Denoising Algorithm for Bilateral Filtered Point Cloud Based on Variance Threshold

Linzhe Chen[1, a], Baohong Feng [1, b]

[1] *College of Geomatics Engineering, Nanjing Tech University, Nanjing 211800, China*
[a] *798942031@qq.com,* [b] *fbhhll@163.com*

Keywords: Points Cloud denoising; Bilateral filtering algorithm; K neighbor search; Variance threshold

Abstract: Based on the traditional K neighborhood search algorithm, a K neighborhood search algorithm based on space cell is proposed, which changes the construction of the k-d tree and improves the search speed of the nearest neighbor (k) neighbor. In view of the denoising of space scattered point clouds, a data point classification method based on variance threshold judgment is proposed, which is based on the improvement of bilateral filtering algorithm. The experiment shows that the algorithm is better than the bilateral filtering algorithm, and the defect of the bilateral filtering algorithm is improved to a certain extent under the condition that the processing speed is greatly improved, and it has strong practicability.

1. Introduction

In the original point cloud data directly obtained by the 3D laser scanning equipment, the loss of noise points, the loss of data points, and the loss of the point cloud geometry are unavoidable. In order to meet the requirements of the quality of the acquired cloud, some pretreatment is needed before using these point cloud data. Among them, denoising of point cloud data is a very important part. The effect of point cloud de-noising determines the accuracy of the data processing in the subsequent point, and the geometric model obtained after the surface reconstruction is often rough, so the point cloud denoising algorithm plays an indispensable role in the point cloud processing. In recent years, many scholars have studied the denoising algorithm of spatial point clouds.

Wang [1] proposed a point cloud de-noising algorithm based on the moving least square method. By moving the sampling point to the corresponding local surface, the noise can be eliminated. The algorithm is robust, but it does not have very strong time complexity. Rosman [2] use the Laplace Belturm operator to generate smooth surface to process high frequency noise, but reduce the accuracy. Xu [3] can effectively improve the efficiency of point cloud denoising based on the segmentation and denoising of point cloud geometric features, but it does not pay attention to the retention of geometric features of objects. Yang [4] proposes a system smoothing algorithm for 3D point cloud denoising, which achieves the purpose of point cloud denoising and better maintains the surface features.

Taubin[5] applies the Laplasse operator in the digital image processing to the point cloud triangular mesh model, and proposes a triangular mesh denoising method based on the Laplasse operator. Alexa [6] uses the moving least square method to fit the neighborhood of any point in the cloud, and then moves the point to the corresponding fitting surface to eliminate the noise, but the algorithm has a large and complex calculation.

In the point cloud denoising method, bilateral filtering has better feature retention and smoothness. By controlling the size of the neighborhood of each point in the three dimensional point cloud data, the model can be maintained, but the large range noise can not be processed, and there is a certain probability of loss of the feature [7]. In addition, only using bilateral filtering to denoise the three-dimensional point cloud data with complex surface geometry, it is difficult to keep the feature rich in the feature rich area of the 3D point cloud data model, and at the same time, it is highly smooth in the flat area with the lack of features. Aiming at the shortcomings of bilateral filtering, a bilateral filtering algorithm of point cloud based on variance thresholding is proposed in this paper. First, a topological relationship is established for point cloud, and then the neighborhood information of data points can be obtained and calculated. Then, the data points in the point cloud are classified according to the variance of the distance from the fitting plane to all points in the point cloud. For different regions of the point cloud, the algorithm will adopt different denoising algorithms.

2. Improved K Neighborhood Search Algorithm Based on Space Cells

The traditional K neighborhood search algorithm is to establish the k-d tree for all data points in the point cloud, which is unnecessary in practical application, resulting in the waste of computer resources. In order to optimize this problem, an improved neighbor search algorithm based on spatial cells is proposed in this paper.

