

# Point Cloud Denoising based on Adaptive Wavelet Transformation

Zhou Baoxing

Department of Civil Engineering  
Shandong Jiaotong University  
Jinan, Shandong, China  
E-mail: zbxsdjtu@163.com

**Abstract**—The point cloud data obtained by 3D laser scanner are not only simple in structure, easy to operate, but also don't need store topological relationships between points. They can be used to express complex geometry and surface characteristics of irregular objects. However, in the process of obtaining the data, because of many factors, such as the human factor, the change of the environment or the defects of the equipment itself, the data obtained are contaminated by noise. Therefore, point cloud data denoising is an important post-processing step performed on potentially noisy data obtained from a 3D scanner. A new point cloud denoising method is proposed based on adaptive wavelet transformation, which includes three steps: namely, point cloud data decomposition, wavelet coefficients neighborhood adaptive division and wavelet coefficients inverse transformation. The method can not only effectively remove the noise, but also preserve sharp features and surface details. At last, the performance of the proposed method was illustrated with a validation experiment.

**Keywords**—Point Cloud; Denoising; Wavelet Transformation; Mean Square Error; Spatial Distribution Error

## I. INTRODUCTION

Three dimensional laser scanner can acquire a large number of point cloud data, which contain the geometrical information of the object. It has become one of the common methods for representing 3D objects. It can be used to reconstruct the digital model of the object. However, when the 3D object data model is obtained using scanner instrument, the device defect or human disturbance will make the scan data inevitably have gross error and noise data. The error in 3D point cloud data will affect the accuracy and effect of the model parameters. Therefore, the noise of point cloud model need to be removed [1]. At present, the noise elimination method of point cloud mainly including:

A. *Grid-based denoising*: this method requires a local topology configuration or a local parameterization in point cloud neighborhood [2-3]. The point cloud data, however, do not contain any topological information. If the scattered point data are directly used for reconstruction, triangulation or parameterization, they will take a great price [4-5].

B. *Direct de-noising algorithm*: the algorithm can be realized through multiple iterations based on point cloud data [6-7].

C. *Curve or surface approximation algorithms*: these algorithms cannot effectively distinguish the point cloud

data in the gross errors and outliers, resulting in over smoothing phenomenon [8].

All above de-noising techniques do not preserve sharp features and surface details, or require considerable time to obtain fine filtering results.

## II. THE BASIC PRINCIPLE OF 2D WAVELET TRANSFORM

Let  $f(x_1, x_2) \in L^2(R^2)$  represent a 2D signal, and  $x_1, x_2$  is respectively horizontal and vertical coordinates.  $\psi(x_1, x_2)$  is a 2D basic wavelet, the 2D continuous wavelet definition [9]:

Supposing  $\psi_{a;b_1,b_2}(x_1, x_2)$  is the scale and 2D displacement of  $\psi(x_1, x_2)$

$$\psi_{a;b_1,b_2}(x_1, x_2) = \frac{1}{a} \psi\left(\frac{x_1 - b_1}{a}, \frac{x_2 - b_2}{a}\right) \quad (1)$$

The 2D continuous wavelet transform is:

$$\begin{aligned} WT_f(a; b_1, b_2) &= \langle f(x_1, x_2), \psi_{a;b_1,b_2}(x_1, x_2) \rangle \\ &= \frac{1}{a} \iint f(x_1, x_2) \psi\left(\frac{x_1 - b_1}{a}, \frac{x_2 - b_2}{a}\right) dx_1 dx_2 \end{aligned} \quad (2)$$

In which, factor  $\frac{1}{a}$  is normalization factor, which can guarantee wavelet energy unchanged before and after stretching.

Inverse wavelet transform of 2D continuous wavelet transform:

$$\begin{aligned} f(x_1, x_2) &= \frac{1}{c_\psi} \int_0^{+\infty} \frac{da}{a^3} \iint WT_f(a; b_1, b_2) \psi\left(\frac{x_1 - b_1}{a}, \frac{x_2 - b_2}{a}\right) db_1 db_2 \\ c_\psi &= \frac{1}{4\pi^2} \iint \frac{|\psi(\omega_1, \omega_2)|^2}{|\omega_1^2 + \omega_2^2|} d\omega_1 d\omega_2 \end{aligned} \quad (3)$$

The parameters  $a, b$  is discretized:  $a = 2^j, b_1 = al$ , And  $b_2 = am$ . The discrete wavelet transform is expressed:

$$W_f(j, l, m) = 2^{-j} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x_1, x_2) \psi(2^{-j}x - l, 2^{-j}y - m) dx_1 dx_2 \quad (4)$$

The 2D wavelet function  $\psi(x_1, x_2)$  is a binary function which variables can be separated, its formula can be expressed as  $\psi(x_1, x_2) = \psi(x_1)\psi(x_2)$ .

