniform grid algorithm, it assigns all data points into the corresponding grids based on "median filter" principle. Then a median point to replace all points in this grid was selected. This is an improved bounding box algorithm in fact.

The uniform grid method can overcome some shortcomings of the spline curve, the same grid size and the grid that is divided too small, easily result in producing some empty grids, and a lack of the flexibility for capturing the shape will lead to a waste of time and space.

Considering the large quantity of dense points and complex surfaces, the above traditional algorithms are not good at processing the point cloud data since the features of some key positions could be lost. The subsequent surface modelling could also be affected and the precision of model could not be guaranteed. The main reason is that different degrees of data reduction cannot be determined according to the different surface features.

### 2.1 CURVATURE-BASED ALGORITHM OF DATA REDUCTION

Because of the insufficiencies of traditional algorithms, the algorithm of curvature has been drawn a lot of attention by researchers and a number of research achievements have been made including neighbourhood establishment, curvature estimation and principles for reducing data [8, 9]. There are still deficiencies in these research achievements in some degree. For example, in the principles of data reduction, some mistakes appear constantly by using the methods of minimum distance and angular deviation. Therefore, this paper proposes a curvature algorithm based on paraboloid fitting, which is suitable neighbourhood search of even or uneven point cloud. In the described algorithm, the mean value of the curvature is regarded as a criterion for data reducing and data subdivision in which the data are uneven in high curvature region. The algorithm is good at dealing with the point cloud with complex features and high curvature.

### 2.2 CURVED SURFACE AND CURVATURE

Curvature is a basis for measuring the uneven degree of geometry and reflects important features of the surface. The normal curvature at the main direction of one point on the surface is regarded as the principal curvature of the point.

Suppose a surface $S$:$r$= $r(u,v)$, make $v$=$v_0$ and fix it, let $u$ change, then a curve, named $u$-curve, will be drawn by $r$=$r(u,v_0)$; Likewise, make $u$=$u_0$ and fix it, let $v$ change, then a curve, named $v$-curve, will be drawn by $r$=$r(u_0,v)$. Curve $u$ and Curve $u$ form a coordinate network of curved lines. The mesh consisting of the line of curvature is a coordinate network of curved lines. Setting the principal curvatures along the "line" as $k_1$, the principal curvatures along the "line $v$" as $k_2$ and the intersection angle between the random direction of surface, $d$=$du$:$dv$, and the curve as $\theta$, then $k_n$, the normal curvature along ($d$), satisfies the Euler's formula: $k_n = k_1\cos^2\theta + k_2\sin^2\theta$. Setting ($d$)= $du$:$dv$ as the main direction of curved surface $s:\vec{r}=\vec{r}(u,v)$ at point P, the principal curvatures along the main direction as $k_N$, then the computational formula of $k_N$ is:

$$\begin{vmatrix} L-k_N E & M-k_N F \\ M-k_N F & N-k_N G \end{vmatrix} = 0 . \quad (1)$$

That is:

$$(EG-F^2)k_N^2 - (LG-2MF+NE)k_N + (LN-M^2) = 0 , \quad (2)$$

where $E$, $F$, $G$ are first type elements of surface and $L$, $M$, $N$ are second type elements of surface.

Suppose $k_1$, $k_2$are two principal curvatures of a point on the surface, the $k_1k_2$ is called Gaussian curvature of the point and recorded as $K$, that is $k$= $k_1k_2$; the average of the curvatures is called mean curvature of this point on the surface and recorded as $H$, that is:

$$H = (k_1 + k_2)/2 , \quad (3)$$

According to the formula of principal curvature and Wada's theorem, the formulas of Gaussian curvature are shown as:

$$K = \frac{LN - M^2}{EG - F^2} \text{ (Gaussian curvature)}, \quad (4)$$

and

$$H = \frac{LG - 2MF + NE}{2(EG - F^2)} \text{ (Mean curvature)}. \quad (5)$$

## 2.3 NEIGHBOURHOOD SEARCH

There are several method in solving neighbourhood, including line-by-line searching method, neighbourhood ball searching method [11], 3D grid searching method [12] and the octree searching method [13] etc. Among them, line-by-line searching method is only suitable for the point cloud with obvious scanning line characteristics, such as point cloud by laser scanning.

