Segmentation of 3D outdoor scenes using hierarchical clustering structure and perceptual grouping laws

Yusheng Xu, Sebastian Tuttas, and Uwe Stilla
Photogrammetry and Remote Sensing
Technische Universität München
80333 Munich, Germany
{yusheng.xu, sebastian.tuttas, stilla}@tum.de

***Abstract*—In this paper, we propose a bottom-up point cloud segmentation method, which utilizes a hierarchical clustering structure combined with the perceptual grouping laws. Our method allows a learning-free and completely automatic but parametric process for segmenting point clouds of 3D outdoor scenes. The experiments using terrestrial laser scanning dataset have demonstrated that our proposed method can achieve good results, especially for complex scenes and nonplanar surfaces of objects. The quantitative comparison between our method and the region growing based method also confirms the effectiveness and efficiency of our method.**

Keywords-point cloud; outdoor scene segmentation; hierarchical clustering; perceptual grouping laws

I. INTRODUCTION

In the last decade, point clouds obtained via laser scanning, photogrammetry, and range imaging cameras have been widely employed to represent 3D geospatial information, with applications in a variety of fields, including geodesy, geomatics, geology, forestry, and archaeology. Among all the applications, the 3D scene reconstruction is of increasing importance to many related tasks such as constructing virtual reality, generating digital surface models (DSM), urban planning, or monitoring construction projects. In particular, since the emergence of Building Information Model (BIM), point clouds have been proved to be quite suitable for reconstructing geometric models as the 3D points provide direct measurements of 3D coordinates of the building surfaces sinuously [1].

However, the majority of 3D indoor and outdoor scenes consist of various objects and complex structures, having various surfaces, sections, and combinations. Thus, in practical applications, prior to the reconstruction of 3D models, individual objects must be identified from the point clouds. To achieve this task, segmentation algorithms are normally applied to the unstructured and massive original point cloud datasets. An effective segmentation algorithm can significantly increase the quality of models reconstructed and largely decrease the workload, but the performance of conventional algorithms is always restrained by the complicated environment of real outdoor scenes. Therefore, apart from the effectiveness, the reliability plays a vital role in the development of segmentation algorithms as well. On the other hand, since the point cloud segmentation is computationally intensive [1], an efficient strategy for the segmentation of points of a large scale scene is also in urgent need.

II. RELATED WORK

Enormous attempts and contributions have been done for the point cloud segmentation, which can be roughly grouped into three major categories [1][2]: the model-based methods, the region growing-based methods, and the clustering based-methods.

The model-based segmentation methods directly fit the geometric shapes of objects according parametric equations of their mathematical models. For example, the 3D Hough Transform (HT) [3] and the RANSAC [4] are two kinds of widely used algorithms [5]. The HT and it variations adopt a voting strategy for extracting planes [6], cylinders [7], and spheres [8] from the point cloud in the parameter domain. Whereas RANSAC and its extensions directly estimate optimal parameters of the geometric models in spatial domain [4]. The advantage of the model-based methods is that they are robust to noise and outliers in the dataset and provide optimized parameters for modelling simultaneously. However, when coping with large scale datasets, they require normally a long processing time and high memory consumption [1]. Besides, these methods can hardly cope with objects having no explicit mathematical expressions like irregular shaped or curvature surfaces.

The region growing-based methods utilize an iterative process, which examines points in the vicinity of an initial seed and judge whether they belong to the region of the seed or not via given criterions. This examination of points is an inside-out process, like the growth from seeds. For selecting appropriate initial seeds, regions or points with the least curvature [9] are usually preferred. As for the growing criterions, the normal vectors [10][11] is the most commonly used growing criteria. Recently, structured voxels are introduced as basic elements for the growing process instead of points for the purpose of efficiency [1]. The growing-based methods are simple and can keep the boundaries well [11], but the selection of the growing criteria and seeds, which are significantly affected by the quality of dataset are influential to the segmentation performance, making this kind of method sensitive to noise and outliers.

The clustering-based methods group points having similarities in a spatial or feature space on the basis of their geometric characteristics or spatial distribution. This kind of methods requires no seeds and deems more robust under noisy conditions. Euclidean distance [12] and normal vector [1] are representative similarity measures for the clustering. Recently, the graph-based algorithms are frequently used for the clustering process as well. The graph cut based methods [13][14], the Markov Random Field (MRF) [15] or Conditional Random Field (CRF) [16][17] based methods are the representatives. However, clustering methods commonly suffer expensive computational cost due to the calculation of similarities and the optimization of cost functions [1][18].

