Indoor rigid sphere recognition based on 3D point cloud data

Jifang Duan, Kishan Lachhani, Hadi Baghsiahi, Eero Willman, David R. Selviah
Department of Electronic and Electrical Engineering
University College London (UCL)
Torrington Place
WC1E 7JE London
Jifang.duan.13@ucl.ac.uk, d.selviah@ucl.ac.uk

Abstract

This paper presents a method for recognising spherical shapes in 3D point cloud XYZ coordinate data obtained by scanning an indoor environment using a LIDAR scanner. Firstly, bilateral smoothing is performed to smooth the surfaces consisting of points. Then, the surface curvature and surface roughness of each point in the scan are extracted by analysing the point cloud data. Finally, a three layer multilayer perceptron neural network trained by the Levenberg-Marquardt algorithm is used to automatically distinguish points belonging to spheres from all the other points making use of extracted features. A novel feedback technique is applied in which the neural network is used several times on the recognised data. This method can be applied to automate 3D scan alignment.

Keywords: Point cloud data, Object recognition, Neural networks, 3D laser scanning, LIDAR.

1 Introduction

The development of 3D scanning and camera projection technology make it possible for people to have better access to large amounts of accurate 3D point cloud data. The point cloud data, recorded by 3D LIDAR scanners, is used in a variety of fields, including architecture, medical science, surveying and mapping. It is widely applied in generating 3D models, undertaking metrology inspection and performing medical imaging.

In general, point cloud data itself is not directly usable in most 3D applications as it occupies a lot of memory and storage, requiring further analysis and processing. For example, point clouds can be converted to mesh models or CAD models for further use. In the field of architecture, the Building Information Modelling (BIM) concept was introduced in recent years. BIM describes the whole life cycle of a project and gives very detailed information of everything related to the building, including cost, construction, project and facility management. Currently, BIM is mainly used at the start of construction projects where laser scanning may not be that useful since 3D models of buildings would normally already exist. However, with the expansion of the BIM industry, existing buildings will require BIM as well, and that will offer a market for laser scanning and modelling automation. At present, the conversion from point cloud data obtained from a scanned building to BIM is typically performed by manual means, which is time consuming and labour intensive. In order to facilitate the procedure, S. Oesau et al. [1] proposed a method using feature sensitive primitive extraction and graph-cut for automatic reconstruction of permanent structures, such as walls, floors and ceilings. X. Xiong et al. [2] succeeded in identifying and modelling the main visible structural components of an indoor environment. Apart from large planar areas, windows and doorways are also able to be identified by applying this method. The question is whether other shapes such as spheres can also be automatically identified.

This paper focuses on indoor sphere recognition. Spheres have the characteristic of rotational symmetry, which can be regarded as a distinctive feature for alignment between different scans. As

a basic geometric shape, spheres of small or large radius appear everywhere in buildings. Being able to recognise spherical objects can be seen as a start for the recognition of more complex 3D objects. The Hough transform is a powerful tool in shape analysis. O. Ogundana et al. [3] extended the strategy for detecting circles in 2D images to detecting spheres in 3D point clouds. RANSAC is also a useful tool for 3D sphere extraction [4]. This paper proposes a new method for sphere recognition. Our algorithm can be divided into four main steps: 1) bilateral smoothing, in which the point clouds are smoothed; 2) calculation of the surface curvature and the surface roughness; 3) multilayer perceptron neural networks are trained using supervised learning by the Levenberg-Marquardt algorithm, and used to distinguish points belonging to spheres from other points; 4) low-density filtering, in which low-density points are removed from the point cloud.

2 Methodology

2.1 Bilateral smoothing

The 3D point cloud data is obtained by Faro Focus 3D LIDAR. Due to the tolerances of the scanner itself, the 3D data, inevitably, contains range noise. In order to improve the accuracy and reliability to point cloud computation, it is important to de-noise and smooth the point cloud. The bilateral filter, introduced by Tomasi and Manduchi [5], is a non-linear, edge-preserving and noise-reducing filter, which was first used to filter images. It has a simple and intuitive formulation and can be adapted to point cloud data easily and successfully [6-7].