For neighbourhood ball searching method, a good deal of time and internal storage are occupied during searching neighbourhood ball of every point according to the radiuses. In addition, this method has the following problems while dealing with non-uniform scattering point cloud. Since some neighbourhoods probably have no point or fewer than three points in small density point area, the subsequent curvature estimation could not be accomplished. However, in large density point area, some neighbourhoods may have too many points, which will slow the computational speed.

For 3D grid searching method, firstly, the measured data must be read-in one-dimensional array. Then the maximum value and minimum value of the data on the coordinate axis of *X*, *Y*, *Z* could be found. The minimum space bounding box and the three edges length are constructed as following:

$$\begin{cases} x_{box} = x_{MAX} - x_{MIN} \\ y_{box} = y_{MAX} - y_{MIN} \\ z_{box} = z_{MAX} - z_{MIN} \end{cases}, \quad (6)$$

where $x_{max}$ is max and $x_{min}$ is min in X direction; $y_{max}$ is max and $y_{min}$ is min in Y direction; $z_{max}$ is max and $z_{min}$ is min in *Z* direction. $x_{box}$, $y_{box}$ and $z_{box}$ are edges length in *X*, *Y*, *Z* direction respectively.

In order to ensure that all data points are included in the minimum space bounding box, the (6) is revised as:

$$\begin{cases} x_{box} = (x_{MAX} - x_{MIN}) \times 1.1 \\ y_{box} = (y_{MAX} - y_{MIN}) \times 1.1. \\ z_{box} = (z_{MAX} - z_{MIN}) \times 1.1 \end{cases} \quad (7)$$

Then the minimum bounding box is divided into $M \times N \times L$ small cube grids and the length of edge is "*cube_size"*. The numbers of points included in each cube grid are the same. The number of cubic grids in *X*, Y, and Z axis direction are:

$$M = \left[ \frac{x_{box}}{cube\_size} \right], \; N = \left[ \frac{y_{box}}{cube\_size} \right], \; L = \left[ \frac{z_{box}}{cube\_size} \right], \quad (8)$$

and its edge length "*cube_size"* is:

$$cub\_size = \sqrt{2 \times (x_{box} \times y_{box} + x_{box} \times z_{box} + y_{box} \times z_{box}) \times \frac{NUM}{N}}, \quad (9)$$

where *N* is the number of points of this cloud point, the *NUM* is the expected number of cubic grid contained points.

In practice, the measured points are non-uniformly distributed. Therefore, the number of cubic grid is not necessarily the expected number *NUM*. Hence for calculating the edge length of cubic grid, an adjusting coefficient $\beta$ is added. The edge length is adjusted dynamically:

$$cub\_size = \beta \sqrt{2 \times (x_{box} \times y_{box} + x_{box} \times z_{box} + y_{box} \times z_{box}) \times \frac{NUM}{N}}. \quad (10)$$

Next, a linked list array *List*[*i*][*j*][*k*] is established. According to the index numbers of the data points in the *X*, *Y*, *Z* axial directions, the points are inserted into this linked list. For example, the point $P_i=(x_i,y_i,z_i)$, and its index number are:

$$i=\left[\frac{x_i-x_{MIN}}{cub\_size}\right];\quad j=\left[\frac{y_i-y_{MIN}}{cub\_size}\right];\quad k=\left[\frac{z_i-z_{MIN}}{cub\_size}\right],\quad (11)$$

and the point $P_i$ is inserted into *List*[*i*][*j*][*k*].

After the list of cube grid is established, searching for the nearest neighbour points of each sample point would continue and the number of nearest neighbour points is *k*. While establishing a nearest neighbour points linked list for every sample point, the distance from the neighbour point to base point is calculated, the point into the nearest neighbour points list is added based on the distance value. The method for searching a nearest neighbour points is, firstly its index number is calculated. Secondly, its nearest neighbour point is searched in the cube grid including itself. But this cubic grid can't be considered alone since no matter how big this cube grid is, the neighbour points of sample point are likely to exist in the grid adjacent to this grid. For example, when a point is located on the faces of the cube, then its neighbour points may be found in the other cube which has the same face with this cube. Therefore, when searching the *k* neighbour points of a sample points, its twenty six adjacent cubes must be searched.

In the searching process, an adjacent ball is used to describe the current *k* nearest neighbour points. The adjacent ball is updated if the nearest neighbour is found. Thus, this adjacent ball is always the smallest one that contains the searched k neighbouring points. The process starts with the searching of the sample points in the cubic grids. Then it searches the points in the nearest neighbour grid. When a grid is identified, the cubic grid disjointing with the current adjacent ball will be removed from the grid list. In the meantime, the identified grid will also be excluded from the grid list after searching. Likewise, the searching will be continued until the grid list becomes empty. Such a method guarantees the minimum number of checked cubic grids and the improved computational efficiency.

