International Journal of Hybrid Information Technology
Vol.8, No.9 (2015), pp.157-170
http://dx.doi.org/10.14257/ijhit.2015.8.9.16

A New Kmeans Clustering Algorithm for Point Cloud

Kun Zhang[1,2], Weihong Bi[1], Xiaoming Zhang[2], Xinghu Fu[1] , Kunpeng Zhou[1] and Li Zhu[1]

[1]*College of Information Science & Engineering, Yanshan University*
[2]*Hebei University of Science & Technology*
euphkun@163.com, bwhong@ysu.edu.cn, zhangxiaom@hebust.edu.cn, fuxinghu@ysu.edu.cn, kunpeng032@126.com and gauldothsin@sina.com

Abstract

With development of 3D scanner, it becomes more convenient to access point data. However, for processing the large-scale point cloud, it raised a new challenge for computer graphics. This paper places an emphasis on the point data own characteristics, and then the point data have been divided into certain point sets by clustering algorithm, that is will be divided into different clusters. In order to suit for the point data organization or space division, the clustering algorithm would be improved. This paper provided a new Kmeans algorithm with density constraints. Before processing the point cloud by Kmeans algorithm with density constraints, the density of the point cloud have been defined in this paper, the density of the point cloud can be used for quantification of the convergence. Finally, the Kmeans algorithm with density constraints is verified by the experiment results. Our experiment showed that the improved Kmeans can reduce the processing time, especially, As the increase of the value of K, that is number of cluster, the calculating time of the clustering algorithm can be decreased greatly. In addition, with the increases of the the scale of data size, the stability of the improved Kmeans algorithm has been verified.

Keywords: *Clustering Algorithm; Kmeans; Point Cloud; Contour Estimation*

1. Introduction

At present the point data can be got conveniently by 3D scanner. With CAD (computer aided designer) ,object surface would be reversely reconstructed by point data[1]. Now, it would be as an important, effective and popular method in 3D imaging technique. The reverse reconstruction can be broken down into five steps. 1) data acquisition; 2) data noise-elimination; 3) point cloud reduction; 4) point cloud registration; 5) surface reconstruction.

During the reverse reconstruction of point cloud , how to organized the data more effectively, that is an important thing in data reduction, registration and so on. The reasonable data organization is decreased the query time on the special domain, the time of human-interactive programs and dynamical update.

The method of point data organization can be concluded as two main categories in this paper. The first one is take point data as space data, and we can using the existed software of space database, for example Arcgis, Terrosolid, Polyworks and RealWorks[2]. But these softwares of space database place the emphasis on data processing, then most of them take the original point data as one single file, or take the point data acquired from one device as one single file. So the most time of the software was spent on data processing, for example reduction or reconstruction. The second kind of point cloud data organization is building index by space division [3, 4]. There are many methods for

ISSN: 1738-9968 IJHIT

space division in the literature, such as grid method, quadtree and octree subdivision method, KD-tree.

Grid method[5, 6]: For many organizations, grid computing may be a simple and viable method. Grid algorithm is to divide space to average size as 2D. The algorithm cannot analyze the inherent correlation in data itself. Therefore, It is an efficient method during building grid index. However, facing to the non-uniform distribution and scattered point data, the grid method cannot provide an efficient index process.

Quadtree and octree subdivision method [7]: Simply stated, the quadtree and octree is manager the space using tree method. From space dividing technology, using quadtree and octree, the data are respectively computed as 4 directions and 8 directions, the information of subspace stored in a tree of nodes. Relative to grid method, quadtree and octree can receive more details in subspace. So the processing of search the special data, which you want to, has low time complexity. However, using quadtree and octree to manage space, it would lead to the resource waste, especially, for non uniform point clouds.

KD-tree[8]: In order to store high-dimensional data, KD-tree method had been provided. This method organizes D-dimensional data as a binary tree. Using value of point data, the space can divided. In a tree hierarchy, one dimension can be divided by the value of point data, on the left branch in the binary tree the data less than the parent's data will be sorted. On the other hand, the right branch in the tree store the data more than the parent's data. Then once the D-dimensional space divided operating, there will be completed at least in D hierarchies of tree establishment. The KD-tree used wildly in computing graphic, especially in the nearest neighbor query of the spatial database. But, the KD-tree method is sensitive to the order data inserted. On the other hand, it's hard to control balance of KD-tree.