In 3D point cloud data smoothing, let, $\mathbf{p}$, be the 3D coordinates of a point in the scan. After the application of the filter, updating, $\mathbf{p}$, as is given in Eq. (1):

$$\mathbf{p}' = \mathbf{p} + b \cdot \mathbf{n} \tag{1}$$

Where, $\mathbf{n}$, is the surface normal of the point, b , is the bilateral smoothing factor defined as follows:

$$b = \frac{\sum_{\mathbf{p_i} \in R} W_c\left(\left\|\mathbf{p} - \mathbf{p_i}\right\|\right) \cdot W_s\left(\left|\left\langle \mathbf{n}, \mathbf{n_i} \right\rangle\right|\right) \cdot \left\langle \mathbf{n}, \mathbf{p} - \mathbf{p_i} \right\rangle}{\sum_{\mathbf{p_i} \in R} W_c\left(\left\|\mathbf{p} - \mathbf{p_i}\right\|\right) \cdot W_s\left(\left|\left\langle \mathbf{n}, \mathbf{n_i} \right\rangle\right|\right)} \tag{2}$$

Where, R, is the spherical neighbourhood of $\mathbf{p}$, $\mathbf{p_i} \in R$ apart from the point $\mathbf{p}$, $\left\|\mathbf{p} - \mathbf{p_i}\right\|$ is the distance between point $\mathbf{p}$ and $\mathbf{p_i}$, $\left\langle \mathbf{n}, \mathbf{n_i} \right\rangle$ is the angle between vector $\mathbf{n}$ and $\mathbf{n_i}$. The closeness smoothing filter is a standard Gaussian filter with parameter σ_c: $W_c(x) = e^{-x^2/2\sigma_c^2}$. A feature-preserving weight function with parameter σ_s is defined as: $W_s(x) = e^{-x^2/2\sigma_s^2}$.

2.2 Surface curvature and surface roughness

The surface curvature (SC) of each point of the scan is computed from eigenvalues of a local 3 by 3 covariance matrix [8] of a certain region of interest around the point. This region is usually taken to be a sphere. The radius of the sphere is required to be chosen wisely. It should be much smaller than the size of the scanned spheres while relatively larger than the surface thickness of the scanned objects. The surface roughness can be obtained by comparing surface normals of neighbouring points. Again this is calculated over a spherical region of interest whose radius is set to be the same as that for the SC calculation. The surface normal of each point is calculated using principle component analysis and covariance analysis [9] with an Octree-based 3D-grid method [10] for efficient neighbouring point searching.

Theoretically, the average SC of a sphere should be invariant despite the any changes in the distance between the parts of the sphere and the scanner and the variable density of points, and the SC of each point on one sphere should be the same. However, in practice this is not the case. The point cloud data in the experiments is from a single scan, so objects are partially scanned and will have point cloud boundary edges. The discrepancy in surface curvature calculation is mainly due to the effect of edges and the spatially varying range noise.

2.3 Neural networks trained by the Levenberg-Marquardt algorithm

Artificial neural networks (ANN) can detect complex nonlinear relationships between dependent variables and separate and distinguish different classes of pattern. Several algorithms can be used for the training procedure of an ANN. The Levenberg-Marquardt (LM) algorithm [11] is selected for this research. The LM algorithm is a combination of the steepest descent method and the Gauss-Newton algorithm. It inherits the stability of the steepest descent method and the speed advantage of the Gauss-Newton algorithm.