First, based on the three dimensional values of all point cloud data, a space encircling box which is parallel to the X, Y, and Z axes, the starting point and the end point of each coordinate axis is the minimum and maximum value of the three axis direction of the point cloud data points.Then the bounding box is evenly divided into several small space cells, and the space cells are carried out. Numbered. When searching the K neighborhood of a point in a point cloud, we first judge which cell in the space by the coordinates of the point, and then establish the k-d tree for the data points in the space cell in the surrounding space, which can greatly save the computer resources and improve the search efficiency. In practice, in order to ensure that the number of points in the K neighborhood is enough, select $4 \times 4 \times 4$ cells in the cell and its surrounding area to carry out the algorithm. The number of space cells in the direction of the X, Y, Z axis and The ordinal number of the space cell at any point in the direction of X, Y, Z axis. can be calculated by the equations below After obtaining the serial number of the space cell, the 63 space cells around it are obtained. Then the k-d tree is constructed according to the data points in the 64 space cells. Finally, the K neighborhood of the point will be searched on the the built k-d tree.

$$\begin{cases} count_x = (x_{\max} - x_{\min}) / s \\ count_y = (y_{\max} - y_{\min}) / s \\ count_z = (z_{\max} - z_{\min}) / s \end{cases} \tag{1}$$

$$\begin{cases} Num_x = (x - x_{\min}) / s \\ Num_y = (y - y_{\min}) / s \\ Num_z = (z - z_{\min}) / s \end{cases} \tag{2}$$

3. Calculation of Variance

For any point $p(x,y,z)$ in the space point clouds, the K neighborhood $\{N_p : p_i(x_i, y_i, z_i) \in R^3, i = 1, 2, ..., k\}$ can be obtained by the algorithm described above. Its least square fitting plane is $Ax_i + By_i + Cz_i + E = 0$. Then the distance from any point p_i to the least squares fitting plane in the K neighborhood of P point can be calculated by the equation below.

$$d_i = \frac{|Ax_i + By_i + Cz_i + E|}{\sqrt{A^2 + B^2 + C^2}} \tag{3}$$

We assume the $M_j = \{d_1, d_2, ..., d_i\}, i = 1, 2, ..., k$, is the distance set for fitting the least squares plane from the point j to all points in its neighborhood, and its variance is $D(M_j)$. Then the Mean variance can be calculated by the equation below.

$$\overline{D} = \frac{\sum_{j=1}^{n} D(M_j)}{n} \tag{4}$$

4. Point Cloud Denoising Based on Threshold Judgment

We assume that the variance of the distance of the fitting plane of the data point in the point cloud of the flat region to the middle point of its neighborhood is D_1, and the variance of the distance between the data points of the point cloud in the distinct region and the fitting plane of the middle point of its neighborhood is D_2. According to the least square plane fitting principle, the more smooth the distribution of point cloud is, the closer the fitting plane is to the actual distribution of point cloud. So we can know that $D_1 < \overline{D} < D_2$.

In this paper, the threshold value of $\overline{D}$ is used to judge whether a point in a point cloud belongs to a flat area or a characteristic region. If $D(M_j) < m * \overline{D}$, the point j belongs to the flat area, otherwise it belongs to the characteristic region.

4.1 Point cloud denoising in the flat area

The plane projection is used in this part. All the points, including data pionts and noise points, are moved to the least square plane. The point coordinates after denoising can be calculated by the equations below.

$$\begin{cases} x_1 = x_0 - KA \\ y_1 = y_0 - KB \\ z_1 = z_0 - KC \end{cases} \tag{5}$$

$$K = \frac{|Ax_0 + By_0 + Cz_0 + E|}{\sqrt{A^2 + B^2 + C^2}} \tag{6}$$

4.2 Point cloud denoising in the characteristic region

Traditional bilateral filtering denoising algorithm is used in this part. The point coordinates after denoising can be calculated by the equations below.

