Proceedings Paper:
Shui, W., Liu, J., Maddock, S. et al. (2016) Automatic planar shape segmentation from indoor point clouds. In: VRCAI '16 Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry. VRCAI 2016, December 03 - 04, 2016 , Zhuhai, China. ACM, pp. 363-372. ISBN: 978-1-4503-4692-4 .

https://doi.org/10.1145/3013971.3014008

Automatic planar shape segmentation from indoor point clouds

Wuyang Shui[1,*], Jin Liu[2], Pu Ren[1], Steve Maddock[3], Mingquan Zhou[1]

1. College of Information Science and Technology, Beijing Normal University, China
2. Institute of Forensic Science, Ministry of Public Security, China
3. Department of Computer Science, University of Sheffield, UK

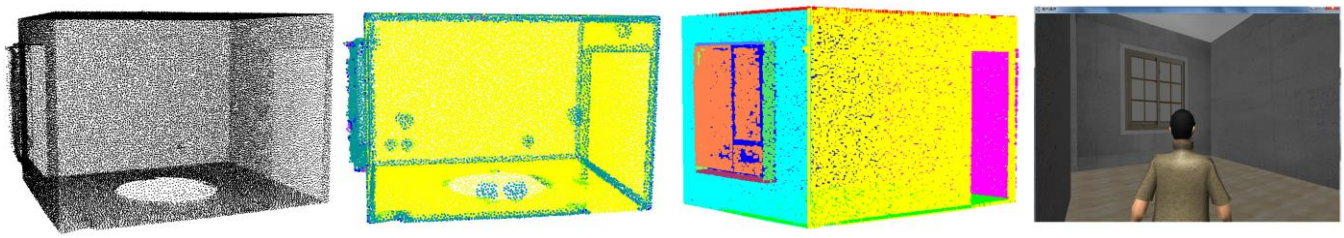

Figure 1: *Planar shapes segmentation and virtual roaming. From the sampled scanned point clouds, the planar shapes of indoor scene (such as floor, walls, ceilings, etc.) have been finely segmentation. Subsequently, the indoor scene is represented by only fewer number of planar patches and the digital model with texture is imported to Unity 3D to enable members of the public virtually experience the space.*

Abstract

The use of a terrestrial laser scanner (TLS) has become a popular technique for the acquisition of 3D scenes in architecture and design. Surface reconstruction is used to generate a digital model from the acquired point clouds. However, the model often consists of excessive data, limiting real-time user experiences that make use of the model. In this study, we present a coarse to fine planar shape segmentation method for indoor point clouds, which results in the digital model of an indoor scene being represented by a small number of planar patches. First, the Gaussian map and region growing techniques are used to coarsely segment the planar shape from sampled point clouds. Then, the best-fit-plane is calculated by random sample consensus (RANSAC), avoiding the negative impact of outliers. Finally, the refinement of planar shape is produced by projecting point clouds onto the corresponding best-fit-plane. Our method has been demonstrated to be robust towards noise and outliers in the scanned point clouds and overcomes the limitations of over- and under-segmentation. We have tested our system and algorithms on real datasets and experiments show the reliability of the proposed method against existing region-growing methods.

Keywords: indoor scene modeling, scanned point clouds, Gaussian map, planer shape segmentation

Concepts: •Computing methodologies → Point-based models; Shape analysis;

1 Introduction

High quality 3D models of real-world indoor scenes have an important role in forensic scene reconstruction [Buck et al. 2013], indoor scene understanding [Nan et al. 2012], interior design [Shao et al. 2012], robot navigation [Mozos et al. 2012] and building information models (BIM) [Xiong et al. 2013]. The acquisition of such scenes has been supported by the low-cost of 3D acquisition sensors and the popularization of commercial RGB-D sensors [Silberman and Fergus 2011;Silberman et al. 2012; Zou et al. 2012]. Mobile scanning systems can be used to produce a complete single point cloud of a large indoor space with a complex geometrical structure by using simultaneous localization and mapping (SLAM) [Adán et al. 2015; Izadi et al. 2011; Whelan et al. 2015]. However, inaccurate registration leads to low-quality point clouds fusion, especially in dark and sparse textured areas. Terrestrial laser scanners (TLS) can be used to tackle this problem and to acquire point clouds of indoor scenes. In comparison to an RGB-D sensor, the benefit of TLS is that high precision and complete point clouds can be acquired using only a few scan positions.

Since a typical indoor scene is primarily composed of planar shape elements, such as walls, floors, and ceilings, it can be represented by a limited number of planar patches, instead of over-detailed triangle meshes. The benefits of this representation are as follows [Mura et al. 2014; Ochmann et al. 2016]: (a) Low storage and less RAM is required for processing; (b) A parameterized scene model can be generated permitting easy scale and rotation of the whole scene or parts of it; (c) Wall thickness and building information models can be generated. Planar shape segmentation from point clouds was the fundamental step in the Mura et al. and Ochmann et al. studies. This step is also the focus of our work.

Most previous methods use normal, curvature and other geometric descriptors to segment planar shapes. However, these techniques negatively impacted on the segmentation in the case of noisy and smooth regions. We use a TLS to acquire point clouds and our method then segments planar shapes. In addition, we use a graph structure to represent an indoor scene, which can be used in a real-time virtual experience. In summary, the contributions of this paper

are:

- We introduce a novel coarse to fine segmentation algorithm to finely segment planar shape from point clouds, avoiding over- and under-segmentation.
- We use a geometric representation for indoor scenes, relying on a graph structure and a limited number of planar patches.

2 Related work

We focus our review of related work on the two prominent approaches proposed for planar shape extraction in previous studies: region growing and model fitting.

2.1 Region growing

Region growing is a popular feature-based method to segment a region of interest (ROI) from a point cloud. Once a seed point and growing conditions are given (for example the normal direction, curvature, roughness, etc.), neighboring points are merged only if the growing conditions are satisfied. This growing step is repeated until all the points have been considered. Ease of implementation and fast speed mean that region growing has been intensively used. The drawbacks of this method are that it is sensitive to the choice of seed and the computation accuracy of growing conditions such as normal and curvature calculations.

Rabbani et al. used local surface normal and the region growing algorithm to group points belonging to smooth surfaces [Rabbani et al. 2006]. Xiong et al. discretized point clouds using a uniform 3D grid data structure and planar patches were extracted by connecting neighboring points with similar surface normal [Xiong et al. 2013]. Zhang et al. employed mean-shift curvature to distinguish features and then region growing was used to cluster [Zhang et al. 2008]. Deschaud et al. estimated a better normal by a weighted fitting plane and extracted the planar shape by a voxel-based growing algorithm, instead of growing with k nearest neighbours [Deschaud and Goulette 2010]. Ma et al. selected a point with the lowest curvature as a seed and adopted region growing to add the k neighbours of the seed point to the current segmentation region [MA et al. 2013]. To tackle the problem of segmentation related to the choice of seed, fuzzy clustering [Biosca and Lerma 2008] and mean-shift [Liu and Xiong 2008; Wang et al. 2013] have been used.