By considering data characteristic itself, this paper provided a new Kmeans clustering algorithm. Firstly, the point clouds have been processed using original Kmeans algorithm. We found the original Kmeans cannot be applicable to point cloud. Especially, as the bigger value of K, the algorithm spends more time on clustering operation. In order to improve performance of algorithm, and suit for data pre-processing capabilities, a new Kmeans clustering algorithm, Kmeans algorithm with density constraints, have been provided in this paper. Then, Kmeans algorithm with density constraints will produces a new form of result. And the data would not be convergent to one cluster. So, the intersection cluster, adjacent cluster and isolation cluster will be defined. The next, the point cloud can be organized as the new defined set of cluster. At last, experiments are carried out and successfully verify the feasibility and efficiency of Kmeans algorithm with density constraints.

2. Related Work

The Kmeans algorithm with density constraints method is suitable for an uneven distributed and a huge amount of data. Because point data, which we choose, just is a small set of scanning data with concentrated distribution and original surface changed. And this feature give us a hit that if the distance of original points as an affecting factors involved into data organization, the reconstruction algorithm speed would be speed up[9]. Hence, the clustering algorithm has been applied to point cloud. The process of point data organization will be shown as Figure 1.

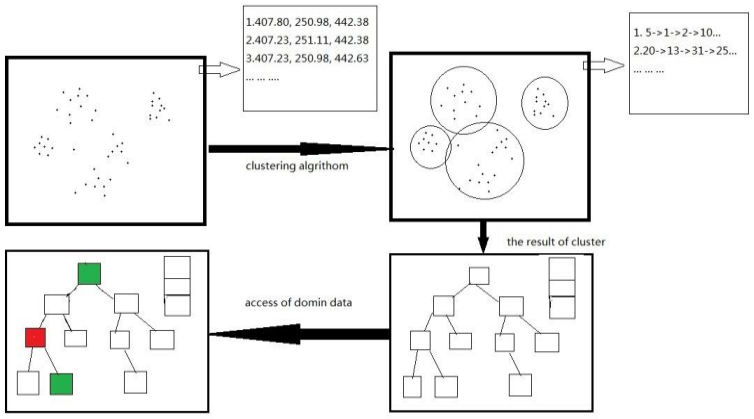

Figure 1. Principle Diagram of Point Cloud Storing Algorithm

Clustering algorithm is an important method of data mining technology. The purpose of clustering operation is to put the similar object through the method of static classification into different categories or a different subset. So, the same subset of objects will have the similar the properties. The common approaches are as follows: hierarchical clustering, partition clustering and density clustering, *etc*.

Kmeans[10]: Kmeans is a clustering algorithm for multi-dimension. It is a non supervised machine learning method and it has been applied in many fields. Because of the effective and non-requirement for prior knowledge, we choose the Kmeans to per-processing point data. The main shortcoming of Kmeans is the objects in algorithm limited in numeric data, and algorithm is easily trapped in the local optimum.

Objective function of Kmeans is as equation (1) and equation (2).

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \left\| x_n - \mu_k \right\|^2 \qquad (1)$$

$$r_{nk} = \begin{cases} 0 & x_n \notin K(\mu_k) \\ 1 & x_n \in K(\mu_k) \end{cases} \qquad (2)$$

In equation (1), x_n represents point data, N can be expressed as data size. And, the K is the number of cluster in result. μ_k is the center of one cluster. Then, the Kmeans method is to find a value of μ_k and r_{nk}, it can make the objective function converge to local minimum, which can get the minimal J. The pseudo-code algorithm of original Kmeans is as follows.

Step1. center of the initial cluster, $\{ \mu_k \mid k = 1 \ldots K \}$, randomly chosen ;

Step2. Repeat

Step3. Divide every data into one cluster, and μ_k as the center of the cluster;

Step4. Recomputing the new μ_k, that is the new center of the cluster;

Step5. Until getting the stable set of cluster or recomputing counts more than maximum number of iteration.