$$p' = p + n*d \tag{7}$$

$$d = \frac{\sum_{i=1}^{N} W_C(\| p_i - p \|)W_s(\|< n, p_i - p >\|) < n, p_i - p >}{\sum_{i=1}^{N} W_C(\| p_i - p \|)W_s(\|< n, p_i - p >\|)} \tag{8}$$

The process of the algorithm in this paper can be described by Figure 1.

```
                    ┌──────────────┐
                    │    Begin     │
                    └──────┬───────┘
                           │
                           ▼
                    ┌──────────────┐
                    │ Calculate the K │
                    │  neighborhood   │
                    └──────┬───────┘
                           │
                           ▼
                    ┌──────────────┐
                    │ Calculate the │
                    │   variance    │
                    └──────┬───────┘
                           │
                           ▼                No      ┌──────────────────┐
                    ◇ D(M_j) > D̄ ◇ ──────────────▶│ Plane Projection │
                           │                         └──────────────────┘
                          Yes
                           │
                           ▼
                    ┌──────────────┐
                    │ Bilateral filtering │
                    └──────────────┘
                           │
                           ▼
                    ┌──────────────┐
                    │     End      │
                    └──────────────┘
```
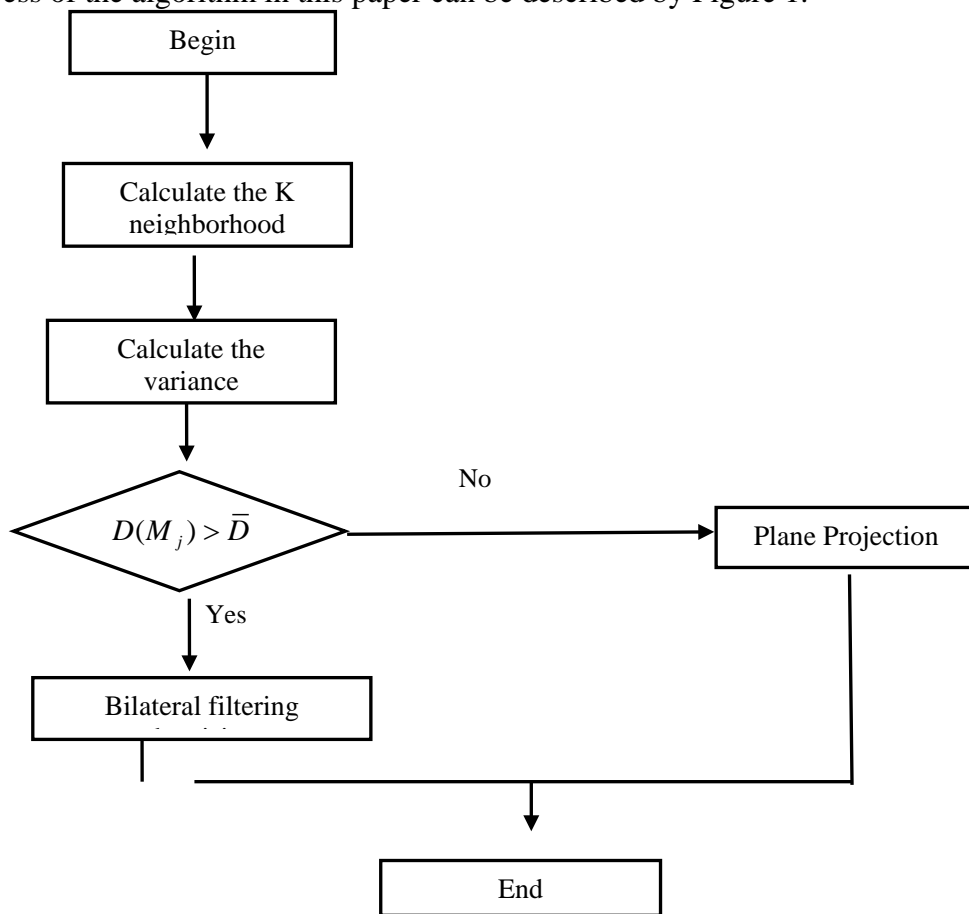
Fig1 A schematic diagram of the whole process of point cloud denoising