### III. POINT CLOUD IMAGE DECOMPOSITION AND RECONSTRUCTION

#### A. Point Cloud Image Decomposition

For a point cloud image  $f_1(x_1, x_2)$ , when  $j = 0$ , this scale of point cloud image is  $2^j = 2^0 = 1$ . Since the  $j$  value increases one time, it will double the size of the scale, and the resolution is reduced to half of the original [10].

The point cloud image can be extended using 2D wavelet transform technique. At every level of transformation, each point cloud image is decomposed into four images, the decomposed image size is a quarter of the original image. Each image is formed by the inner product between a wavelet base and original image, then by resampling in the x-direction at 2 times interval. When  $j = 1$ , its expression is:

$$\begin{cases} f_2^0(n_1, n_2) = [f_1(x_1, x_2), \psi(x_1 - 2n_1, x_2 - 2n_2)] \\ f_2^1(n_1, n_2) = [f_1(x_1, x_2), \psi^1(x_1 - 2n_1, x_2 - 2n_2)] \\ f_2^2(n_1, n_2) = [f_1(x_1, x_2), \psi^2(x_1 - 2n_1, x_2 - 2n_2)] \\ f_2^3(n_1, n_2) = [f_1(x_1, x_2), \psi^3(x_1 - 2n_1, x_2 - 2n_2)] \end{cases} \quad (5)$$

For subsequent level  $j > 1$ ,  $f_{2^j}^0(x_1, x_2)$  can be decomposed into smaller four images on the  $2^{j+1}$  scale. The expression of convolution form:

$$\begin{cases} f_{2^{j+1}}^0(n_1, n_2) = \{[f_{2^j}^0(x_1, x_2) \times \psi(-x_1, -x_2)](2n_1, 2n_2)\} \\ f_{2^{j+1}}^1(n_1, n_2) = \{[f_{2^j}^0(x_1, x_2) \times \psi^1(-x_1, -x_2)](2n_1, 2n_2)\} \\ f_{2^{j+1}}^2(n_1, n_2) = \{[f_{2^j}^0(x_1, x_2) \times \psi^2(-x_1, -x_2)](2n_1, 2n_2)\} \\ f_{2^{j+1}}^3(n_1, n_2) = \{[f_{2^j}^0(x_1, x_2) \times \psi^3(-x_1, -x_2)](2n_1, 2n_2)\} \end{cases} \quad (6)$$

At every level four identical interval sampling filter should be carried out. Because the wavelet function and scaling function is separable, each convolution can be decomposed into 1D convolution of rows and columns of  $f_{2^j}^0(x_1, x_2)$ .

The calculation process is shown in Fig.1,  $cA$  represents the decomposed low-frequency coefficients,  $cD^{(b)}$ ,  $cD^{(v)}$ , and  $cD^{(d)}$  represent high frequency coefficients. LD and HD represent low-pass and high-pass decomposition filters.

In each scale,  $f_{2^j}^0(x_1, x_2)$  contains low frequency information of previous stage, and  $f_{2^j}^1(x_1, x_2)$ ,  $f_{2^j}^2(x_1, x_2)$  and  $f_{2^j}^3(x_1, x_2)$  contain horizontal, vertical and diagonal direction edge information.

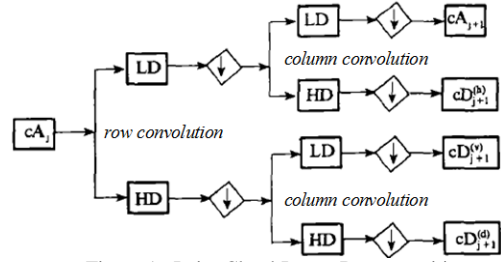


Figure 1. Point Cloud Image Decomposition

#### B. Inverse Transformation

Inverse transformation process is similar to the process of transformation, and the calculation process is shown in Fig.2.