Traditional Kmeans algorithm can convergent to one cluster, using iterative operation. And the stable cluster presents the end of the iteration operation. That means the more the

iteration counter, the more time that the algorithm spent. For point cloud data, the Kmeans clustering operation worked on the 3-dimension, that is X, Y, Z in geodetic computation. The Euclidean distance formula can be used to calculate the distance between the point parameter. The formula is as follows:

$$dist(\mathrm{x},\mathrm{y}) = \sqrt{\sum_{i=1}^{n} (\mathrm{x}_i - \mathrm{y}_i)^2} \tag{3}$$

In order to find data domain fast and conveniently and improvement the speed of human-computer interaction, this paper took into account the Kmeans clustering algorithm by the data itself feature. The convergence of Kmeans clustering algorithm obviously exists some inappropriate to point cloud data. In this paper, since the data domain is important to the data organization, the point cloud data convergent to one cluster is not suitable way to the data per-processing. In this paper, the third part illustrates the improved Kmeans clustering algorithm with density constraint for the per-processing to point cloud.

3. Kmeans Clustering Algorithm with Density Constraint

Based on Kmeans clustering algorithm, this paper set the end condition of cluster density, that makes some data cannot be reach convergence on one cluster. Lucky, the purpose of us is to find the data domain, and the data set which convergence on two or more cluster, and that means these cluster is adjacent data.

In the fourth part, the data organization would be provided by clustering algorithm with density constraint. The improved algorithm pseudo-code is as follows:

Step1. $\{\mu_k \,|k = 1 \ldots K\}$ Center of the initial cluster randomly chosen;

Step2. While $((\mathrm{P}_k > \sigma) \vee (\mathrm{NumIteration} > \mathrm{Max}))$

Step3. Computing distance of each data to μ_k , that is D_k , and μ_k as the center of the cluster;

Step4. Based the difference of distance of D_k , recompute the new μ_k , that is the new center of the cluster;

Step5. Then the value of cluster density will be estimated, that is P_k ;

Step6. Getting the stable set of cluster.

3.1 The Center Shifting In The Cluster

MS（Mean Shifting）: In 1975, Fukunaga[11] provided the MS algorithm, then this algorithm have been found an increasingly wide utilization in computer graphic, as for the fast convergence rate speed and without prior knowledge support.

Assuming that the kernel vector function in MS algorithm $K(\mathrm{x})$, as follows:

$$K(\mathrm{x}) = \begin{cases} 1 & (\|x\| \le \lambda) \\ 0 & (\|x\| > \lambda) \end{cases} \tag{4}$$

In equation (4), the given width of data as λ , and x as independent variable.

Based on the equation (4), the mean shifting of muti-dimensional data using MS algorithm can be described as equation (5).

$$m(x) = \frac{\sum_{i=1}^{n} K(x - x_i) x_i}{\sum_{i=1}^{n} K(x - x_i)} - x \tag{5}$$

Then the cluster center in the last iteration clustering algorithm processing can be adjusted. As for the Kmeans clustering, the value adjusted as follows.

$$m(\mu_k) = \frac{\sum_{i=1}^{n} K(\mu_k - x_i) \mu_k}{\sum_{i=1}^{n} K(\mu_k - x_i)} - \mu_k \tag{6}$$

3.2. The End Condition of Kmeans with Density Constraint

The computing of density is an issue of crucial importance in the improved Kmeans algorithm. Because the computing set of cluster, the relevant characteristic of point clout is some vital reference data. And he relevant characteristic of point clout can reach to evaluate the density.

Point Cloud Density: In order to set end condition of the iteration, we define the density in cluster. The density can be computed based the euclidean distance of data in one cluster.

Because of the arbitrariness of cluster contour, the reference [12] provides the density of point cloud which can not suitable this clustering algorithm.

DEDINITION 1: Contour Data Approximate Presentation. Given K orthogonal planes Π_k, the point cloud, $c_i (i = 1 \ldots n)$, would be set up an projection in one plane, and the projection of the point cloud can be decomposed into d different directions. Then the $\min(c_{ikd})$ and $\max(c_{ikd})$ can be defined as contour data of the point cloud, $c_i (i = 1 \ldots n)$.