5. Experiments

This algorithm in this paper and the traditional bilateral filter denoising algorithm are used to deal with the work point cloud, street point cloud and building point cloud collected in actual production, in order to carry on the comparison experiment. The workpiece point cloud is collected by a handheld 3D laser scanner HandyScan. Street point clouds and building point clouds are collected by FARO 3D laser scanner. The result of the experiment can be see in Figure 2 ,Figure3 and Figure 4,and the processing time comparison can be see in Table 1. Through the experiment, we can see that compared

with the traditional bilateral filter denoising algorithm, the result of the algorithm in this paper not only keeps the geometric feature better, but also the point cloud after processing is more smooth. At the same time, the speed of this algorithm is much faster.
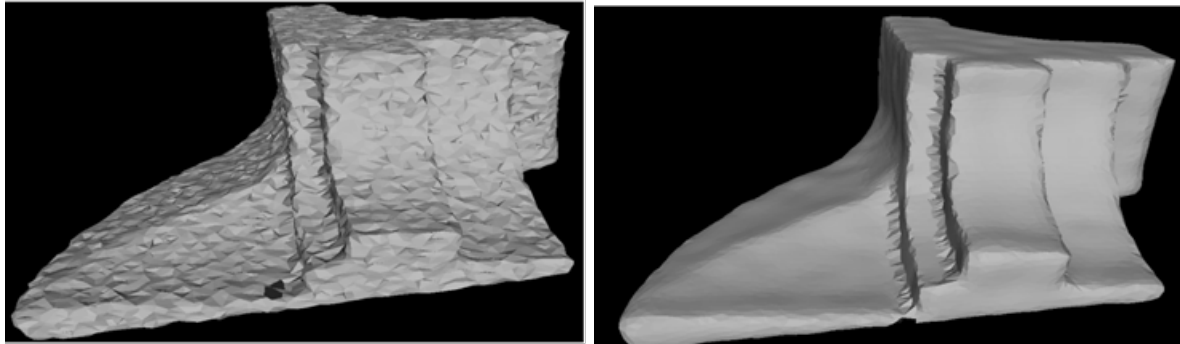
(a) Tranditional algorithm   (b) The algorithm in this paper
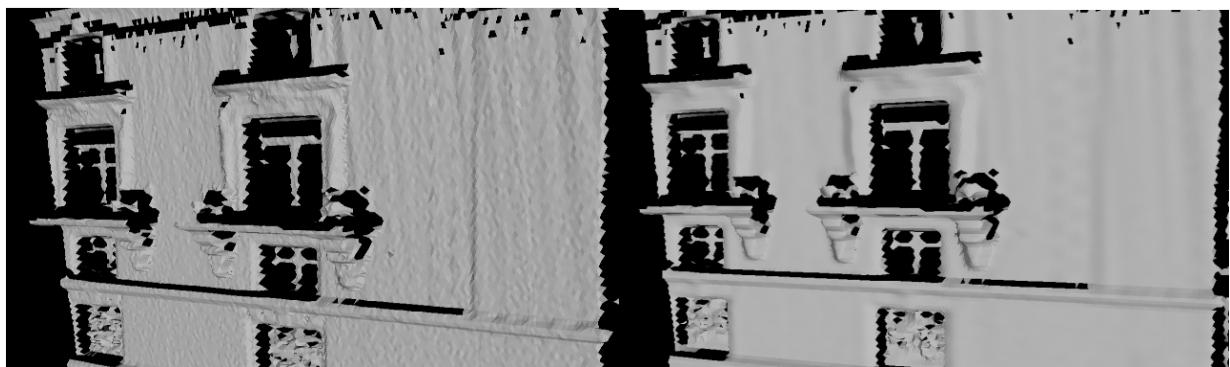
Fig2   Results of denoising a workpiece point cloud
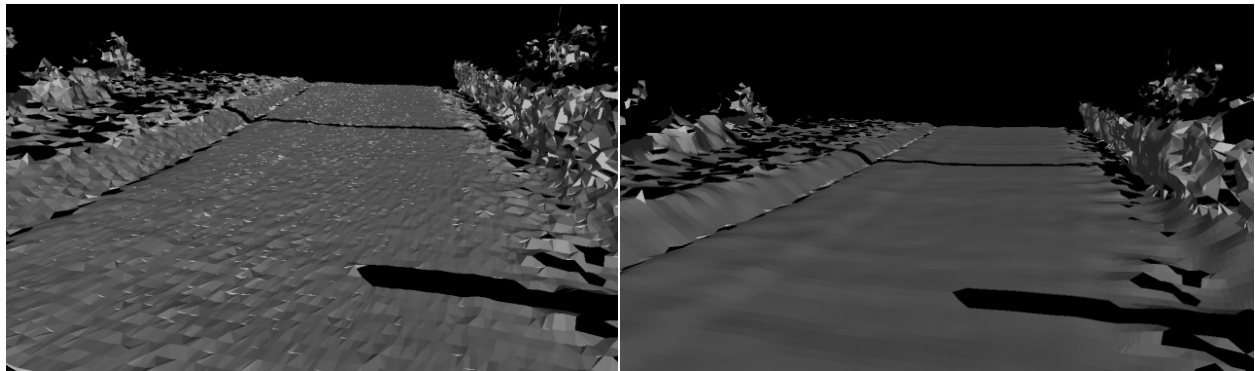
(a) Tranditional algorithm   (b) The algorithm in this paper