Therefore, the path of an octree for seeking space point can be obtained by interpreting the position codes in the process of subspace decomposition and transforming decomposition range each time. Due to the direct relationship of position codes and the position of cubes, it can be specified as the position code of a child node on X axis plus 1 relative to the adjacent nodes on the left side, the one on Y axis plus 2 relatives to the adjacent nodes on the bottom, and the one on Z axis plus 4 relatives to the adjacent nodes on the rear side.

According to the encoding characteristics of the octree, each node of the tree is solely corresponding to a 8-hexadecimal number coded in binary form based on the coordinates of nodes in the divided space. These coordinates are calculated as follows:

$$\begin{cases} x=a_{n-1}2^{n-1}+a_{n-2}2^{n-2}+\cdots+a_k2^k+\cdots+a_12^1+a_02^0 \\ y=b_{n-1}2^{n-1}+b_{n-2}2^{n-2}+\cdots+b_k2^k+\cdots+b_12^1+b_02^0 \\ z=c_{n-1}2^{n-1}+c_{n-2}2^{n-2}+\cdots+c_k2^k+\cdots+c_12^1+c_02^0 \end{cases},\quad (12)$$

where $a_k,b_k,c_k\in\{0,1\}$, *and* $k\in\{0,1,...,n-1\}$.

If the number of a cube uniting the octree were known, the coordinates can be rewritten as:

$$\begin{cases} x=\sum_{i=1}^{n-1}(q_i\ \mathbf{mod}\ 2)\times 2^i \\ y=\sum_{i=1}^{n-1}((q_i/2)\mathbf{mod}\ 2)\times 2^i \\ z=\sum_{i=1}^{n-1}((q_i/4)\mathbf{mod}\ 2)\times 2^i \end{cases},\quad (13)$$

where n is the depth of the current node.

The following equation (14) can be obtained based on the numbers of cube units in the octree space:

$$q_i=c_i2^2+b_i2^1+a_i2^0.\quad (14)$$

Likewise, if one sub-cube unit number can be known, its coordinate can be calculated inversely using equation (13) and (14). Assume the relative coordinate of the node in the final bounding box is (*x*,*y*,*z*), 26 relative coordinates of the adjacent minimum bounding box can be represented by the following equation considering the division characteristics of a bounding box space:

$$\begin{cases} x'=x\pm\delta \\ y'=y\pm\delta \\ z'=z\pm\delta \end{cases},\quad (15)$$

where $\delta\in[0,1]$; *x'*, *y'*, *z'* and *x*, *y*, *z* cannot be equal simultaneously.

Moreover, for such isolated and sparse noisy points, Due to the far distance from other points, their bounding boxes for the isolated and sparse noisy points are in relatively isolated positions when the divided bounding box space or the bounding box contains fewer points. Given a threshold value *T*, any points outside the centre with a value less than *T* are identified. The points contained in the bounding box are noisy points and thus

can be removed to avoid their impact on the following work.

Further, several issues associated with the partition of the space with bounding box are noticed: (a) the positions of some bounding boxes are relatively isolated; (b) the number of points in the bounding box is less because of sparsity, and (c) isolated noisy data points, which are far away from other data collection. In order to solve these issues, a threshold value "T" is used. If the identified points are less than the appointed value after the outward expansion of "T" times, the points in the bounding box are treated as noisy points that can be deleted.

Based on aforementioned information, this paper proposes an approach that is applicable to search in the neighbourhood of uniform and non-uniform distribution point clouds.

Firstly, meshing the point clouds and searching in the grids that potentially will lead to avoid calculating the distance between $P_i$ and other points. Secondly, the maximum and minimum neighbour points are denoted as $P_i$. In the grid that includes the point $P_i$, once the number of points is less than the minimum, one is added to the length of a side of the grid in order to expand the grid until at least "min" points are searched. If the number of points in the grid including the point $P_i$ is greater than the max, then the value is kept at the closest to the max. Thus, the extreme conditions with either too many or very limited number of neighbour points can be avoided. The searching process is shown as Figure 1.