A Gauss map can also be used to detect features from point clouds, relying on the normal of every point [Weber et al. 2010]. Wang et al. adopted a Gaussian map and mean-shift to recognise planes, cylinders, cones and spheres [Wang et al. 2013]. Liu et al. proposed a non-parametric cell mean-shift method to extract clusters from point clouds based on a Gaussian map and an orientation analysis was used to identify hyperbolic and elliptical regions [Liu and Xiong 2008].

2.2 Model fitting

The RANSAC and Hough transform are two prominent model fitting methods. RANSAC is an effective model fitting method from inliers and robustly against noise, outliers and missing data [Fischler and Bolles 1981]. Schnabel et al. analyzed the complexity of RANSAC and detected a variety of different types of shapes, including plane, sphere, cylinder, cone and torus [Schnabel et al. 2007]. Ochmann et al. represented wall candidates as triples (t_w, n_w, d_w), considered the thickness of a wall, and detected planes from point clouds after RANSAC implementation [Ochmann et al. 2016]. To avoid spurious surface extraction, Awwad et al. proposed an improved Seq-NV-RANSAC approach by checking the normal between the existed points and hypothesized RANSAC plane [Awwad et al. 2010].

The Hough transformation algorithm is another type of model fitting method. Early work used the 2D Hough transform to detect geometric primitives [VC 1962]. The 3D Hough transform is an extension to detect planar patches. Vosselman and Dijkman used the 3D Hough transformation to extract planar patches from irregular point clouds and the problem of over-segmentation was tackled by merging small planar patches [Vosselman and Dijkman 2001]. Leng et al. used the Hough transformation to extract planar clusters from rock-mass point clouds, taking a cluster to the voting distribution in the parameter space [Leng et al. 2016]. Oesau et al. proposed an automatic reconstruction method of planar shape. Vertical walls were detected through clustering in a Hough transform space, and both inside and outside segmentation were performed by graph-cut [Oesau et al. 2014]. It has, however, been demonstrated that the RANSAC method generally has higher performance and is much faster than the 3D Hough transform method [Tarsha-Kurdi et al. 2007].

3 Our Approach

The pipeline that we use for segmenting planar shapes from TLS point clouds is shown in Figure 2. First, the TLS is positioned so as to acquire the point clouds (Figure 2a). Next is point clouds sampling (Figure 2b). Next, we separated the window and room from the complete point cloud of the indoor scene (Figure 2c) and our segmentation algorithm was carried out on each component (Figure 2d). The result is a representation of the indoor scene by a group of planar patches (Figure 2e). The final digital model with texture is imported to Unity 3D to enable members of the public to virtually experience the space (Figure 2f). Within the segmentation process, we calculate normal and curvature for every point using the weighted principal component analysis and the planar regions are coarsely extracted by region growing on the Gaussian map. Planar shape regions are extracted by projecting a point cloud on a best-fit-plane. An important aspect of our approach was to overcome the problem of over- and under-segmentation.

3.1 Coarse segmentation

3.1.1 Sampling

Since a TLS point cloud is often composed of tens of millions of points, it is sampled, often uniformly, to accelerate subsequent data processing. Poisson-disk sampling satisfies three properties: the distance between any two disk centers should be larger than the sampling radius; the union of the disks should cover the entire sampling domain; each point in the domain has a probability that is proportional to the sizing at this point to receive a sampling point [Guo et al. 2015]. We used the Poisson-disk sampling algorithm to reduce the number of points in the point cloud.

3.1.2 Normal and curvature computation

Reliable normal and curvature estimation at every point in a point cloud is challenging, and subsequently has an impact on, for example, surface reconstruction [Kazhdan et al. 2006] and point cloud segmentation [Crosilla et al. 2009]. Moving least square [Pauly et al. 2003a], quadric surface fitting [Douros and Buxton 2002], the fitting plane and improved algorithms[Hoppe 1992] have been used to estimate normal and curvature.

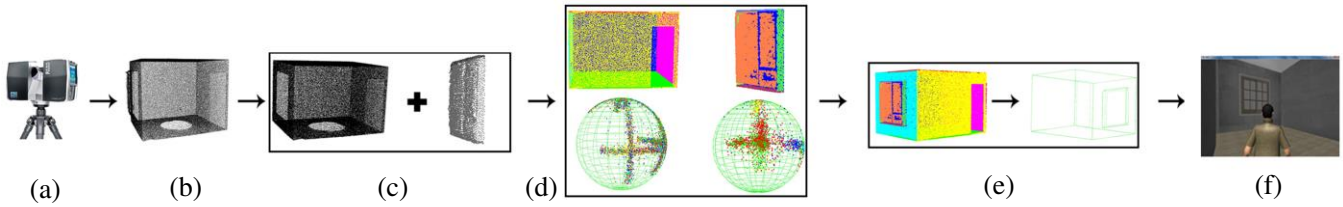

Figure 2: *Overview of our approach. (a) Terrestrial laser scanner; (b) Sampled point clouds. A hole occurs on the floor, where the TLS was placed for 3D acquisition. (c)Window and room were split from the sampled point clouds. (d)Planar shape regions segmentation by our proposed algorithm. Various color depicted various plane. (e) Planar patches and corresponding boundary. (f) The digital model with texture was virtually experience using Unity 3D.*

Consider the neighbour points of every point p_i are denoted as $Q=\{q_1,q_2,,...,q_m\}, q_i=(x_i,y_i,z_i)\in\mathbb{R}^3$ and the centroid of these neighbour points is denoted as $\bar{q}$, then the normal n_i can be optimized by the following equation:

$$E=\arg\min\sum_{i=1}^{m}(q_i-\bar{q})\cdot \mathrm{n}_i \tag{1}$$

The least squares method is used to solve this equation. The eigenvector corresponding to the smallest eigenvalue of symmetric positive covariance matrix $\Sigma=\frac{1}{m}\sum_{i=1}^{m}(q_i-\bar{q})(q_i-\bar{q})^{\mathrm{T}}$ describes the normal n_i . Following [Pauly et al. 2002], we use eigenvalues to describe the surface variation and the curvature at p_i is defined as:

$$\lambda=\frac{\lambda_3}{\lambda_1+\lambda_2+\lambda_3} \tag{2}$$

where λ_i denoted the eigenvalues of the corresponding covariance matrix and $\lambda_1\geq\lambda_2\geq\lambda_3$.

The result is sensitive to the choice of neighbour points, since neighbour points can belong to various surfaces around boundary regions and corners. To tackle the problem of the choice of neighbour, *k* nearest neighbours (KNN) [Hoppe 1992; Nurunnabi et al. 2015], fixed radius neighbours [Awwad et al. 2010; Mitra and Nguyen 2003; Xiong et al. 2013] and the weighted adaptive method [Deschaud and Goulette 2010; Pauly et al. 2003; Wang et al. 2013] are common approaches to use. Considering that the TLS acquires point clouds with equal density h_r , we used the fixed radius method and defined a weighted fitting plane to estimate normal and curvature. The general idea of our method is that the closest point impacts highly on the estimation.

