

Generating Orientated Point Clouds from RGB-D Sensor

W.B. Shan, Y. Yao, J.J. Liu, M. Guo
Rapid Manufacturing Engineering Center

Y. Yao
Shanghai Key Laboratory of Manufacturing Automation and
Robotics
Shanghai University
Shanghai, China

Abstract-In order to reconstruct a surface by using the data from RGB-D sensor, this paper presents a viewpoint based method for estimating the orientation of point clouds. Firstly, a PCA method is used to compute the normal vector of each point. In this process, the orientation of the normal is determined by the viewpoint. Secondly, the point clouds generated from each frame are combined to an overall model with directional information. Finally, the quality of the model is further improved by a local smooth method. The experimental results show that the method can compute accurate normal vectors on the fly. Compared to current methods, it can estimate the sign of the normal automatically.

Keywords-RGB-D sensor; point clouds; normal estimation; 3D reconstruction

I. INTRODUCTION

3D reconstruction has been widely used in entertainment, industry, archaeology, digital simulation and virtual reality, etc.[1]. From a technical point of view, 3D scanner can be divided into the laser scanner, the structured light scanner, and the binocular vision scanner. The first two kinds are popularly used in the industry. However, they are expensive and generate a large amount of corresponding data and thus are time-consuming. These shortcomings are reasons why they cannot be widely used. Establishing cheap and fast reconstruction system is one of the hottest topics in the field of 3D reconstruction. In 2010, Microsoft Kinect launched low-cost RGB-D sensors, which provides an inexpensive hardware platform for 3D reconstruction using the depth camera[2].

At present, many researchers are focused on reconstruction object from RGB-D data [3], and the results are mesh models with textures. In the reconstruction, computing directional point clouds is still an important step, because numerous surface reconstruction algorithms rely on the normals estimation. The point clouds with directions can adopt some widely used methods to generate high quality meshes. Normal estimation algorithms can be grouped into three different categories [4]: Robust statistics method, Delaunay/Voronoi method and Surface fitting method. The robust statistics method can effectively handle noise, outliers and sharp characteristics in point clouds. However, it needs to classify neighboring points. It is a time-consuming work. Therefore, it is unsuitable for reconstruction and it is difficult to realize real-time surface reconstruction. Delaunay/Voronoi method is

sensitive to noise, and it cannot be applied on the point clouds with the noise. Dey expanded the application of Voronoi cell pole to Delaunay ball [5], however, this method will cause incomplete calculation. Local surface fitting method has some resistance for noise [6], which is widely used in normal estimation for point clouds. This method is fast, therefore, it is fit for those applications that need real-time interactions. However, noise in the point cloud will also affects the accuracy of this method [7].

Many current algorithms only calculate the normal in a straight line. However, they do not determine the orientation of the normal, which leads to an orientation ambiguities problem. So Hoppe etc.[8] put forward the method of minimum spanning tree. The method cannot be applied to a large-scale point cloud. Radu [9] proposed a method that making use of the center of the point cloud model to orient all the normals, which is very effective for simple model. However, the complex model who with local sharp or thin-walled structure will produce a lot of mistakes.

In order to estimate normals for each point and reduce the difficulties of the above works, we propose a composite method that adopting PCA and viewpoint to compute normals for point cloud model based on RGB-D sensor. We have the follows contributions in this paper: (1) Our method is based on local covariance analysis method to compute normals for each point cloud and making the scanner position as the viewpoint to determine the orientation of normals. (2) The correction process is based on weighted averages of neighborhood points' normals, which can improve the accuracy of normal estimation. Therefore, we can generate normals automatically during the scan process.

II. POINT CLOUD NORMALS GENERATION

The camera and environment would generate much noise in the point clouds, therefore, this paper selects robust normal estimation method based on local covariance analysis. From single point cloud to the orientated point cloud model (see fig. 1.), first, computing normals for point cloud which is the i th point cloud data from RGB-D sensor. Then, registration and superposition generate the orientated point cloud model. Finally, after correction normals, we can output directed point clouds model.

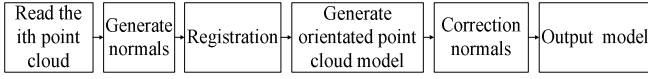


FIGURE 1. ORIENTATED POINT CLOUD MODEL GENERATION PROCESS.