LEMMA1: The Area Approximate Evaluate. Given set of projection plane Π_k, and the contour data $\min(c_{ikd})$, $\max(c_{ikd})$. Then through equation (7), the area of the contour data in one projection plane, can be evaluated.

$$S_{\Pi_k} = \frac{1}{2}\left(\sum_j \begin{vmatrix} \max(c_{idjx}) - o_{kx} & \max(c_{idjy}) - o_{ky} \\ \max(c_{id(j+1)x}) - o_{kx} & \max(c_{id(j+1)y}) - o_{ky} \end{vmatrix} + \sum_j \begin{vmatrix} \min(c_{idjx}) - o_{kx} & \min(c_{idjy}) - o_{ky} \\ \min(c_{id(j+1)x}) - o_{kx} & \min(c_{id(j+1)y}) - o_{ky} \end{vmatrix}\right) \tag{7}$$

In equation (7), variable S_{Π_k}, represents the area of the the k projection plane. The j is as the *jth* orthogonal direction in that plane. And o stands for the center of point cloud projection of plane. So, the x and y stands for the *jth* direction and the orthogonal direction, respectively.

LEMMA2: The Density Approximate Evaluate in Cluster. Given set of projection plane Π_k, and the contour data, $\min(c_{ikd})$ and $\max(c_{ikd})$. Then through equation (8), the density in the cluster, can be evaluated.

$$P(c_i) \approx \frac{\sum_k S_{\Pi_k} \times E_{\bar{n}_k}(c_i)}{n} \tag{8}$$

$$P(c_i) \approx \frac{\frac{2}{k}\sum_j \prod_{d/2} \frac{\left\|\max(c_{ikd}) - \min(c_{ikd})\right\|}{d}}{n} \times E_{\bar{n}_k}(c_i) \tag{9}$$

In equation (8, 9), $E_{\bar{n}_k}(c_i)$ is the mathematical expectation of the normal vector of Π_k on the k projection plane.

The end of the condition for the improved Kmeans algorithm is density of the cluster. In general, the threshold of the density σ, can be computer as the parameter of the device(3D scanner) and the movable platform, and these uncertainty data have been discussed in the reference[13] .

Table 1: An Algorithm for Areas Estimation Algorithm S_{Π_k}

Algorithm: Estimation Areas($c_i, \vec{x}_k, \vec{y}_k$)

Input: 1):Point Cloud Data, c_i

2):orthogonal vector, $\vec{x}_k$ and $\vec{y}_k$;

Output: The areas estimation algorithm , S_{Π_k} of point cloud data c_i on the projection planes Π_k

1. $c_{\Pi_k} = Data\,Proj(\vec{x}_k, \vec{y}_k)$
2. $o = MS(c_{\Pi_k})$
3. $c_{kd} = OrthoDivi(c_{\Pi_k}, d/2)$
4. $contour_i = \{\min(c_{ikd}), \max(c_{ikd})\}$
5. $S_{\Pi_k} = \sum AreaFun(contour_i)$
6. Return(S_{Π_k})

Through the orthogonal vector $\vec{x}_k$ and $\vec{y}_k$, the set of projection plane Π_k, can be find out. And c_{Π_k} can be computed as the c_i projection on the plane Π_k. Then, using MS algorithm the center of the cluster, $o = MS(c_{\Pi_k})$, can be adjusted. By the input value of decomposed direction d, notice that d is an even number, and the d/2 orthogonal direction can be find out. That c_{kd} is the the direction component of c_{Π_k} in the d different direction, In algorithm the function can be described as $c_{kd} = OrthoDivi(c_{\Pi_k}, d/2)$; The contour

data approximately presented as the set of $\min(c_{ikd})$ and $\max(c_{ikd})$, $contour_i = \{\min(c_{ikd}), \max(c_{ikd})\}$.

Finally, by the equation (7), the areas estimation algorithm S_{Π_k} of Point Cloud Data c_i on the projection planes Π_k can be computed.

This paper provided two methods for density estimated. Table 2 is the density algorithm based areas P, It is intuitively $\min(c_{ikd})$ found out based on EstimationAreas algorithm. Though, the process of the algorithm based on areas, we provided a new method to density estimated, as the table 3 shown. It can spend less time than the algorithm for density algorithm based areas P.