The normal n_i can be optimized by the following equation:

$$E=\arg\min\sum_{i=1}^{m}\varpi_i\cdot(q_i-\overline{q_\varpi})\cdot \mathrm{n}_i \tag{3}$$

The weighted covariance matrix of neighbour points is defined as:

$$\Sigma=\sum_{i=1}^{k}\varpi_i\,(q_i-\overline{q_\varpi})(q_i-\overline{q_\varpi})^{\mathrm{T}} \tag{4}$$

where $\overline{q_\varpi}$ is the weighted centroid of neighbour points and ϖ_i is the weighted coefficient of every point.

The bandwidth is defined to decide the weighted coefficient of every point. For every point p_i , the weighted coefficient ϖ_i is defined as:

$$\varpi_i=\exp(-\|q_i-p_i\|^2/h_r^{\,2}) \tag{5}$$

where $h_r=k/r$, k is the number of neighbour points in the range of radius *r,* and $r=\lambda\cdot dist$. Here λ denotes the coefficient and $dist$ is the average distance between neighbor points.

Figure 3 shows the curvature estimation of a cube using various radii. The higher curvature occurs around the boundary regions, with the highest curvature located at the corners, whereas lowest curvature is located on the planar surface. The choice of neighbour points impacts on the curvature computation.

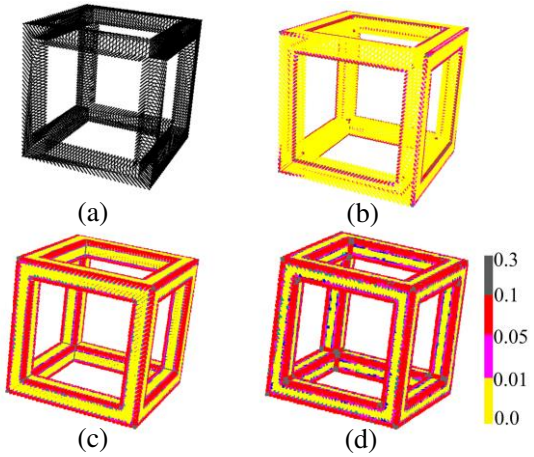

Figure 3: *Curvature estimation using various radius. (a) Cube point clouds ;(b) Radius is* $2\cdot dist$ *; (c) Radius is* $3\cdot dist$ *; (d) Radius is* $4\cdot dist$

3.1.3 Region growing

Let $p_i=(x_i,y_i,z_i)\in\mathbb{R}^3$ be the original coordinates of every point, each with a corresponding normal represented as n_i . The discrete Gaussian map of every point p_i can be defined as the mapping of T onto the unit sphere $S=\{(x,y,z)\in\mathrm{R}^3\mid x^2+y^2+z^2=1\}$ centered at $\bar{p}$. Therefore, the coordinate of point p'_i using a Gaussian map is:

$$p'_i=\bar{p}+\frac{n_i}{\|n_i\|} \tag{6}$$

The p'_i mainly rely on the centroid and corresponding normal of every point. In the ideal situation, point clouds in the Descartes coordinate system located on a plane map to a point (the black point in Figure 4a) after the Gaussian map transformation, because the normal of every point has an equal orientation. However, the TLS point clouds contain noise and, as a result, the normal of every point

on a plane doesn't have equal orientation. The mapped points cover a small region, as shown by the black points in Figure 4b.

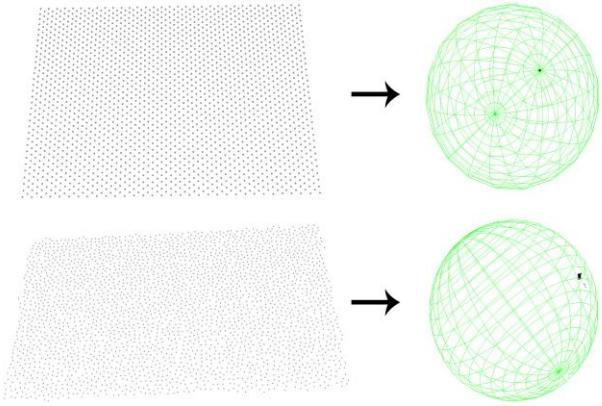

Figure 4: Gaussian *Map. (a) An ideal planar point cloud maps to a point on the* Gaussian *map; (b) Point cloud with noise maps to an area on the* Gaussian *map.*

The purpose of planar shape segmentation for indoor scene is to partition the point cloud into various planar clusters. Region growing is a well-known segmentation algorithm, the principle of which is to start with a seed region and to grow it by neighborhood when the neighbors satisfy some conditions. In this study, we used the mapped points on the Gaussian map, instead of the original point cloud. The region growing condition is defined as:

$$\begin{cases} \left\| p'_{\mathrm{i}} - p'_{j} \right\| < \varepsilon \\ curvature_{j} \approx 0 \end{cases} \tag{7}$$

where p'_{i} and p'_{j} are the current point and neighbor point on Gaussian map, ε is the threshold and $curvature_{j}$ is the curvature of the *j* index point using equation 2.

The general idea of our method is to merge neighbor points with a similar normal. The procedure is as follows:

Step 1: Compute normal and curvature of every point p_{i} and compute the p'_{i} by Gaussian map transformation.

Step 2: A point on the planar patch, i.e. whose curvature is approximately zero, is chosen as a start seed.

Step 3: The region growing strategy is greedily performed on the Gaussian map such that one neighbour point p'_{j} is added to a planar patch and labelled when it satisfies equation 7.

Step 4: If any neighbour point doesn't satisfy the region growing condition, a new planar patch is added. Then, another point with lowest curvature is chosen as a seed for the new planar patch and step 3 is repeated. The process continues until no points remain.

Step 5: According to the label of every point, planar patches on the Gaussian map and original point clouds are respectively visualized by various colours.

3.2 Fine segmentation

Because of smooth regions at wall boundaries, sharp features on corners and noise around the room windows, it is difficult to produce an exact segmentation. Refinement is thus required, avoiding over- and under-segmentation (Figure 5).