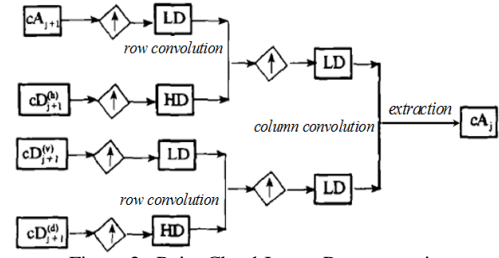


Figure 2. Point Cloud Image Reconstruction

### IV. POINT CLOUD DATA DENOISING BASED ON NEIGHSHRINK ALGORITHM

The algorithm can determine whether the wavelet coefficients of the center of the window are zero or shrink based on the square sum of all wavelet coefficients within the neighborhood window. It can also estimate parameters of each wavelet coefficient based on their respective neighborhood and can obtain a threshold value for each sub-band coefficient. Due to the size and shape of the neighborhood is fixed and cannot be changed adaptively according to the nature of the point cloud image itself. The effect of this denoising method is limited. In this paper, the method of wavelet coefficients neighborhood division is improved.

#### A. Wavelet Coefficient Neighborhood Adaptive Division

Let  $k$  is a certain even, line number of wavelet coefficients is  $i$  ( $i = 1, \dots, m$ ), and column number is  $j$  ( $j = 1, \dots, n$ ). According to the following principles, the neighborhood boundaries of each wavelet coefficient can be determined based on rank order. The wavelet coefficients neighborhood are divided as follows:

- When  $i = 1$ , the row neighborhood of the wavelet coefficients is the first and second row. When  $j = 1$ , the column neighborhood of the wavelet coefficients is the first and second column.
- When  $1 < i \leq \frac{k}{2}$ , the row neighborhood of the wavelet coefficients is between first and  $2i - 1$  row. When  $1 < j \leq \frac{k}{2}$ , the column neighborhood of the wavelet coefficients is between the first and the  $2j - 1$  column.

- When  $\frac{k}{2} < i < m - \frac{k}{2}$ , the row neighborhood of the wavelet coefficients is between  $i - \frac{k}{2}$  and  $i + \frac{k}{2}$  row. When  $\frac{k}{2} < j < n - \frac{k}{2}$ , the column neighborhood of the wavelet coefficients is between  $j - \frac{k}{2}$  and  $j + \frac{k}{2}$  column.
- When  $m - \frac{k}{2} \leq i < m$ , the row neighborhood of the wavelet coefficients is between  $2i - m$  and  $m$  row. When  $n - \frac{k}{2} \leq j < n$ , the column neighborhood of the wavelet coefficients is between  $2j - n$  and  $n$  column.
- When  $i = m$ , the row neighborhood of the wavelet coefficients is between  $m - 1$  and  $m$  row. When  $j = n$ , the column neighborhood of the wavelet coefficients is between  $n - 1$  and  $n$  column.

#### B. Point Cloud Denoising Using Adaptive Neighshrink

- The 2-D discrete wavelet transform is carried on the point cloud image with noise, and the coefficients of each sub-band each layer are obtained respectively;
- In the high frequency sub bands, neighborhood division is used to the coefficients using above method, and the coefficients are processed by the neighborhood threshold:

Treating threshold wavelet coefficients as  $d_{m,n}$ , using formula (3.32) shrink,  $d'_{m,n}$  is factor after shrinkage:

$$d'_{m,n} = d_{m,n} \alpha_{m,n} \quad (7)$$

The shrinkage factor is defined as  $\alpha_{m,n}$

$$\alpha_{m,n} = \begin{cases} 1 - \left( \frac{\lambda}{S_{m,n}} \right)^2 & \text{if } S_{m,n} > \lambda \\ 0 & \text{else} \end{cases} \quad (8)$$

In which,  $S_{m,n} = \sum_{(m,n) \in W_{m,n}} d_{m,n}^2$ ,  $\lambda$  is threshold.

- The low frequency and the high frequency sub bands are reconstructed by the 2-D discrete wavelet transform, then the point cloud image after de-noising can be obtained.

#### V. EFFECT EVALUATION OF POINT CLOUD IMAGE DE-NOISING

In order to comprehensively evaluate de-noising effect of the point cloud image, we use mean square error and spatial distribution error of point cloud image to assess the quality of point cloud noise cancellation.

##### A. Mean Square Error

The template is designed so that author affiliations are not repeated each time for multiple authors of the same affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization). This template was designed for two affiliations.

$$MSE = \frac{1}{mn} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} \|I(x, y) - K(x, y)\|^2 \quad (9)$$

##### B. Spatial Distribution Error

$$SDE = I(x, y) - K(x, y) \quad (10)$$

In which,  $I$  represents the original point cloud data without noise.  $K$  represents the point cloud data after noise elimination. The point cloud image noise spatial distribution is shown as Fig.3 and Fig.4.