The overall goal of this research is to automatically segment 3D point clouds of different outdoor scenes. Here, the definition of segmentation is limited to partitioning point clouds of the entire scene into meaningful segments on the basis of their natural geometric characteristics. Especially, we intend to segment the surfaces of buildings into individual objects, referring to logical groups of points pertaining to same structures, which can be easily used for the further modelling work. To that end, we develop a learning-free and bottom-up segmentation method utilizing a hierarchical clustering structure based on the perceptual grouping laws, allowing completely automatic but parametric segmentation of outdoor scene from point clouds. The proposed method clusters oversegmented patches of different levels in a hierarchical structure, in which different principles of the perceptual grouping laws are considered as the basic criterion of clustering.

III. METHODOLOGY

A. *Workflow*

Conceptually, the implementation of our segmentation method focuses on two crucial aspects: the clustering criterions and the hierarchical clustering structure. The workflow is depicted in Fig. 1, with involved algorithms and sample results illustrated. All the approaches and algorithms shown in this workflow will be introduced in the following sections.

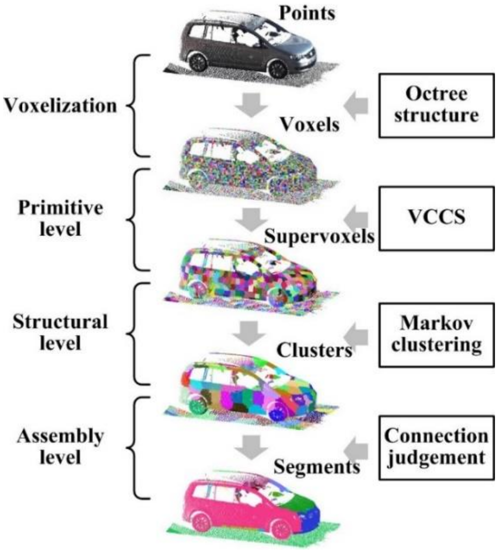


Figure 1. Workflow of the proposed segmentation method.

B. *Clustering criterion based on perceptual grouping laws*

Perceptual grouping has a long history in the field of computer vision for recognizing objects from the scene, which refers to the process of determining regions and parts of the visual scene belonging to the same part of higher-level perceptual units (e.g., objects or patterns) [19]. In our work, we aim to segment the outdoor scene by imitating the natural way of human perceiving objects, following the perceptual grouping laws combined with our hierarchical structure. To that end, three representative principles of the grouping laws are selected as our clustering criterion, namely the proximity, the similarity, and the continuity.

According to [20], the proximity principle claims that elements are highly likely to be aggregated into a same group when they are close to each other. Whereas the similarity principle states that the elements tend to be summed into a group if they resemble each other. As for the continuity principle, it indicates that the oriented elements are disposed to be integrated into one part if they can be aligned with each other.

To measure the proximity between two elements V_i and V_j (i.e., voxels or supervoxels), we consider the spatial distance $d(V_i,V_j)=\left\|\vec{X}_i^c-\vec{X}_j^c\right\|$ referring to the Euclidean distance between the centroids of points within the elements. Here, $\vec{X}^c$ is a vector from the origin to the centroid of the element.

As for the similarity, the geometric coherence of shapes formed by the points are estimated as criteria. To achieve this, we use the eigenvalue based geometric features [21] to delineating the 3D properties of points inside an element. For the points within an element, their eigenvalues $e_1 \geq e_2 \geq e_3 \geq 0$ can be calculated via the covariance tensor $C=\frac{1}{k}\sum_{i\in V}(\vec{X}_i-\bar{X})(\vec{X}_i-\bar{X})^T$, where $\vec{X}_i$ denotes the point within the element V_i, k represents the number of points, and the $\bar{X}$ stands for their medoid. All the eight geometric features employed are listed in Table 1 [21].