In this research, two features, surface curvature and surface roughness, are extracted from point cloud data for each point. They are used for the inputs of a 3 layer multilayer perceptron ANN for each point. Four spheres and two non-sphere backgrounds selected from the scan are used for training this ANN. After being well trained but not overtrained, the ANN model is applied to each point in a sample area in order to identify points belonging to spheres. Repeated application of the ANN helps in obtaining better discrimination results. Each time after the ANN model is applied, the points identified by the ANN as not belonging to a spherical surface are removed from the scene. This reduction of number of points results in the modification of the curvature and the roughness values for points remaining in the scene when they are recalculated before putting back into the same ANN. As a result each time the ANN acts as a filter removing non-sphere points and the SC and SR are recalculated and the ANN is applied again. After applying the ANN for several times, there still remain some points that do not belong to spheres, so a low-density filter is applied, which is described in section 2.3, to remove these points.

2.3 Low-density filtering

For each point, the number of its neighbouring points within a certain spherical volume of radius, r, around that point is calculated. The distance, d, between each point and the LIDAR is also computed. Points, whose neighboring points count is smaller than a threshold number, th, are removed from the point cloud. The threshold, th, is calculated as follows:

$$th = \frac{\sigma r^2}{2d^2} \tag{3}$$

where, σ, is the number of points per unit area at 1 metre away from the scanner.

2.4 Flowchart of methodology

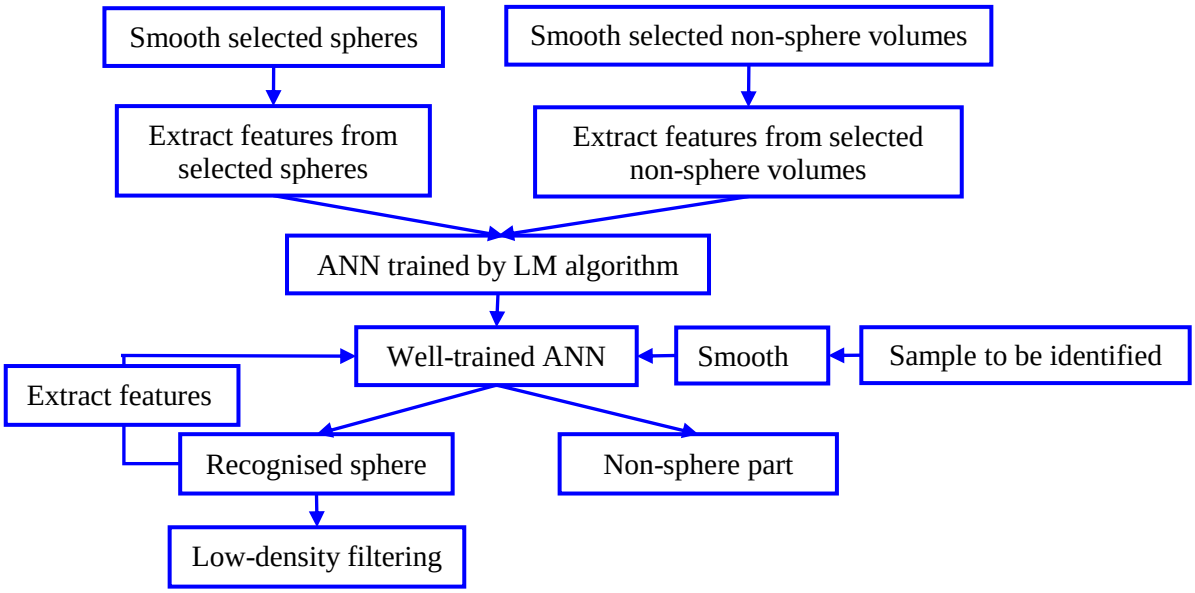
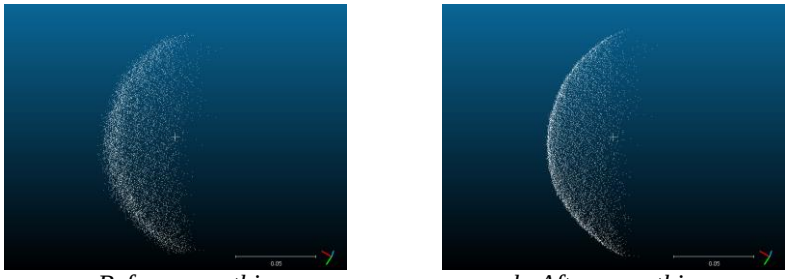