Table 2. An algorithm for Density Algorithm based Areas P

Algorithm: EstimationDensitysBasedArea($coutour_i$ **,** Π_k **)**

Input: Contour data, $\min(c_{ikd})$ and $\max(c_{ikd})$;

The set of projection plane Π_k.

Output: density algorithm based areas P

1. $J = IndexDirection(\Pi_k)$
2. $\vec{n}_k = \vec{x}_k \times \vec{y}_k$
3. $S_{\Pi_k} = EstimationAreas(c_i, \vec{x}_k, \vec{y}_k)$
4. $E_{\vec{n}_k}(c_{ik}) = MeanNormal(\vec{n}_k, c_i)$
5. $p = EstimateDensity(S_{\Pi_k}, E_{\vec{n}})$
6. Return(p)

Building the index of the d different directions on the projection plane Π_k, the index mark is presented as j. Then, the normal of the vector can be computed as the equation $\vec{n}_k = \vec{x}_k \times \vec{y}_k$. In the algorithm, using algorithm for areas estimation algorithm, the S_{Π_k} can be got. Along normal direction of projecting plane, the point cloud data, c_i, can be projected; Then the $E_{\vec{n}_k}(c_{ik})$, mathematical expectation of the projection vector module, can be computed. Finally, the density algorithm based areas P can be computed as equation (8).

Just like the algorithm of the EstimationDensitysBasedArea, in the algorithm of the EstimationDensitysBasedCoutour, shown in table3, the normal and the index have to be built firstly. Instead of estimating the areas of projection, algorithm of EstimationDensitysBasedCoutour compute $\Pi\|\bullet\|$ of contour data which the index number is d_i and $d_{i+d/2}$, Finally, we can computed the density based on equation (9). Obviously, the time complexity of this algorithm is lower, because not need to computing the areas of projection.

Table 3. An Algorithm for Density Algorithm Based Outline P

Algorithm: EstimationDensitysBasedCoutour($coutour_i$ **,** Π_k **)**

Input: Contour data, $\min(c_{ikd})$ and $\max(c_{ikd})$;

The set of projection plane Π_k.

Output: density algorithm based areas P

1. $J = IndexDirection(\Pi_k)$
2. $\vec{n}_k = \vec{x}_k \times \vec{y}_k$
3. $normvalue = NormNormal(\text{coutour}_i, d)$
4. $p = EstiamteDe\quad nsity\ (normvalue\quad, E_{\vec{n}})$
5. Return(p)

4. Data Organization by Set of Cluster

In the data mining field, visualization of the data mining processing is a hot academic branch, especially, the visualization of data mining result. The visualization can improve man-machine interactive capability and increase the data value. For example, the scatter chart, parallel coordinates, matrix, and so on.

4.1. The Relation of Cluster

Using Kmeans algorithm with density constraint, the set of cluster can be getting. And not all of the clusters is dependent, there is correlation among the cluster, so the cluster organization can be described a forest. To simplify, we qualify the degree of the close correlation. And we organization the cluster as a tree and a table can be store as the dependent clusters. Therefore, we defined the correlation of the set of cluster. Though the definitions, we can compute the degree of the close correlation obviously.

DEFINITION 2: (intersection cluster) Given the one cluster K_1 and the other cluster K_2 are intersection clusters, if and if only, there exits one direction, on that direction, the contour data can satisfy the expression, as follows.

$$((\min(c_{ik_1d}) < \min(c_{ik_2d})) \wedge (\max(c_{ik_1d}) > \min(c_{ik_2d})) \tag{10}$$

Beside, the intersection degree can be seen as the number of the contour data, which satisfy the expression just mentioned.

DEFINITION 3: (adjacent cluster) Given adjacent threshold Δ, the contour data on planes Π_k would be sorted as: $sort(\min(c_{ik_1d}), \min(c_{ik_2d}), \max(c_{ik_1d}), \max(c_{ik_2d}))$. Then the one cluster K_1 and the other cluster K_2 are adjacent clusters, if and only if, the distance of the contour data less than threshold Δ.

DEFINITION 4: (isolation cluster) Given adjacent threshold Δ, and the contour data processing as: $sort(\min(c_{ik_1d}), \min(c_{ik_2d}), \max(c_{ik_1d}), \max(c_{ik_2d}))$, the one cluster K_1 and the other cluster K_2 are isolate, if and only if, the distance of the contour data more than threshold Δ.