Now we will give the details about how to generate normals with our method. When input point cloud $P=\{p_i\}, i \in I, p_i \in R^3$, making a KD tree for the point cloud, any point $p_i \in P$, searching its neighboring points. These neighboring points can be created a tangent plane with the least square fitting. As is shown in fig. 2., the normal of the tangent plane is as the normal of the corresponding point. The method of solving the tangent plane is creating a 3x3 covariance matrix C . These eigenvectors of covariance matrix C are making up the principal component direction of the local spatial point cloud. The eigenvector is as the approximate vector of the point, which is corresponding to the smallest eigenvalue, for eigenvalues measure the change of direction in each of independent components [7][10].

$$C = \frac{1}{k} \sum_{i=1}^k (p_i - \bar{p}) \bullet (p_i - \bar{p}) \quad (1)$$

$$C \vec{v}_j = \lambda_j \vec{v}_j(0) \quad (2)$$

Where k is the number of neighborhood points, \bar{p} is the center of these neighboring points, λ_j is the j th characteristic value of the covariance matrix, \vec{v}_j is the j th feature vector.

To determine the final direction of the normal, we put forward the location of RGB-D sensor as the viewpoint v_p , its coordinates is $(0,0,0)$. As is shown in Fig. 2, every normal \vec{n}_i should meet $\vec{n}_i \bullet (v_p - p_i) > 0$. At the same time, normals of point clouds have been generated, and they are all in the same direction toward to the outside of the model.

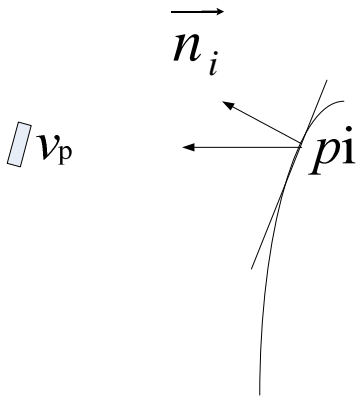


FIGURE 2. NORMALS CALCULATION.

After generating normals for point cloud, we adopt the method based on a linear least square optimized (LLS) ICP registration [8]. In the registration step, normals can keep the original features, namely, for each point p_i $(x_i, y_i, z_i, n_i^x, n_i^y, n_i^z)$, Suppose T is the is the transposed matrix, after registration $(x_i, y_i, z_i) \bullet T \rightarrow (x_i^o, y_i^o, z_i^o)$. And

the three-dimension point p_i becomes p_i' $(x_i^o, y_i^o, z_i^o, n_i^x, n_i^y, n_i^z)$. At last, we get orientated point clouds model.

A. Normals Correction

Normals calculated by fitting point clouds surface method may be affected by noise, some normals may can't correctly reflect the characteristics of the model [7]. To improve precision of the point cloud model, we can evaluate each normal by weighted average of normals which are neighbouring. Point cloud normals would become more accurate and smooth. Suppose P_i is a point in point clouds, (n_i^x, n_i^y, n_i^z) is its normal. We can compute the weighted average of normals N_i by formula (3)

$$N_i = (N_i^x, N_i^y, N_i^z), \{N_i^x = \sum_{i=1}^k \alpha_i \bullet n_i^x, N_i^y = \sum_{i=1}^k \alpha_i \bullet n_i^y, N_i^z = \sum_{i=1}^k \alpha_i \bullet n_i^z\} \quad (3)$$

In the formula, α_i is the weight of the i th neighboring point's normal, N_i^x, N_i^y, N_i^z are respectively weighted average of N_i^x, y, z component. k is the number of points which are in the scope of a certain radius r . α_i is obtained by formula (4):

$$\alpha_i = 1 - \frac{l_i^2}{l_{max}^2} \quad (4)$$

In the formula, l_i is the distance of the neighboring point to point, l_{max} is the max distance from point to its neighbors. According to the weighted average, evaluating each point normal, the formula (5) as follows:

$$\cos \theta = \frac{n_i \bullet N_i}{|n_i| \bullet |N_i|} \quad (5)$$

The $\cos \theta$ is confident of a point's normal. Then we set a threshold to assess point cloud normals. The threshold is set by your own. If the confidence is greater than threshold, the normal of the point is credible. Otherwise it is incredible, the weighted average will take as the original normal.

Our method consists two steps: the single piece of point cloud's normal calculation and the global point cloud model normal correction. According to the method steps, we have given the complexity of the algorithm. Assume the number of points is n , original dimension is D , the number of neighboring points is k . First of all, the complexity of establishment of KD tree is $O(3*n*\log n)$. The complexity of the query process is $O(n)$. The complexity of the query neighborhood points is $O(k)$. The complexity of PCA is $O(nD)+O(D^3)$, the complexity of normal correction is $O(n)$. So the complexity of our algorithm is $O(n)+O(3*n*\log n)$. To test our method, we take experiments, exam the effects of the reconstruct mesh model.