Figure 1: Flowchart of methodology

3 Experimental results

In the experiment, eight spheres were randomly placed in an indoor environment. The scene was then scanned with Faro Focus 3D LIDAR. Figure 2 shows a side view of a partially captured sphere. After applying bilateral smoothing, the surface of this sphere appears smoother and better defined. A quantitative comparison of the calculated surface thicknesses for spherical objects and the scene floor before and after smoothing is presented in Table 1. Filtering improves the accuracy in computing the curvature and the roughness measures for each point, which are illustrated in Figure 3 and Figure 5. Figure 4 shows the calculated curvature value distribution on the surface of a partially captured sphere. Near the edges of the partial sphere, the curvature is relatively low, with a value close to that of a flat surface. Conversely, points on the sphere, located further away from the edges possess relatively higher curvature. Figure 6 plots both the curvature and the roughness of a sphere after smoothing shows that there exists a relationship between the two. These two different properties of the sphere surface have a similar tendency.

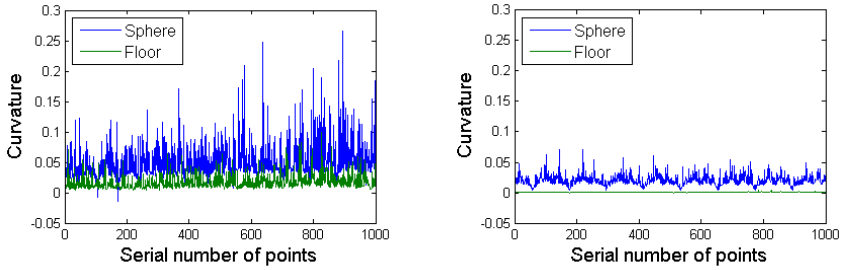


a. Before smoothing *b. After smoothing*

Figure 2: Comparison of a scanned sphere before/after smoothing

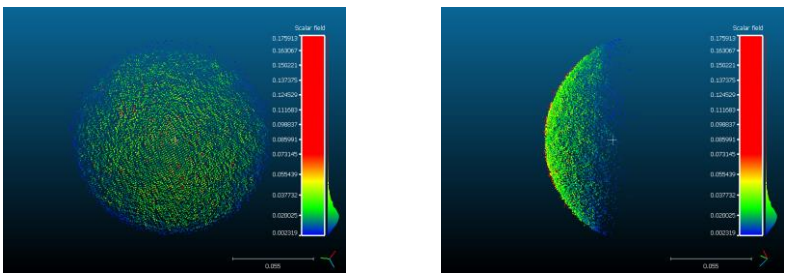
Surface thickness	Sphere (2 m away from the scanner)	Sphere (4.5 m away from the scanner)	Floor (2 m away from the scanner)	Floor (4.5 m away from the scanner)
Before smoothing (mm)	20	15	12	3
After smoothing (mm)	8	6	2	1

Table 1: Comparison of the surface thicknesses of scanned objects before and after smoothing



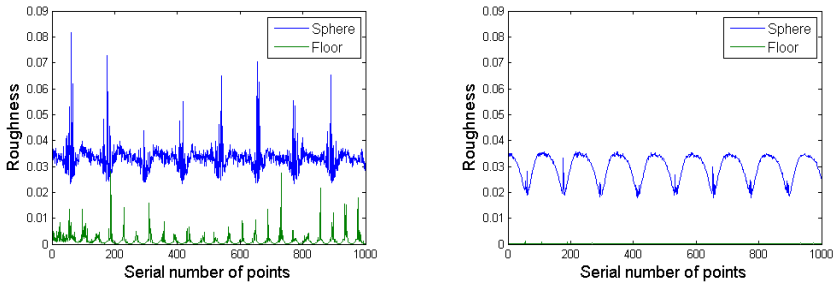
a. Before smoothing *b. After smoothing*