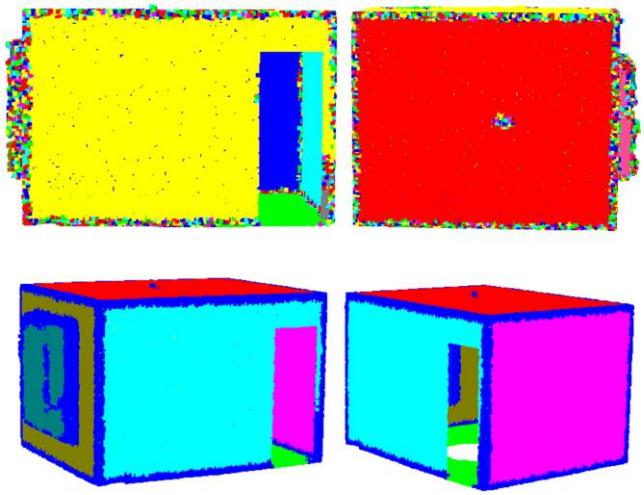

Figure 5: *Over-segmentation (top figure) and under-segmentation (bottom figure) after coarse segmentation.*

3.2.1 Fitting plane by RANSAC

Given the plane equation $z = ax + by + c$ and every point of the coarse segmentation represented by $s_{\mathrm{i}} = (x_{\mathrm{i}}, y_{\mathrm{i}}, z_{\mathrm{i}}) \in \mathbb{R}^3$, a fitting plane can be optimized using:

$$S = \min \sum_{i=0}^{n-1} \left(ax_i + by_i + c - z_i \right)^2 \tag{8}$$

The least squares method (LS) can be used to calculate the coefficients (a,b,c). However, we found that outliers, i.e. points not belonging to the plane, negatively impacted on the plane fitting. To address this problem, we used the RANSAC algorithm to fit the plane, even with a high number of outliers.

RANSAC includes two iterative stages: hypothesis and test. In the hypothesis stage, we randomly selected three points to compute plane coefficients by LS. In the test stage, every point was taken as the input of the estimated plane and the distance between point and plane was calculated. The point was taken as an inlier only when this distance was less than a threshold. The most probable inliers were chosen after several iterations and further used to compute the plane coefficient by LS. Figure 6 shows an example of calculating a best-fit-plane using RANSAC. The coarse segmentation (Figure 6a) was taken as the input of RANSAC. In comparison to LS (Figure 6c), we acquired better fitting results using RANSAC (Figure 6d). The plane fitting procedure using RANSAC is as follows:

Step 1: Randomly select three points with lowest curvature and estimate plane coefficients;

Step 2: Calculate the distance between point and the plane, and the point is taken as inlier when distance is less than threshold.

Step 3: Compute the number of inliers. If the number is greater than threshold, go to Step 4.

Step 4: Re-compute a best-fit-plane using all the inliers.

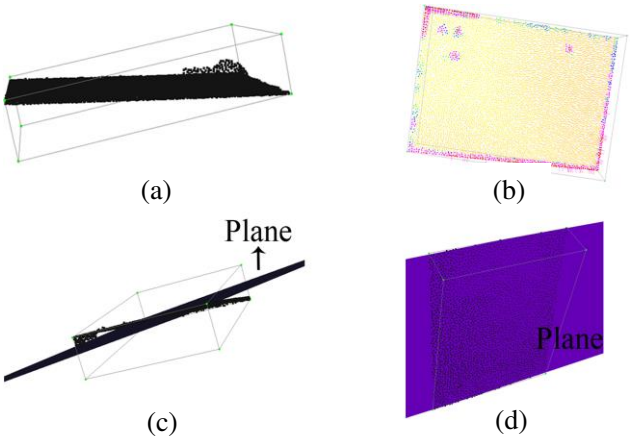

Figure 6: *Plane fitting by RANSAC. (a) Input point clouds with outliers; (b) Curvature computation; (c) Plane fitting by least squares method; (d) Plane fitting by RANSAC.*

3.2.2 Fine segmentation

To improve the refinement, let every point $p_i = (x_i, y_i, z_i) \in \mathbb{R}^3$ project on the corresponding best-fit-plane. The corresponding projection point $p''_i = (x''_i, y''_i, z''_i)$ is defined as:

$$\begin{cases} x''_i = x_i - a \cdot t \\ y''_i = y_i - b \cdot t \\ z''_i = z_i + t \end{cases} \tag{9}$$

where $t = \dfrac{ax_i + by_i - z_i + c}{a^2 + b^2 + 1}$.

The distance between point and corresponding plane is calculated and the point is added to the current plane patch only when the distance is less than a threshold.

3.3 Indoor scene representation

Indoor scene understanding has application in architecture and design. Instead of producing an over-detailed triangle mesh from a point cloud, it is more useful to segment a limited number of planar patches. Our method produces an effective scene representation in three steps. First, we segment the indoor scene into various components (such as room and window) according to the geometry layout. Second, we finely segment planar patches for each component. Third, we devise a data structure to describe the connection between components and planar patches. In our study, the procedures of splitting window and room from point clouds are as follows:

Step 1: Layout calculation by projecting the point cloud on the best-fit-plane corresponding to the floor. The projection points for floor (Figure **7**b) and the projection points for indoor scene except ceilings (Figure 7c) are respectively denoted by set A and B;

Step 2: Label projection points. The intersections of two sets are labelled and taken as projection points for the point cloud of the room, and the complements of the two sets are labelled and taken as projection points for the point cloud of the window (Figure 7d);

Step 3: Segmentation. According to the label of every projection point, room (Figure 7e) and window (Figure 7f) can be separated.

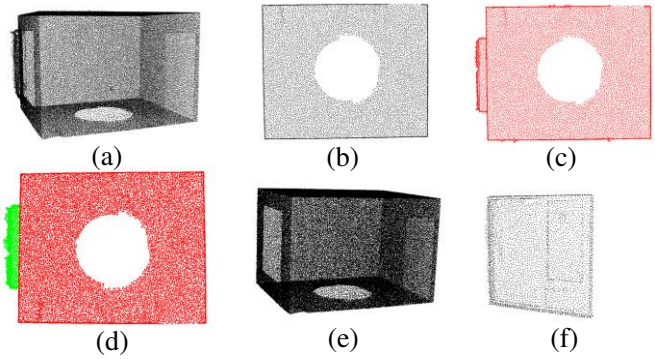

Figure 7: *Window and room separation from indoor point cloud. (a) Input indoor point cloud; (b) Projection points for floor; (c) Projection points for indoor scene except ceilings; (d) Projection points segmentation; (e) Room segmentation ;(f) Window segmentation.*

4 Experimental Results

We used a Faro Focus3D, a high-performance time-of-flight TLS, to record the point cloud of an indoor scene, consisting of more than 120,000 points after Poisson-disk sampling. All experiments were performed on an Intel Core i7 computer with 32GB RAM. All the algorithms were written using Microsoft visual studio 2008.net with C++ and OpenGL.

Curvature computation: Figure 8 shows the result of curvature computation on a real dataset using various radius. The results are sensitive to the choice of radius, which can negatively impact on the segmentation. In the experiment, we calculated the average distance between points in the point cloud and adjusted coeffeicent λ of equation 5 to obtain a suitable radius until the coarse segmentation was good. In addition, the direction of the normal at every point can impact on the coarse segmentation in our method, especially in the case of parallel surfaces. There are two possible directions for the normal using equation 3. To tackle this problem, [Pang et al. 2010] can be used to correct the normal direction.