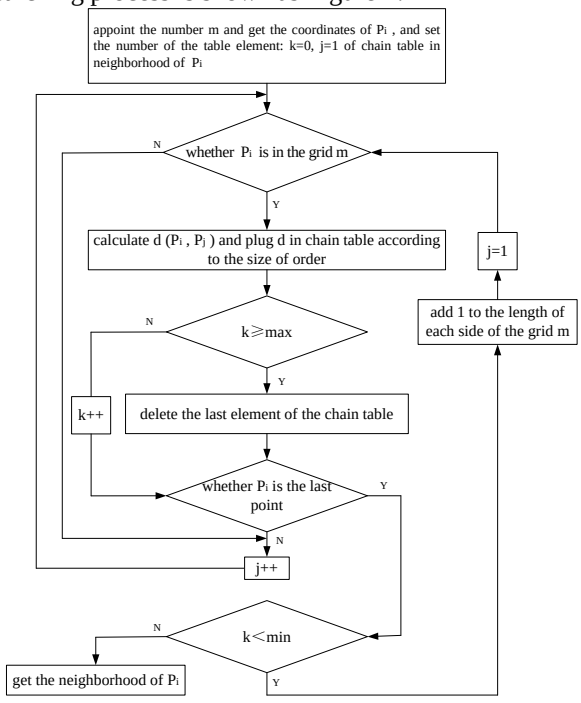


FIGURE 1 T the process diagram of searching neighbourhood

## 2.4 CURVATURE ESTIMATION

Curvature estimation, an important algorithm, has become the thematic topic in areas such as computer vision, computer graphics, geometric modelling and bioengineering. The frequently used methods of curvature estimation are: paraboloid fitting, circle fitting, Gauss-Bonnet, Watallable & Belyaev, Taubin, etc. Among these methods, Gauss-Bonllet can gain optimal Gauss curvature; paraboloid fitting can obtain suboptimum Gauss curvature as well as optimal average curvature. Particularly, paraboloid fitting has optimal stability in the neighbourhood. Thus, paraboloid fitting is adopted in this paper.

Set the paraboloid equation as:

$$z = ax^2 + bxy + cy^2, \tag{16}$$

where $x$, $y$ and $z$ are three coordinates of points on parabolic, and $a$, $b$ and $c$ are the coefficients of the equation.

There are $k$ points in neighbourhood of $P_i$.

The problem to obtain the coefficients a, b and c by the least square paraboloid fitting for $P_i$ and its neighbourhood essentially is a linear least square problem. Householder converter technique can be used to solve the linear equations. The process to estimate the curvature of arbitrary falls mostly into four steps:

***Step 1***: Plug $P_i$ and the points in its neighbourhood into (16). Then the equation set is obtained as follows:

$$AX=Z, \tag{17}$$

where $A$ is the transformation matrix composed of $P_i$ and K points' $x$ and $y$ coordinates in the neighbourhood of $P_i$:

$$A=\begin{pmatrix} x_1^2 & x_1y_1 & y_1^2 \\ x_2^2 & x_2y_2 & y_2^2 \\ x_{k+1}^2 & x_2y_2 & y_{k+1}^2 \end{pmatrix}_{(k+1)\times 3}. \tag{18}$$

$X$ is the transposed matrix of the parabola coefficients:

$$X=\{a,b,c\}^T. \tag{19}$$

$Z$ is the transposed matrix consists of $P_i$ and $K$ points' $z$ coordinate in the neighbourhood of $P_i$:

$$Z=\left|z_1,z_2,\cdots,z_{k+1},\right|_{k+1}^{T}. \tag{20}$$

***Step 2***: Solve (18) and (17) using Householder converter technique and obtain the coefficients a, b, and c of the paraboloid equation.

According to the properties of parabola, we can obtain:

$$K=4ac-b^2 \quad (\textit{Gaussian curvature}), \tag{21}$$

and

$$H=a+c \ (\textit{Mean curvature}). \tag{22}$$

***Step 3***: Repeat steps 1 and 2 until the Gaussian curvatures and the average curvatures of all points are obtained.

### 2.5 IMPLEMENTATION OF THE ALGORITHM

It is common that the point clouds are reduced according to the curvature when reconstructing their surface. The basic principle is to keep a small number of points in a small curvature zone and sufficient points in large curvature zone for reserving the detailed feature of the surface. The method cannot only reduce the number of the points effectively but also keep the features of the surface accurately. In this paper, a module for reducing point clouds based on the curvature has been developed, and the procedure of the algorithm is described below.