Figure 3: Comparison of surface curvature value (a) before and (b) after smoothing. The curvature is calculated for each point on a selected sphere and for each point on a selected area of floor



a. Front view *b. Side view*

Figure 4: Calculated surface curvature distribution over a sphere



a. Before smoothing *b. After smoothing*

Figure 5: Comparison of surface roughness before and after smoothing. The roughness is calculated for each point on the selected area of the sphere and for each point on the selected area of the floor

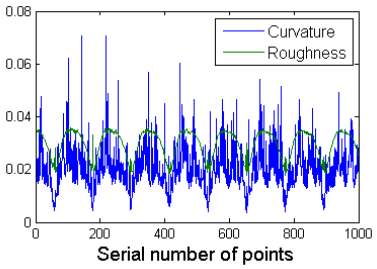


Figure 6: Surface curvature and roughness for points of a selected area of the sphere after smoothing

The distance between each of these eight scanned spheres and the scanner was in the range from 2 m to 6 m. As is shown in Figure 7 and Figure 8, the implementation of bilateral smoothing reduces the variability of average curvature and average roughness for spheres placed at different distances with respect to the LIDAR scanner. For detecting objects, it is desirable that the calculated curvature and roughness are independent of their distances to the scanner.

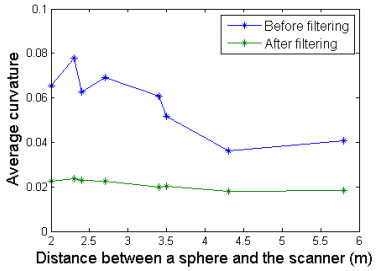


Figure 7: Comparison of the average curvature before/after smoothing between spheres

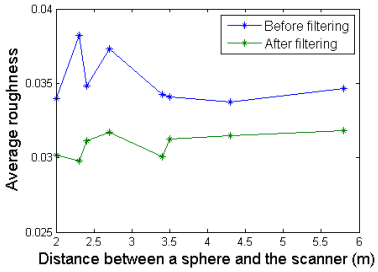
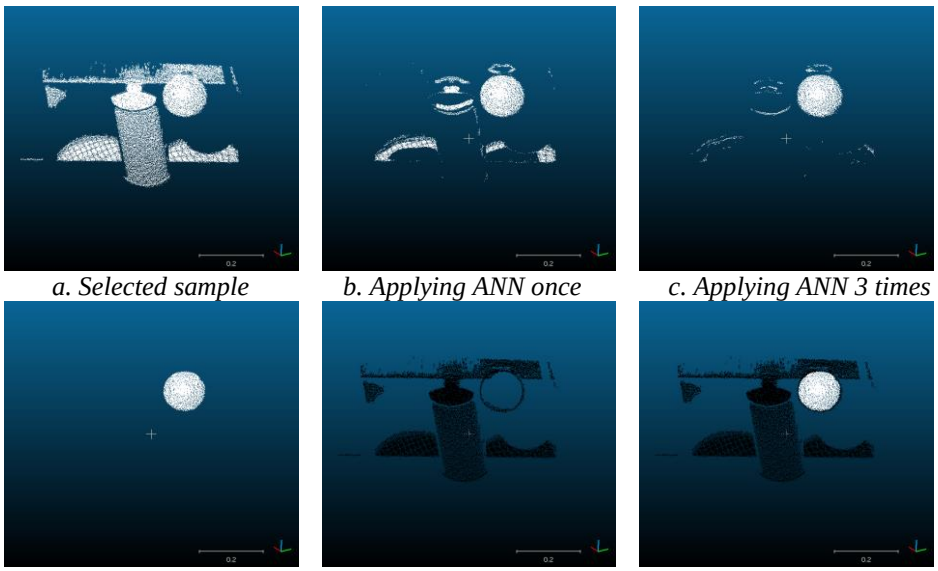