Fig3   Results of denoising a street point cloud

(a) Tranditional algorithm   (b) The algorithm in this paper

Fig4 Results of denoising a building point cloud

Table 1 Different algorithms for processing time comparison table

Name	Number	Tranditional algorithm	The algorithm in this paper
Workpiece	37124	56.58 s	31.38 s
Street	126796	1698.61 s	1016.98 s
Building	445385	6376.52 s	3612.69 s

6. Conclusion

In this paper, a new K neighborhood search algorithm based on space cell is proposed based on the traditional K neighborhood search algorithm, which changes the construction of the k-d tree and improves the search speed of the nearest neighbor (k) neighbor. Then, a point cloud data point classification method based on variance threshold judgment is proposed. Based on this, the traditional bilateral filtering algorithm is improved. Finally, the experiment shows that compared with the traditional bilateral filtering algorithm, the algorithm improves the defect of the bilateral filtering algorithm to a certain extent, and has a strong practicability.

References

[1] Wang X Z, Li Z K, Mai Y Q, et al. Robust denoising of unorganized point clouds[C]// Conference Anthology, IEEE. IEEE, 2014:1-3.

[2] Rosman G, Dubrovina A, Kimmel R. Patch-Collaborative Spectral Surface Denoising[J]. 2012.

[3] Xu S, Yang Z, Wu W. Algorithm of 3D reconstruction based on point cloud segmentation denoising[C]// International Conference on Information Science and Engineering. IEEE, 2011:3510-3513.

[4] Yang Z, Xiao D. A systemic point-cloud de-noising and smoothing method for 3D shape reuse[C]// International Conference on Control Automation Robotics & Vision. IEEE, 2013:1722-1727.

[5] Taubin G. A signal processing approach to fair surface design[C]// Conference on Computer Graphics and Interactive Techniques. ACM, 1995:351-358.

[6] Alexa M, Behr J, Cohen-Or D, et al. Point set surfaces[C]// Visualization, 2001. VIS '01. Proceedings. IEEE, 2009:21-28.

[7] Fleishman S, Drori I, Cohen-Or D. Bilateral mesh denoising[C]// Acm Siggraph. ACM, 2003:950-953.