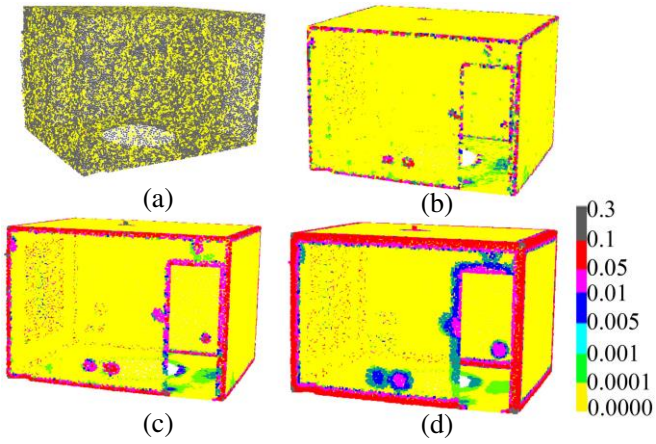

Figure 8: *Curvature computation for indoor scene using various radii. (a) Radius is* $2 \cdot dist$ *; (b) Radius is* $4 \cdot dist$ *; (c) Radius is* $6 \cdot dist$ *; (d) Radius is* $10 \cdot dist$.

Coarse segmentation: In Figures 9 and 10, curvature, normal, Gaussian map and region growing are depicted and planes are visualized by various colours. Figure 9 shows the results of coarse

segmentation form the point cloud. We calculated the curvature (Figure 9a) and normal (Figure 9b) of every point, and then every point was transformed by Gaussian map (Figure 9c). After region growing (Figure 9d), the majority of points in the point cloud located on the planar shape have been extracted (Figure 9e). Neighbour points at boundary and corner regions belong to multiple planes, thus over-segmentation occurs around the boundary regions. Figure 10 shows an example of planar shape segmentation from the point cloud of the window.

Fine segmentation: Figure 11 shows the segmentation results from the point cloud of the window after fine segmentation, which consists of five planes. From left to right, each figure shows the point clouds after coarse segmentation, fitted plane using RANSAC and fine segmentation. As seen from these experiments, regardless of the results of coarse segmentation, our method can improve the accuracy of planar patches segmentation after fine segmentation.

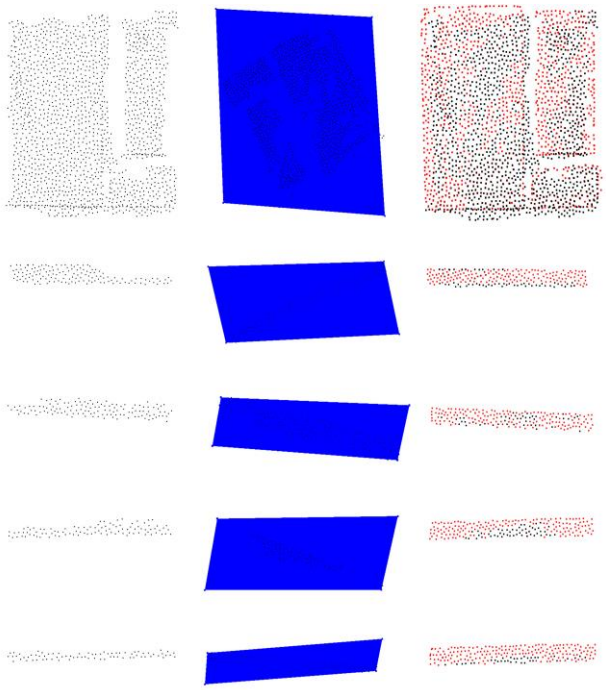

Figure 11: *Fine segmentation from point cloud of the window.*

Figure 12 shows the segmentation results from the point cloud of the room after fine segmentation, which consists of six planes. Each left figure shows the point cloud after coarse segmentation (black colour) and right figure shows fine segmentation (red colour). It can be clearly seen that the bottom corner of the door and boundary of every plane cannot be accurately segmented after coarse segmentation.

Figure 13 shows the final segmentation results of the indoor scene using our method. We compare our planar patches segmentation process against a previous method based on curvature and normal similarity (Figure 5). Our method produces a better result. Three aspects are worth commenting on: (a) Noise is the main issue for segmentation of the window, whereas the main issue of room segmentation is sharp features on the corners and boundary regions. We first split the point cloud of the indoor scene into window and room, and then a suitable radius can be chosen for every component. (b) We projected point clouds on the best-fit-plane to extract planar shape point clouds, overcoming the limitations of the geometric similarity method. (c) A TLS can produce a point cloud with holes, or missing data, as in the floor for our data. We filled the holes on the floor by projecting the point cloud on the plane corresponding to floor.

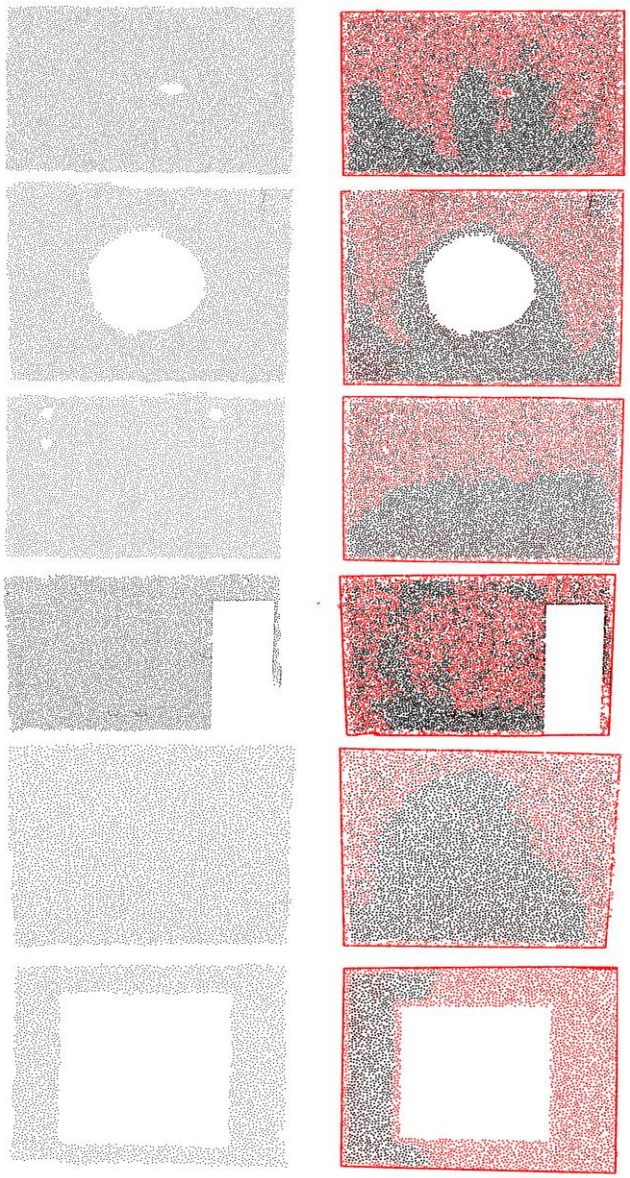

Figure 12: *Fine segmentation from point cloud of the room*

To demonstrate our method for a data set which includes missing regions and occlusion, we tested it on a complicated point cloud, consisting of a room with window, table, chair, human, etc. (Figure 14a). Figure 14b shows the normal and curvature for every point. The coarse segmentation is illustrated in Figures 14c and 14d. Planer patches, including walls, floor, ceilings, and desktop have been illustrated in various colors Figure 14e). The refinement planar shape segmentation is shown in Figure 14f.