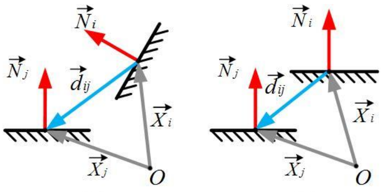
With the geometric features $\bar{S}_i$ and $\bar{S}_j$ of two elements, their similarity $\rho(V_i,V_j)$ is defined by the Person product-moment correlation coefficient as shown in (1):

$$\rho(V_i,V_j)=\frac{\sum_{k=1}^{8}(S_{ik}-\bar{S}_i)(S_{jk}-\bar{S}_j)}{\sqrt{\sum_{k=1}^{8}(S_{ik}-\bar{S}_i)^2}\sqrt{\sum_{k=1}^{8}(S_{jk}-\bar{S}_j)^2}} \tag{1}$$

When it comes to the continuity, we consider the smoothness of the surfaces formed by the points, which is quantified by two contributions: the angle between normal vectors $\vec{N}$ and the offset between surfaces. The implicit plane representation (see (2)) is introduced by assuming that the points in each element can approximate a planar surface with a normal vector and a centroid.

$$<\vec{N},\vec{X}> - D = 0 \tag{2}$$

where D refers to the distance from the origin to the plane.



(a) (b)

Figure 2. Connections between surfaces of elements. (a) General case. (b) "Stair-like" case.

TABLE I. EIGEN VALUE BASED GEOMETRIC FEATURES

Eigenvalue based geometric features	
Sum	$e_1 + e_2 + e_3$
Omnivariance	$(e_1 \cdot e_2 \cdot e_3)^{1/3}$
Eigenentropy	$-\sum_{i=1}^{3} e_i \cdot \ln(e_i)$
Anisotropy	$(e_1 - e_3)/e_1$
Linearity	$(e_2 - e_3)/e_1$
Planarity	$(e_1 - e_2)/e_1$
Surface variation	$e_3/(e_1 + e_2 + e_3)$
Sphericity	e_3/e_1

The angle between approximated planar surfaces is calculated by normal vectors $\vec{N}$, referring to the smallest eigenvectors of the aforementioned covariance tensor C. Here, the angle $g_1(V_i,V_j) = <\vec{N}_i,\vec{N}_j>$ is calculated via the inner product of the normal vectors. Whereas the offset between surfaces is related to the distance between parallel surfaces resembling a "stair-like" connection. To measure this offset, we adopt the distance of the centroid from the best fitting plane of the other centroids [22], which is defined by the following:

$$g_2(V_i,V_j) = \frac{(<\vec{N}_i,\vec{X}_j> - D_i)^2 + (<\vec{N}_j,\vec{X}_i> - D_j)^2}{\left\|\vec{d}_{ij}\right\|^2} \qquad (3)$$

where $\vec{d}_{ij}$ is the vector between the centroids of two elements. Similar to the method described in [22], we also combine these two contributions in a multiplicative way in order to estimate a good coincidence of two connected elements. Thus, the smoothness $s(V_i,V_j)$ is equal to $g_1(V_i,V_j) \cdot g_2(V_i,V_j)$.

C. Hierarchical clustering structure

The hierarchical clustering structure we use is similar to the classificatory structure proposed in [23], which is a bottom-up strategy consisting of three different clustering levels, namely the primitive level, the structural level, and the assembly level. Different grouping principles are used in the different clustering levels. It is notable that we do not define the first signal level mentioned in [23] in our clustering structure. This is because we have already conducted the voxelization to pre-cluster points into 3D voxels as fundamental elements.

The primitive-level is the clustering of voxels carried out by the Voxel Cloud Connectivity Segmentation method (VCCS) [24], which is an effective region growing based oversegmentation method extracting supervoxels from voxels of points. The structural-level represents the aggregation of supervoxels conducted via the local affinity graph and Markov Clustering Algorithm (MCL) [25], in order to group supervoxels into clusters. The assembly-level denotes the merging of clusters achieved by the judgement of connections between adjacent clusters. By the use of this hierarchical clustering structure, we can utilize the features of different abstraction levels in terms of their complexity, which is in coincidence with the perceptual grouping laws we followed.