4.2. Algorithm of Cluster Tree

As it described in the previous section, the set of cluster, can be divided into three parts. The subset of intersection clusters and adjacent clusters is important for neighbor of data domain. Of course, the correlation can be stored as a forest. However, as we all

know, it is not a good method for building and searching. We change the forest into a cluster tree and an isolate table.

Firstly, the sort of contour data have been carry out, the processes can be described as the expression：

$$sort(\min(c_{ik_1d}), \min(c_{ik_2d}), \max(c_{ik_1d}), \max(c_{ik_2d})) .$$

On basis of the result of sort process and the threshold Δ, the adjacent and isolate relation of the clusters can be find out. Because there is no any relation in isolate clusters, the cluster can be organized as an isolate table. The data organization algorithm pseudo-code of cluster tree is as follows:

Step1: $sort(\min(c_{ik_1d}), \min(c_{ik_2d}), \max(c_{ik_1d}), \max(c_{ik_2d}))$;

Step2: Find the subset of pairs of intersectional clusters and adjacent clusters;

Step3: Put cluster which is the most frequent on the pairs in the intersection and adjacent subset, as the cluster tree root;

Step4: The left of the parent is the correlation just as intersect and adjacent clusters;

Step5: The rightmost leaf of tree is the isolate with the parent and there is the other correlation with its left child.

For example, simplify the data as the scatter 2D point as Figure 2. And the data will be organized to the cluster tree as Figure 3.

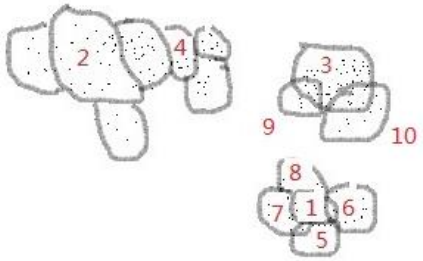

Figure 2. The Scatter Point Data and the Cluster with Density Constraint

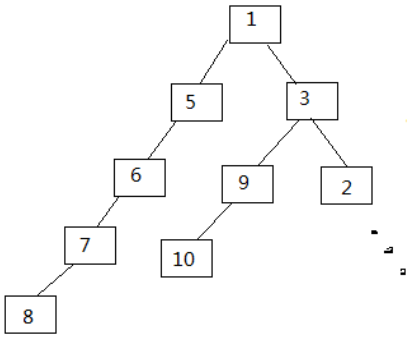

Figure 3. Part of the Cluster Tree

5. Experiment

All the experiments are executed on the same data set with different algorithms. The purpose of the experiments is illustrating the feasibility of the Kmeans with density algorithm. And through result the experiment, we can know the two aspect performance of the improved Kmeans algorithm. (1) In the contrast of the clustering effect, especially for contour; (2) Running time when the improved Kmeans and the original Kmeans algorithm execute on different data size.

5.1. Experiment Setup

Data set of data organization experiment:

1) Bunny rabbit from Stanford University, there is 35947 points.

2) By sick LMS100 scanner, the experiment data can be received. Data size: 10000-168711.

The parameter of sick LMS100 can be shown as the Table 4.

Table 4. The Parameters of LMS100

	wavelength	Measurement frequency	Measurement distance	angle	Statistical error	the angular resolution 25HZ	50 HZ
The Lest	—	25HZ	0.5m	—	—	0.25°	—
Standard	905nm	—	—	—	±12 mm	—	0.5°
The Most	—	50HZ	20m	270°	±20 mm	0.5°	—

The experiment can be executed as shown in the Figure 4.

Figure 4. Environment of Experimental Operation

5.2. Experiment Results

This paper provided a new algorithm, relative to Kmeans clustering algorithm, and it is feasible to capture satisfied cluster set. At first, to verify the new algorithm, we choose the open data which offered by Stanford University Computer graphics Laboratory. The traditional Kmeans algorithm which obtained by Matlab R2010b, which have been executed with K=100, then the size of cluster set is 100, and the result have been shown as Figure 5. The data have been divided into different cluster, and the neighbor cluster set can provide the data domain, and it is important to point cloud process. The running time can be seen in Figure 9 through red curve.