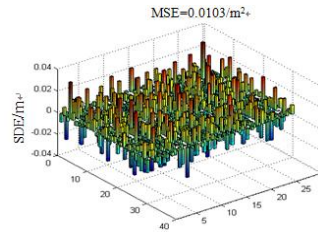


Figure 3. Spatial Distribution Error

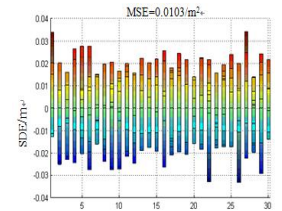


Figure 4. Side View of Spatial Distribution Error

#### VI. EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify effectiveness of the algorithm, as shown in Fig.5-Fig.7, a rectangular point cloud data in building is selected as the study area, the point cloud data error lies between +2mm and -2mm, its mean square error is  $0.3 \text{ mm}^2$ , and the error distribution is shown in Fig.8.

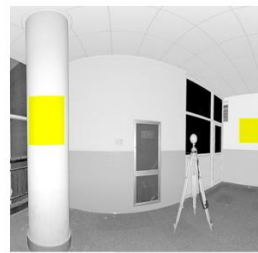


Figure 5. Scanned Image

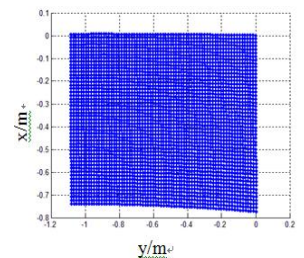


Figure 6. Original Point Cloud (Front View)

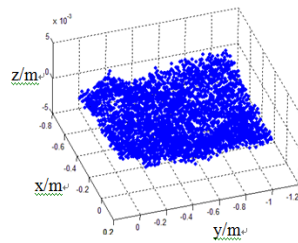


Figure 7. Original Point Cloud (Side View)

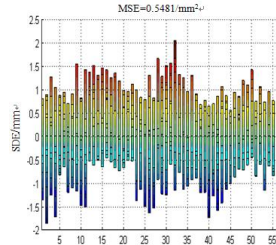


Figure 8. Point Cloud Error Distribution

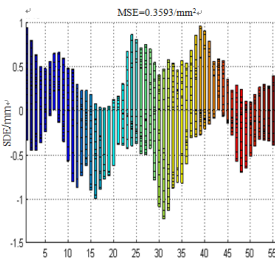


Figure 9. Soft Threshold Denoising

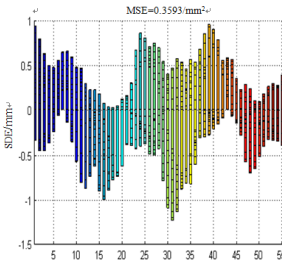


Figure 10. Hard Threshold Denoising

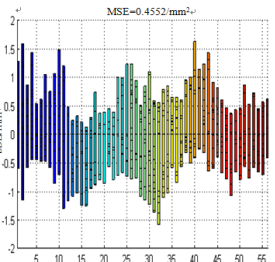


Figure 11. Semi-Soft Threshold Denoising

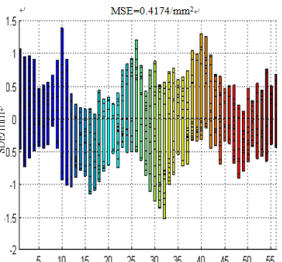


Figure 12. Semi-Soft and Filter Threshold Denoising

Mean Denoising

Denoising

Filter Threshold Denoising

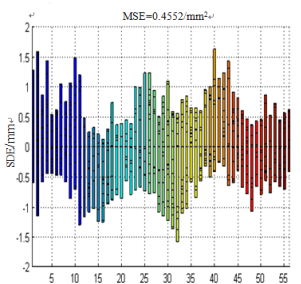


Figure 13. Semi-Soft and Exponential Decay Threshold Denoising

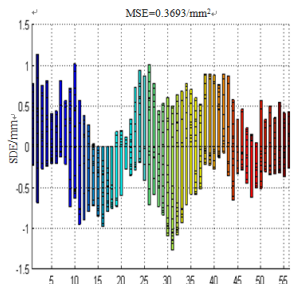


Figure 14. Generalized Wavelet Threshold Denoising

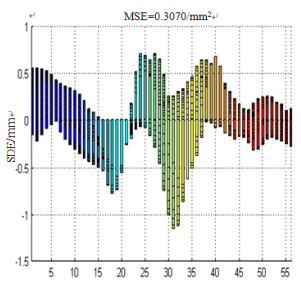


Figure 15. NeighShrink Algorithm Denoising

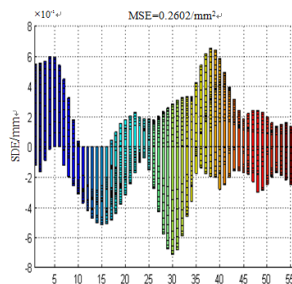


Figure 16. Adaptive NeighShrink Algorithm Denoising

Known from the above Figures:

*A. Soft threshold and hard threshold:* after point cloud data smoothed, the error distribution lies between  $\pm 1\text{mm}$  and  $-1\text{mm}$ , the mean square error is  $0.13\text{mm}^2$ .

*B. Semi-soft threshold:* Using this method and its various modifications of the algorithm for point cloud data smoothing, its error distribution lies between  $\pm 1.5\text{mm}$ , their mean square error of about  $0.2\text{mm}^2$ .

*C. Generalized wavelet threshold:* Using this method, the point cloud data smoothing, its error distribution is between  $\pm 1\text{mm}$ , their mean square error of about  $0.14\text{mm}^2$ .

*D. Adaptive Wavelet Threshold:* Using this method, the point cloud data smoothing, and its error distribution lies between  $\pm 0.6\text{mm}$ , their mean square error of about  $0.07\text{mm}^2$ .

Therefore, all kinds of threshold denoising algorithms can be used to reduce the noise in the point cloud data. However, different threshold denoising methods have different smooth degrees. Wherein the effect of semi-soft threshold is worst, the proposed adaptive neighborhood denoising method works the best.

## VII. CONCLUSIONS

Point cloud data denoising principle and existent key problem based on wavelet transform technology were discussed in this paper. The method of adaptive wavelet coefficients neighborhood in threshold was improved and applied to original point cloud data. According to experiments, the residual error of point cloud data lies between  $\pm 0.6\text{mm}$ , its mean square error is about  $0.07\text{mm}^2$ . Therefore, the technology applied to the point cloud denoising has good prospects.

## ACKNOWLEDGMENT

This work was financially supported by A Project of Shandong Province Higher Educational Science and Technology Program (J14LG07), National Natural Science Fund project (41174002).

## REFERENCES

- [1] J. Vollmer, R. Mencl, and H. Mueller, "Improved laplacian smoothing of noisy surface meshes". Computer Graphics Forum, 1999, 18(3):131-138.
- [2] N. Salman, M. Yvinec and Q. M  rigot, " Feature preserving mesh generation from 3d point clouds". In: Computer graphics forum, vol. 29; 2010. p. 1623-32.
- [3] T. Guillemot, A. Almansa and T. Boubekur, " Non-local point set surfaces". In: 2011 international conference on 3D imaging, modeling, processing, visualization and transmission, vol. 0; 2012. p. 324-31.
- [4] J. Cao, S. Wushour, X. Yao, N. Li, J. Liang, X. Liang, "Sharp feature extraction in point clouds". IET Image Process 2010; 6 (October (7)): 863-9, <http://dx.doi.org/10.1049/iet-ipr.2011.0361>.
- [5] M K. Park, S J. Lee and I Y. Jang, "Feature-aware filtering for point-set surface denoising". Computers & Graphics, 2013, 37 (6): 589-595.
- [6] A. Wetzler, G. Rosman and R. Kimmel, "Patch-space Beltrami denoising of 3D point clouds", Electrical & Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of. IEEE, 2012: 1-5.
- [7] G. Rosman, A. Dubrovina, and R. Kimmel. "Patch-collaborative spectral surface denoising". Technical Report CIS-2012-03, Technion, Department of Computer Science, 2012.
- [8] N. Sapidis, G. Farin, "Automatic Fairing Algorithm for B-spline Curves". CAD, 1990, 22 (2):121-129.
- [9] Al-Ataby, W. Al-Nuaimy, C R. Brett, et al. "Automatic detection and classification of weld flaws in TOFD data using wavelet

transform and support vector machines”. *Insight-Non-Destructive Testing and Condition Monitoring*, 2010, 52 (11): 597-602.

[10] K. Rai R, J. Asnani and T R. Sontakke, “Review of shrinkage techniques for image denoising”. *Int J Comput Appl*, 2012, 42: 13-6.