Indoor scene representation: After indoor scene segmentation and planar patch extraction for each component for our test scene (Figure 13), we identified and traced the boundary points of each planar patch (Figure 15). As seen in Figure 16, the indoor scene can be represented as a room and a window. Furthermore, the room can be represented by a graph structure, which contains six nodes. The

window can also be represented by a graph structure which contains five nodes. Finally, this digital model with texture was imported to Unity 3D to enable members of the public to virtually experience the indoor scene online.

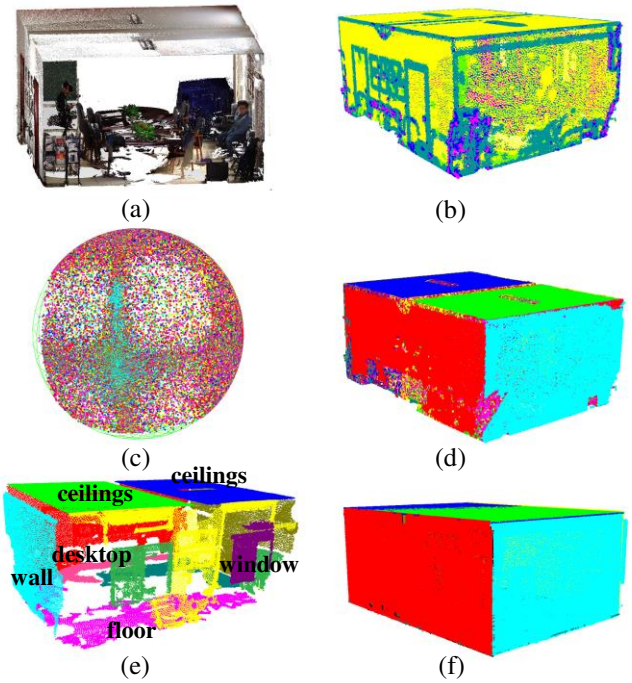

Figure 14: *Planar shape segmentation from a complicated indoor scene. (a) Input indoor point clouds with large missing region; (b) Curvature and normal computation; (c) Region growing on the Gaussian map; (d)coarse segmentation;(e) various planer shapes ;(f) fine segmentation.*

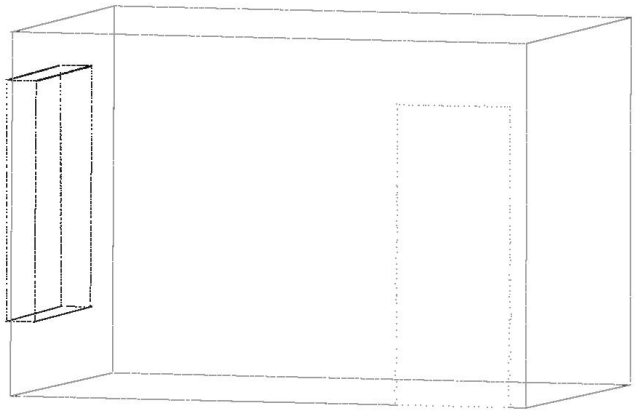

Figure 15: *Boundary of every planar patch*

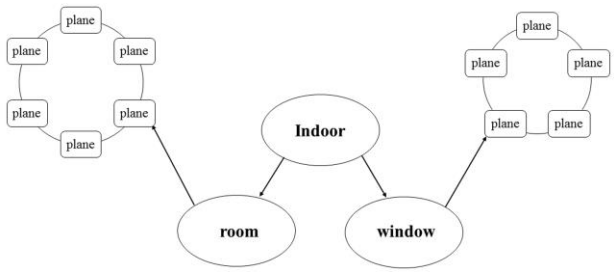

Figure 16: *Indoor scene geometric structure*

5 Conclusion and Future work

In this article, we have proposed a novel coarse to fine segmentation method for planar shape segmentation from TLS point clouds. The Gaussian map and region growing techniques were used to coarsely segment the planar shapes from sampled point clouds. The best-fit-plane was calculated by RANSAC, avoiding the negative impact of outliers. The refinement of planar shape was produced by projecting a point cloud on a corresponding best-fit-plane. The benefit of our method is to overcome the over- and under-segmentation caused by the previous geometric similarity methods. We also provided a simplified representation for an indoor scene based on a graph structure and a limited number of planar patches.

Future work will focus on segmenting planar shapes from complicated indoor scenes containing clutter and occlusions. In addition, we will also extend our method to indoor scenes consisting of multiple rooms.

References

ADAN, A., QUINTANA, B., V ZQUEZ, A.S., OLIVARES, A., PARRA, E. AND PRIETO, S. 2015. Towards the automatic scanning of indoors with robots. *Sensors* 15,5, 11551-11574.

AWWAD, T.M., ZHU, Q., DU, Z. AND ZHANG, Y. 2010. An improved segmentation approach for planar surfaces from unstructured 3D point clouds. *The Photogrammetric Record* 25,129, 5-23.

BIOSCA, J.M. AND LERMA, J.L. 2008. Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods. *ISPRS Journal of Photogrammetry and Remote Sensing* 63,1, 84-98.

BUCK, U., NAETHER, S., RASS, B., JACKOWSKI, C., AND THALI, M.J. 2013. Accident or homicide–virtual crime scene reconstruction using 3D methods. *Forensic Sci. Int.* 225, 1-3, 75-84.

CROSILLA, F., VISINTINI, D. AND SEPIC, F. 2009. Reliable automatic classification and segmentation of laser point clouds by statistical analysis of surface curvature values. *Applied Geomatics 1*, 17-30.

DESCHAUD, J., AND GOULETTE, F. 2010. A fast and accurate plane detection algorithm for large noisy point clouds using filtered normals and voxel growing. *In Proceedings of 3D Processing, Visualization and Transmission Conference (3DPVT2010).*

DOUROS, I. AND BUXTON, B.F. 2002. Three-dimensional surface curvature estimation using quadric surface patches. *Scanning*.

FISCHLER, M.A. AND BOLLES, R.C. 1981. Random sample consensus: a paradigm for model fitting with applications to

image analysis and automated cartography. *Communications of the ACM* 24, 6, 381-395.