The new algorithm has been completed, with the same data set. And the effect of cluster algorithm has been shown as Figure 6, Then, form Figure 6, you can seen the contour data will be find the correlative of cluster sets. And these intersect data will be

stored in one 2d matrix, as the intermediate result. The running time can be seen in Figure9 through blue curve.

In order to test the change of result of cluster, with the bigger K value. The experiment will be executed by changing K value. Then the Figure 7 and Figure 8 indicate the different result of the two different algorithms.

Figure 5. The Result of Cluster Sets as K=100 through Traditional Kmeans

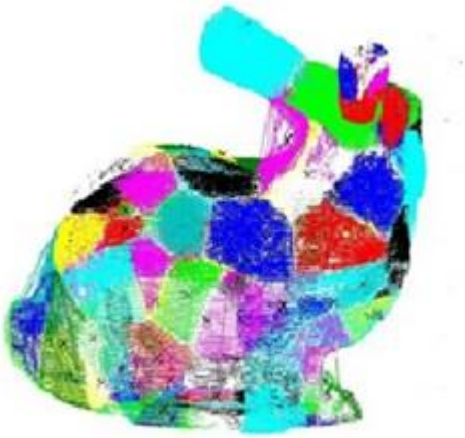

Figure 6. The Result of Cluster Sets as K=100 through Improved Kmeans

Figure 7. The Result of Cluster Sets as K=1000 through Traditional Kmeans

Figure 8. The Result of Cluster Sets As K=100 through Improved Kmeans

The detail of Figure 5, 6 and Figure7, 8 shows the contrast of the cluster sets with different color. And the results are closer to reality the actual distribution of the target object. With iteration number decreased significantly, the improved Kmeans algorithm reduces less cluster sets than the traditional ones. So, it makes the decrease of space and time complexity using the data cluster organization to point cloud.

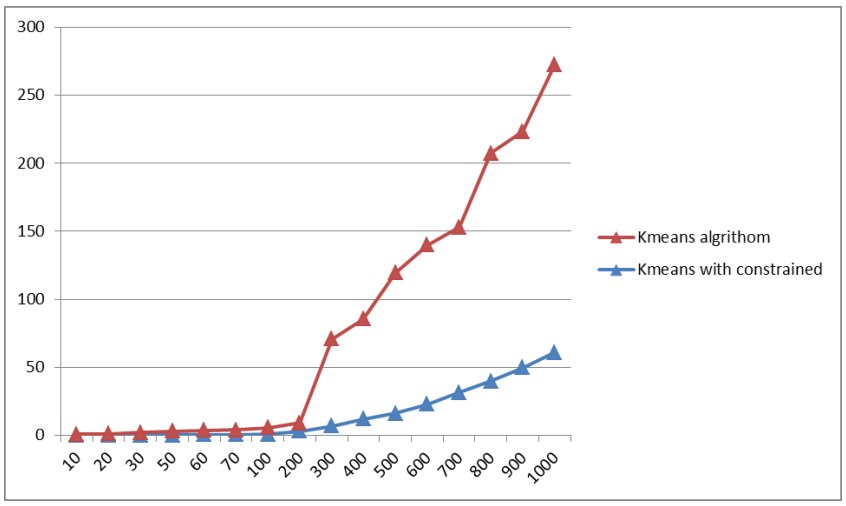

Figure 9. The Contrast of the Running Time

Figure 9 shows the contrast of running time of the traditional Kmeans algorithm and the improved Kmeans, and the blue curve represents the time of improved Kmeans consumed, and the red curve shown running time of the traditional algorithm. So, the improved Kmeans algorithm that this paper provided, can proposed an available method for point clouds per-processing, especially for the small scope data domain, and this is important to point cloud feature extraction and selection.

With the increasing of data size, we performed an experiment, too. The data can be reached by the Sick LMS100 3D scanner. And we choose a different data volume, the running time as the table 5 shown.