Firstly, divide the value of the curvature into many intervals and set the deviation ε based on the curvature. T special condition that the value of the curvature approaches zero (that is, the curve is similar to the line) must be considered.

Secondly, setting the curvature deviation as $\varepsilon$ within one interval. If the point $P_i$ meets the criteria deviation: $|H_j\text{-}H_i| \leq \varepsilon$ ($H_j$ and $H_i$ are the average curvature of $P_j$ and $P_i$, respectively) then delete the point $P_j$. Otherwise, keep the point $P_j$ and set it as the datum point. Repeat this process. This principle cannot only reduce the quantity of the point clouds but also retain the geometrical features of the point clouds better.

The specific realization process is to: (1) search the neighbourhood of each point in point clouds; (2) estimate the curvature in neighbourhood while searching; and (3) reduce the quantity of the points according to the reduced principle in the condition of ensuring the accuracy. The process is illustrated in Figure 2.

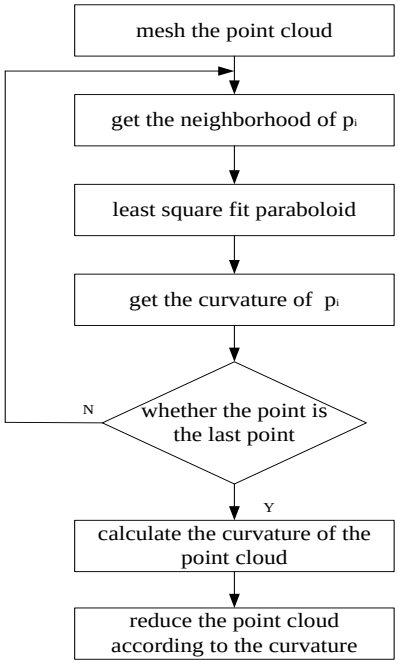


FIGURE 2 The process diagram of reducing point clouds

## 3 Reducing the point cloud of roller bit

In this paper, the method mentioned above is used to reduce the point clouds of roller bit, and the result is shown as Figures 3 and 4.

FIGURE 3 The point clouds of roller bit before reducing

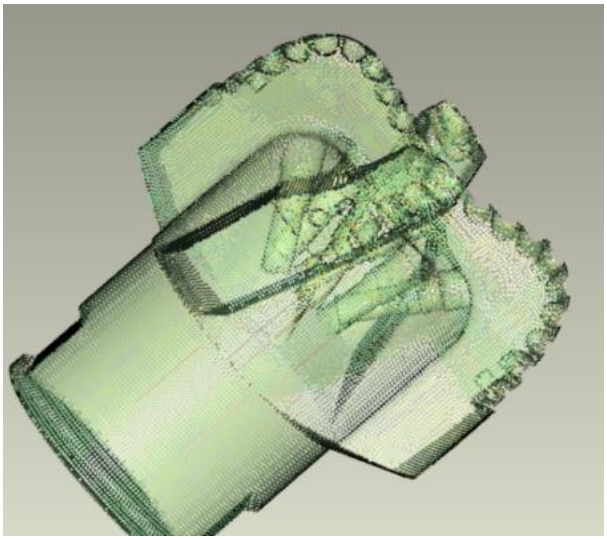

FIGURE 4 The point clouds of roller bit after reducing

## 4 Conclusions

For parts with complex surface in RE process, if the density of metrical data is high, the efficiency of computer is relative low in terms of running, storage and operation. In addition, the smoothness of the reconstructed surface is affected.

This paper focuses on the pros and cons of common algorithms so as to reduce point clouds. It also proposes a new reducing algorithm based on the curvature change.

The case study illustrates that the proposed method performs well in reducing the point clouds with complex surface. Detailed features of the original data are better maintained on the premise of guaranteeing the accuracy and the quantity of the point clouds. These features are very important for further reconstructing surface.

## Acknowledgements

This study is funded by the joint project of school and university of Xinjiang (Grant No. 201204361319) and the Science and Technology Supporting Project of Xinjiang Uygur Autonomous Region (Grant No.: 201304071393) and Science and Technology Major Special Project of Xinjiang Uygur Autonomous Region (Grant No. 201204071134).