GUO, J., YAN, D.M., JIA, X. AND ZHANG, X. 2015. Efficient maximal Poisson-disk sampling and remeshing on surfaces. *Computers & Graphics* 46, 72-79.

HOPPE, H., DEROSE, T., DUCHAMP, T., MCDONALD, J. AND STUETZLE, W. 1992. Surface reconstruction from unorganized points. *In Proceedings of SIGGRAPH'92*, 71-78.

IZADI, S., KIM, D., HILLIGES, O., MOLYNEAUX, D., NEWCOMBE, R., KOHLI, P., SHOTTON, J., HODGES, S., FREEMAN, D. AND DAVISON, A. 2011. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. *In Proceedings of the 24th annual ACM symposium on User interface software and technology*, 559-568.

KAZHDAN, M., BOLITHO, M. AND HOPPE, H. 2006. Poisson surface reconstruction. *In Proceedings of the fourth Eurographics symposium on Geometry processing*, 61–70.

LENG, X., XIAO, J. AND WANG, Y. 2016. A multi-scale plane-detection method based on the Hough transform and region growing. *The Photogrammetric Record* 31, 154, 166-192.

LIU, Y. AND XIONG, Y. 2008. Automatic segmentation of unorganized noisy point clouds based on the Gaussian map. *Computer-Aided Design* 40, 5, 576-594.

MA, L., FAVIER, R., DO, L., BONDAREV, E. AND DE WITH, P.H. 2013. Plane segmentation and decimation of point clouds for 3D environment reconstruction. *IEEE 10th Consumer Communications and Networking Conference (CCNC)*, 43-49.

MITRA, N.J. AND NGUYEN, A. 2003. Estimating surface normals in noisy point cloud data. *In Proceedings of the nineteenth annual symposium on Computational geometry*, 322-328.

MOZOS, O.M., MIZUTANI, H., KURAZUME, R. AND HASEGAWA, T. 2012. Categorization of indoor places using the kinect sensor. *Sensors* 12, 5, 6695-6711.

MURA, C., MATTAUSCH, O., VILLANUEVA, A.J., GOBBETTI, E. AND PAJAROLA, R. 2014. Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts. *Computers & Graphics* 44, 20-32.

NAN, L., XIE, K. AND SHARF, A. 2012. A search-classify approach for cluttered indoor scene understanding. *ACM Transactions on Graphics* 31, 6, Article 137.

NURUNNABI, A., WEST, G. AND BELTON, D. 2015. Outlier detection and robust normal-curvature estimation in mobile laser scanning 3D point cloud data. *Pattern Recognition* 48, 4, 1404-1419.

OCHMANN, S., VOCK, R., WESSEL, R. AND KLEIN, R. 2016. Automatic reconstruction of parametric building models from indoor point clouds. *Computers & Graphics* 54, 94-103.

OESAU, S., LAFARGE, F. AND ALLIEZ, P. 2014. Indoor scene reconstruction using feature sensitive primitive extraction and graph-cut. *ISPRS Journal of Photogrammetry and Remote Sensing* 90, 68-82.

PAULY, M., GROSS, M. AND KOBBELT, L.P. 2002. Efficient simplification of point-sampled surfaces. *In Proceedings of the IEEE conference on Visualization'02*, 163-170.

PAULY, M., KEISER, R. AND GROSS, M. 2003. Multi-scale Feature Extraction on Point-Sampled Surfaces. *Computer Graphics Forum* 22, 3, 281-289.

PAULY, M., KEISER, R., KOBBELT, L.P. AND GROSS, M. 2003. Shape modeling with point-sampled geometry. *ACM Transactions on Graphics* 22, 3, 641-650.

RABBANI, T., VAN DEN HEUVEL, F. AND VOSSELMANN, G. 2006. Segmentation of point clouds using smoothness constraint. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 36, 248-253.

SCHNABEL, R., WAHL, R. AND KLEIN, R. 2007. Efficient RANSAC for Point-Cloud Shape Detection. C*omputer Graphics Forum* 26, 2, 214-226.

SHAO, T., XU, W., ZHOU, K., WANG, J., LI, D. AND GUO, B. 2012. An interactive approach to semantic modeling of indoor scenes with an RGBD camera. *ACM Transactions on Graphics* 31, 6, Article 136.

SILBERMAN, N. AND FERGUS, R. 2011. Indoor scene segmentation using a structured light sensor. P*roceeding of IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 601-608.

SILBERMAN, N., HOIEM, D., KOHLI, P. AND FERGUS, R. 2012. Indoor segmentation and support inference from RGBD images. *Proceeding of European Conference on Computer Vision (ECCV'12)*, 746-760.

TARSHA-KURDI, F., LANDES, T. AND GRUSSENMEYER, P. 2007. Hough-transform and extended ransac algorithms for automatic detection of 3d building roof planes from lidar data. *In Proceedings of the ISPRS Workshop on Laser Scanning*, 407-412.

VC, H.P. 1962. Method and means for recognizing complex patterns. *In US Patent*, 1962

VOSSELMAN, G. AND DIJKMAN, S. 2001. 3D building model reconstruction from point clouds and ground plans. *International archives of photogrammetry remote sensing and spatial information sciences* 34, 37-44.

WANG, Y., FENG, H.Y., DELORME, F.É. AND ENGIN, S. 2013. An adaptive normal estimation method for scanned point clouds with sharp features. *Computer-Aided Design* 45, 11, 1333-1348.

WANG, Y., HAO, W., NING, X., ZHAO, M., ZHANG, J., SHI, Z. AND ZHANG, X. 2013. Automatic segmentation of urban point clouds based on the Gaussian map. *The Photogrammetric Record* 28, 144, 342-361.

WEBER, C., HAHMANN, S. AND HAGEN, H. 2010. Sharp feature detection in point clouds. *In Proceeding of Shape Modeling International Conference (SMI' 10)*, 175-186.

WHELAN, T., KAESS, M., JOHANNSSON, H., FALLON, M., LEONARD, J.J. AND MCDONALD, J. 2015. Real-time large-

scale dense RGB-D SLAM with volumetric fusion. *The International Journal of Robotics Research* 34, 598-626.

XIONG, X., ADAN, A., AKINCI, B. AND HUBER, D. 2013. Automatic creation of semantically rich 3D building models from laser scanner data. *Automation in Construction* 31, 325-337.

PANG X., PANG.,M,, P. AND XIAO, C. 2010. An algorithm for extracting and enhancing valley-ridge features from point sets. *Acta Automatica Sinica* 36, 1074-1083.

ZHANG, X., LI, G., XIONG, Y. AND HE, F. 2008. 3D mesh segmentation using mean-shifted curvature. *In International Conference on Geometric Modeling and Processing*, 465-474.

ZOU, Y., CHEN, W., WU, X. AND LIU, Z. 2012. Indoor localization and 3D scene reconstruction for mobile robots using the Microsoft Kinect sensor. *In IEEE 10th International Conference on Industrial Informatics*, 1182-1187.