III. EXPERIMENT

A. Normal Calculation and Surface Reconstruction

The experiment of 3D reconstruction is taken on a PC with Windows 7 system, i3 processor and 2G memory. To show the result of reconstruction based on point cloud normals, we take multiple views on object to register. Scanning object once every 12° , after achieving point cloud, the system would send message to turntable until accomplish the whole scanning process automatically.

As is shown in fig.3., After generating orientated point cloud model, we adopt the Poisson reconstruction method [11] to generate the mesh surface. In order to make the model more authentic, in the surface reconstruction, we attach the corresponding color information to the mesh model.

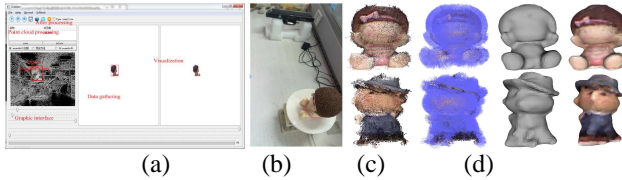


FIGURE 3. OUR EXPERIMENT AND RESULT. (A) OUR SOFTWARE; (B) WORK SCENE; (C) POINT CLOUD MODEL AND ORIENTATED POINT CLOUD MODEL; (D) SURFACE MODEL.

B. The Experiment Result Assessment

In order to verify the effect of the generating normals method, we compare with the method which is usually used PCA to compute the global point clouds model normals [9]. As is shown in fig.4.(a), mesh models have some additional meshes. The reason for this phenomenon is the orientated method using the center as the pointview, it would make some normals direction inside, see fig. 4.(c). Comparing with them, our method (see the fig.4.(d)) has made normals toward to one side and the corresponding models are closed (see the fig.4.(b)).

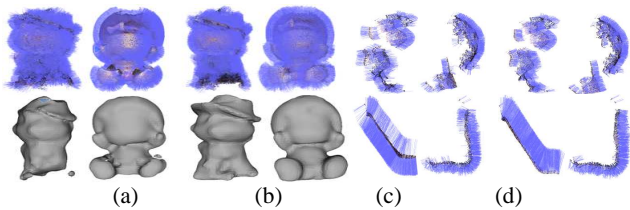


FIGURE 4. NORMALS ON MESH.

IV. CONCLUSION AND PROSPECT

In this work, a novel method of generating orientated point clouds from RGB-D data is proposed. This method can avoid the error of generating wrong normal vectors for some points. Furthermore, it will make the surface reconstruction more stable. The future study will focus on increasing the curvature of points cloud model to adaptively adjust the size of fitting surface and improve precision of normal vectors.

REFERENCE

- [1] Weise, T, Thomas Wismer & Bastain Leibe, Luc Van Gool. Online loop closure for real-time interactive 3D scanning, *Computer Vision and Image Understanding*, 115(5): 635-648, 2011.
- [2] Cruz, Leandro, Lucio, Djalma & Velho, Luiz. Kinect and RGBD Images: Challenges and Applications, 25th SIBGRAPI Conference on Graphics, 2012.
- [3] Jing Tong. Scanning and Reconstruction of 3D Objects and Human Bodies Using Depth Cameras, Zhejiang University, 2012.
- [4] Bao Li, Zhiquang CHEN & Gang DANG, Shiyao JIN. Survey on normal estimation for 3D point clouds, *COMPUTER ENGINEERING AND APPLICATIONS*. 2010.
- [5] Dey TK, Goswami S. Provable surface reconstruction from noisy samples, *Computational Geometry*, 35(1-2):124-141, 2006.

- [6] Yutao Wang, His-Yung Feng, Félix-étienne Delorme, Serafettin Engin. An adaptive normal estimation method for scanned point clouds with sharp features, *Computer-Aided Design*, (45):1333-1348, 2013.
- [7] Su Zhixun, Li Zhiyang & Wang Xiaochao. Denoising of Point-Sampled Model Based on Normal Mollification and Median Filtering, *Journal of Computer Aided Design and Computer Graphics*, 2010.
- [8] Low K L. Linear least-squares optimization for point-to-plane icp surface registration, Technical Report TR04-004 February, 2004.
- [9] Rusu. Semantic 3D Object Maps for Everyday Manipulation In Human Living Environments, TECHNISCHE UNIVERSITÄT MÜNCHEN, 2009.
- [10] Rusu, R.B. & Cousins, S. 3D is here: Point Cloud Library (PCL), Robotics Automation (ICRA), 2011.
- [11] Michael Kazhdan, Matthew Bolitho & Hugues Hoppe. Poisson Surface Reconstruction, *Eurographics Symposium on Geometry Processing*, 2006.