## References

[1] Filip D, Magedson R, Markot R 1986 *Computer Aided Geometric Design* **3**(4) 295-311
[2] Barbero B R 2009 *Computers and Industrial Engineering* **56**(4) 1265-75
[3] Liu Tao, Xu Zheng, Sha Chengmei and Zhao Juntian 2009 Curvature estimation of scattered-point cloud data based on bounding box method *Science Technology and Engineering* **9**(12) 3333-6 *(in Chinese)*
[4] Wan Jun, Ju Lu-yue 2004 Data Reduction for Reverse Engineering *Journal of shanghai university (natural science)* **10**(1) 63-69 *(in Chinese)*
[5] Chen Y H, NG CT, Wang Y Z 1999 Data reduction in integrated reverse engineering and rapid prototyping *International Journal of Computer Integrated Manufacturing* **12**(2) 97-103 *(in Chinese)*
[6] Shangguan Jian-lin, Guo San-ci 2011 A study on data reduction in Reverse Engineering *Mechanical Management and Development* **122**(4) 16-18&20 *(in Chinese)*
[7] Shi Baoquan, Liang Jin 2010 Research on point cloud simplification with preserved features *Journal of Xi'an Jiaotong University* **44**(11) 37-40 *(in Chinese)*
[8] Ma Juan, Fang Yuanmin, Zhao Wenliang,Feng Yujin 2011 Algorithm for finding k-nearest neighbors based on spatial sub-cubes and dynamic sphere *Wuhan Daxue Xuebao (Xinxi Kexue Ban)/Geomatics and Information Science of Wuhan University* **36**(3) 358-62 *(in Chinese)*
[9] Vanco M, Brunnett G, Schreiber Th 1999 A hashing strategy for efficient k-nearest neighbors computation *Proceedings of the International Conference on Computer Graphics*, Canmore, Alta., Canada (7-11 June 1999) 120-8
[10] Goodsell G 2000 *Computer Aided Geometric Design* **17**(4) 387-92
[11] Fu Jing, Joshi S B, Simpson T W 2008 *Computer-Aided Design* **40**(3) 311-23
[12] Zhu Maomao, Feng Zujun, Guo Jianhua, Ju Luyue 2005 Data point reduction using a 3-Dimensional cube algorithm *Journal of Shanghai University(Natural Science Edition)* **11**(3) 242-6 *(in Chinese)*
[13] Liu Chun-ming, Fang Yi 2009 An algorithm of searching topological structure from 3D unorganized points *Journal of QingDao University* **18**(3) 20-4 *(in Chinese)*
[14] Wei Chen, Jiayi Shen, Jiabin Xu 2013 Study on a robot 3D laser radar information collection system *Journal of Digital Information Management* **11**(1) 69-75*(in Chinese)*

## Authors

**Hong Jiang, born on December, 1976, Urumqi, Xinjiang, P.R. China**

**Current position, grades:** the Associate Professor of School of Mechanical Engineering, Xinjiang University, China.
**University studies**: B.Sc in Mechanical Engineering from Xinjiang Institute in China, MSc from Xinjiang University in China
**Scientific interest:** digitized manufacture, RE
**Publications:** more than 30 papers
**Experience:** He has teaching experience of 15 years, has completed ten scientific research projects.

**Wenlei Sun, born on October, 1962, Urumqi, Xinjiang, P.R. China**

**Current position, grades**: the Professor of School of Mechanical Engineering, Xinjiang University, China.
**University studies:** BSc in Mechanical Engineering from Xinjiang Institute in China, MSc from Huazhong University of Science and Technology in China.
**Scientific interest**: His research interest fields include CAD/CAM, RE
**Publications:** more than 250 papers
**Experience**: He has teaching experience of 30 years, has completed eighty scientific research projects.

**Yongfang Shi, born on October, 1978, Urumqi, Xinjiang, P.R. China**

**Current position, grades:** the Associate Professor of School of Mechanical Engineering, Xinjiang Medical University, China.
**University studies:** BSc in Mechanical Engineering from Xinjiang Institute in China, MSc from Xinjiang University in China
**Scientific interest**: Her research interest fields include biomanufacturing engineering, RE
**Publications:** more than 16 papers
**Experience:** She has teaching experience of 15 years, has completed eight scientific research projects.

**Yanhua Huang, born on February, 1978, Urumqi, Xinjiang, P.R. China**

**Current position, grades**: the Associate Professor of School of Mechanical Engineering, Xinjiang University, China.
**University studies:** BSc in Mechanical Engineering from Xinjiang Institute in China, MSc from Huazhong University of Science and Technology in China
**Scientific interest:** Her research interest fields include digital design, RE
**Publications:** more than 40 papers
**Experience:** She has teaching experience of 15 years, has completed ten scientific research projects.