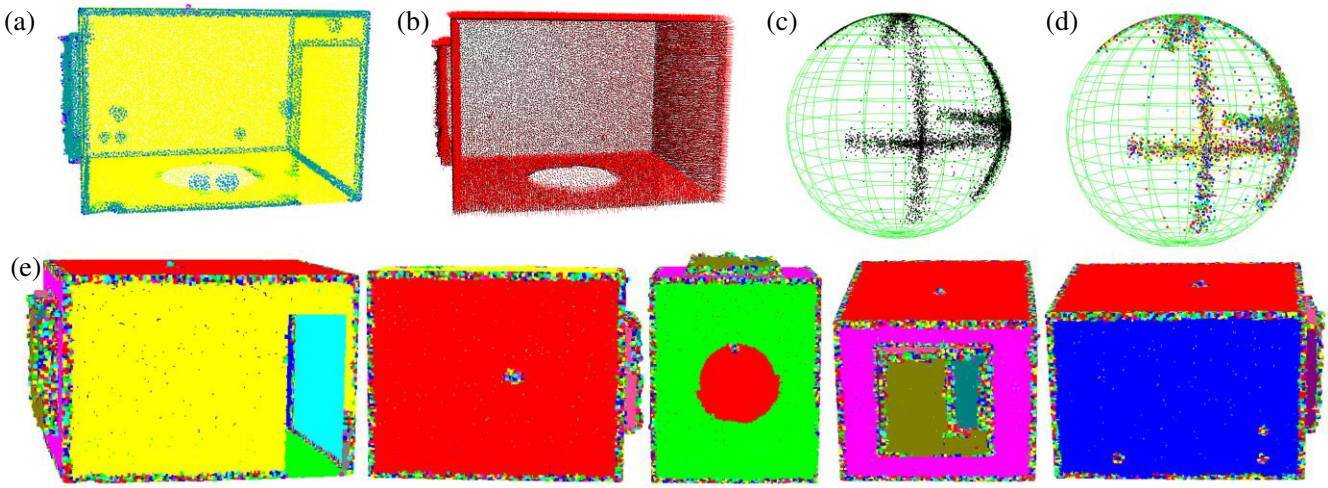

Figure 9: *Coarse segmentation. (a) Curvature computation; (b) Normal computation: the red line segments denote the normals of every point;(c) Point clouds located on the Gaussian map; (d) Coarse segmentation using region growing; (e) Coarse segmentation results from different views.*

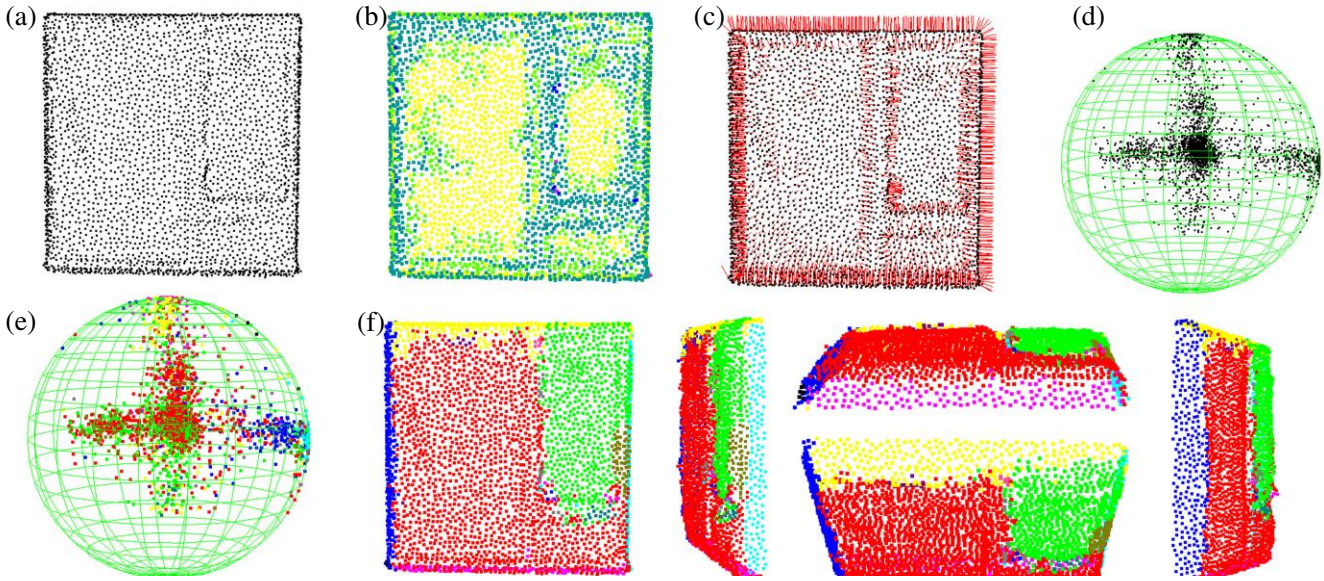

Figure 10: *Coarse segmentation of window. (a) Point clouds of window; (b) Curvature computation; (c) Normal computation, the red line segments denote the normals of every point; (d) Point clouds located on the Gaussian sphere; (e) Segmentation using region growing; (f) Coarse segmentation results of window from different views.*

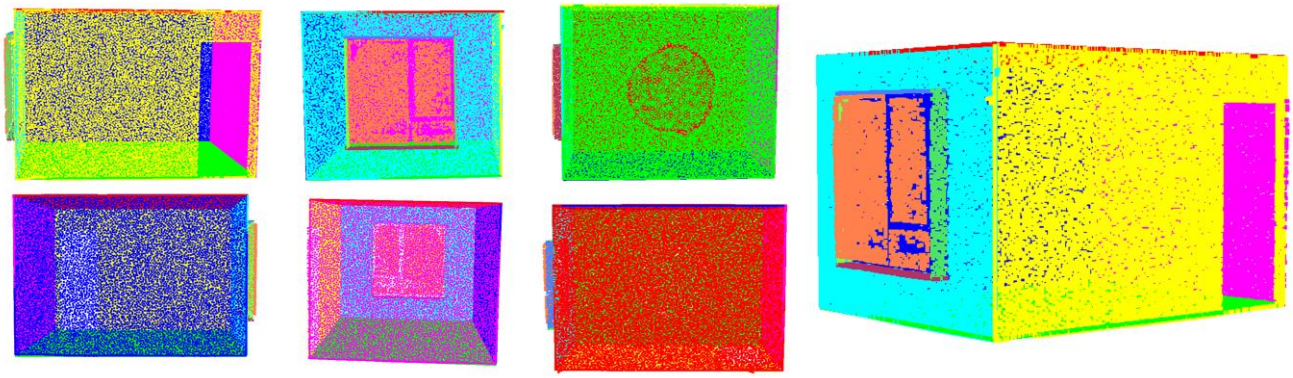

Figure 13: *Fine segmentation from point clouds of indoor scene.*