Table 5. Running Time with the Changing of Data Size

Data size ($\times 10^4$) \ K	10	100	1000
1	0.066804	1.131637	19.166697
	0.071701	1.051197	16.544950
3	0.7646	4.778202	211.50758
	0.059079	0.930863	60.819052
6	0.251096	7.879803	525.142863
	0.100600	5.744732	130.906910
10	3.526699	74.286126	488.722703
	0.219568	4.177856	177.691124

Notes: the unit of time of the algorithm is a second. The higher layer in the table is the time of traditional Kmeans, and the lower on it to the improved Kmeans which is provided in this paper.

The advantage of the improved algorithm that this paper provided can be seen in table 5. Especially, with the large data volume, the Kmeans with density constraint have stable performance characteristics.

6. Conclusions and Acknowledgments

In this paper, in order to adapt to the point cloud data processing, we improved the traditional Kmeans algorithm. Instead of the Euclidean distance, the point cloud density has been computed as a condition of convergence. Similarly, threshold data have to be set for converge, but the convergence does not stead for the one point divided into one cluster. Therefore there will be a correlation of sets of cluster. And this can reflect neighboring of data domain. What's more, it is precisely these geometric position relationship between clusters can be we used for point cloud data of the organization. So, using the correlation between clusters, cluster tree structure can be used for organization of the point cloud. The algorithm is proved to be feasible through the experiment. Due to algorithm based on Euclidean distance, the algorithm may fall into local optimum, but the constraint of density can avoid it at some degree. In the future, we'll continue to analysis features of data, data reduction would be executed based on the cluster organization.

Acknowledgements

The paper is supported by Hebei Natural Science Foundation (No. F2013208107), National Natural Science Foundation of China (No.51271033, No.61440012), Hebei Science and Technology Program (No.14K50120D), Science Foundation of Hebei Education Department (QN2014174), and Top-notch young talent Foundation of Hebei Province (No. BJ2014023).

Reference

[1] D. Barber, J. Mills, and S. Smith-Voysey, "Geometric validation of a ground-based mobile laser scanning system," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 63, pp. 128–141, **(2008)**.

[2] L. Truong-Hong and D. F. Laefer, "Octree-based, automatic building façade generation from LiDAR data," Computer-Aided Design, pp. 46–61, **(2014)**.
[3] H. S. Park, Y. Wang, E. Nurvitadhi, J. C. Hoe, Y. Sheikh, and M. Chen, "3D Point Cloud Reduction Using Mixed-Integer Quadratic Programming," pp. 229-236, **(2013)**.
[4] Y. Chang-qiang, Y. Ze-tian, L. Xiu-shan, and W. Yu, "Transforming vehicle-borne laser point cloud to grid point cloud," Science of Surveying and Mapping, **(2009)**.
[5] J. Zhao, C. Long, S. Xiong, C. Liu1, and Z. Yuan1, "A New K Nearest NeighboursAlgorithm Using Cell Grids for 3D Scattered Point Cloud," ELEKTRONIKA IR ELEKTROTECHNIKA, vol. 20, pp. 81-87, **(2014)**.
[6] S. Peters, "Quadtree- and octree-based approach for point data selection in 2D or 3D," Annals of GIS, vol. 19, pp. 37-44, **(2013)**.
[7] Y. Jiansi, "A Method of Combing the Model of the Global Quadtree Index with Local KD-tree for Massive Airborne LiDAR Point Cloud Data Organization," Geomatics and Information Science of Wuhan University, vol. 39, pp. 918-922, **(2014)**.
[8] B.-Q. Shi, J. Liang, and Q. Liu, "Adaptive simplification of point cloud using k-means clustering," Computer-Aided Design, vol. 43, pp. 910-922, **(2011)**.
[9] Z. Huang, "Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values," Data Mining and Knowledge Discovery, vol. 2, pp. 283-304, **(1998)**.
[10] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," Nformaon Hory Ranaon on, vol. 21, pp. 32 - 40, **(1975)**.
[11] P. Jenke, M. Wand, M. Bokeloh, A. Schilling, and W. Straßer, "Bayesian Point Cloud Reconstruction," Computer Graphics Forum, vol. 25, pp. 379–388, **(2006)**.
[12] K. Zhang, W. Bi, and X. Fu, "A New Method of Point-Clouds Accurate Measurement and Reconstruction," International Journal of Database Theory and Application, vol. 7, pp. 81-94, **(2015)**.