Figure 8: Comparison of the average roughness before/after smoothing between spheres

Four of these scanned spheres and two non-sphere volumes were selected to train the ANN model. Parameters for the ANN chosen for this experiment are presented in Table 2. Figure 9 (a) shows the sample area selected from a scan, which includes a sphere, a lamp and a flat surface. This data was not used in the training of the ANN model. Figure 9 (a) - (c) show the effect of repeatedly applying the ANN to the data. After this, low-density filtering described in section 2.3 is applied to the remaining points. The result of this is shown in Figure 9 (d).

Number of neurons for input/hidden/output layer	2; 10; 2
Transfer function	Tan-sigmoid, Log-sigmoid
Percentage of data for training/validation/testing	70%; 15%; 15%

Table 2: Parameters for ANN



a. Selected sample *b. Applying ANN once* *c. Applying ANN 3 times*

d. Recognised sphere *e. Points other than the recognised sphere* *f. Overview*

Figure 9: Recognition process. (a) Input data. (b)-(c), result of applying ANN repeatedly to input data. (d) remaining points after ANN and density filtering of input data. (e) and (f) Points identified to lie on non-spherical surfaces (black) and Points classified to lie on surface of sphere (white).

4 Conclusions

In this paper, we presented an artificial neural network pattern recognition approach to detect points in a point cloud that define the surface of a spherical object. Our method is able to correctly distinguish points belonging to spheres from other points in the environment which may also include other curved surfaces present. A novel feedback technique is applied in which the neural network is used several times on the input data.

References

[1] O. Sven, L. Florent and A. Pierre (2014). Indoor scene reconstruction using feature sensitive primitive extraction and graph-cut. *ISPRS Journal of Photogrammetry and Remote Sensing*, 90: 68-82.

[2] X. Xuehan, A. Antonio and A. Burcu (2013). Automatic creation of semantically rich 3D building models from laser scanner data. *Automation in Construction*, 31: 325-337.

[3] O. Olatokunbo, C. Russell and L. Richard (2007). Fast Hough transform for automated detection of spheres in three-dimensional point clouds. *Optical Engineering*, 46 (5): 051002

[4] R. Schnabel, R. Wahl and R. Klein (2007). Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum*, 26 (2): 214-226

[5] C. Tomasi and R. Manduchi (1998). Bilateral filtering for gray and colour images. *Sixth International Conference on Computer Vision*, 839-846.

[6] S. Fleishman, I. Drori, and D. Cohen-Or (2003). Bilateral mesh denoising. *ACM Transactions on graphics*, 22: 950-953.

[7] D. Xiaoyuan, J. Xiaofeng, H. Chuangang and W. Yumei (2010). Bilateral filtering denoising algorithm for point-cloud model. *Computer Applications and Software*, 20 (7): 245-264.

[8] M. Pauly, M. Gross and LP. Kobbelt (2002). Efficient simplification of point-sampled surfaces. *IEEE Visualization 2002*, 163-170.

[9] M. Pauly (2003).Point primitives for interactive modeling and processing of 3D geometry. For the degree of Doctor for Sciences. Federal Institute of Technology of Zurich.

[10] H. Woo, E. Kang and SY. Wang (2002). A new segmentation method for point cloud data. *International Journal of Machine Tools & Manufacture*, 42: 167-178.

[11] J. Shawash and D. Selviah (2013). Real-Time nonlinear parameter estimation using the Levenberg-Marquardt algorithm on field programmable gate arrays. *IEEE Transactions on Industrial Electronics*, 60 (1): 170-176.