D. Octree-based voxelization

The octree-based voxelization has been commonly employed in point cloud processing [1][18]. Compared with the processing at point level, the use of voxels can suppress the noise and outliers existing in the point clouds and greatly reduce the computational cost simultaneously. In addition, the effect of anisotropic densities of points caused by the varying distances between the sensor and objects in an outdoor scene [16] can be partially reduced by the voxelization process. We adopt the octree-based voxelization to rasterize the entire point cloud with 3D cubic grids, the nodes of which have explicit linking relations, facilitating the traversal for searching the adjacent ones [1]. However, selecting the size of voxels is a trade-off between the efficiency of computations and the preservation of details. The larger the voxel, the more details will be blurred. In our work, the size of voxel is selected according to the application and the density of points.

E. Clustering of voxels

The VCCS method clusters the voxels of points in terms of the distance between the seed and candidate voxels in a feature space, involving the geometrical features, and RGB colors [24]. Slightly different from the method described in [24], we merely use the normal vectors and spatial coordinates of voxels to define the distance, which is related to the proximity and continuity principles. The VCCS we used is implemented and tailored from the Point Cloud Library (PCL) [26].

One of the most significant advantage of the VCCS is the boundary preservation performance [24], so that we can obtain the supervoxels sharing same boundaries with the major structures of objects in the scene. Note that, the size of the voxel and the resolution of seeds can greatly affect the performance of VCCS. The former one will determine the details preserved in the scene, while the later will influence the effectiveness of keeping boundaries. Empirically, we set these two factors according to the density of points and the varying range from the sensor to the objects.

F. Aggregation of supervoxels

As we mentioned before, the aggregation of supervoxels includes two steps: the construction of local affinity graphs and the partition of these graphs using MCL algorithm.

The adjacency relationship acts as a significant aspect in the aggregation of supervoxels. In the vicinity of each supervoxel, its adjacencies defined by a radius R_a are found to construct a full-connected local affinity graph, in which each node represent a supervoxel. The size of R_a determines the number of adjacent supervoxels in one local affinity graph. The weight of edges in the affinity graph is defined the aforementioned principles of grouping laws. In detail, for two adjacent supervoxels V_i and V_j, the weight w_{ij} of the edge connecting them is estimated by the aforementioned three criterions, namely the spatial distance $d(V_i,V_j)$, the surface smoothness $s(V_i,V_j)$, and the geometric similarity $\rho(V_i,V_j)$, forming the corresponding weights w_{ij}^d, w_{ij}^s, and w_{ij}^ρ, respectively. The weights are calculated via the Gaussian kernel:

$$w_{ij}^d = \exp[-\frac{d(V_i,V_j)}{\sigma_d^2}] \tag{4}$$

$$w_{ij}^s = \exp[-\frac{S(V_i,V_j)}{\sigma_s^2}] \tag{5}$$

$$w_{ij}^\rho = \exp[-\frac{\rho(V_i,V_j)}{2\sigma_\rho^2}] \tag{6}$$

where σ_d, σ_s, and σ_ρ are the global scale factors. The overall weight between two supervoxel is defined in a multiplicative way constrained by the distance:

$$W_{ij} = w_{ij}^d \cdot (w_{ij}^s + w_{ij}^\rho)/2 \tag{7}$$

To partition the local affinity graph, the MCL algorithm is adopted by clustering the nodes in terms of their probabilities of connections. For each supervoxel, the nodes (i.e., supervoxels) clustered into the same group by partitioning its local affinity graph deem to be connected as a cluster. By this way, each supervoxel will be assigned to a cluster containing several connected and adjacent supervoxels. It is noteworthy that we directly use the weights of edges in the affinity matrix to represent the probability of connection between two nodes.

G. Merging of clusters

As we described in the process of the assembly-level, each cluster consists of several supervoxels having similar perceptual characteristics. Since the size of local affinity graph is larger than the size of supervoxel, so there must be supervoxel being judged as parts of more than one cluster in the MCL clustering. These clusters sharing the same supervoxel members will be identified as the "connected ones", and then merged together to form a segment. To achieve this process, the connections of all the supervoxels will be judged, so that whether two clusters should be merged or not can be determined.

IV. EXPERIMENTS AND RESULTS

The experimental datasets used are the terrestrial laser scanning point clouds from the large-scale point cloud classification benchmark datasets published on: www.semantic3d.net by ETH Zurich (Hackel et al., 2016). Specifically, two point clouds of different scenes are tested: one is measured in the urban area of the cathedral of St. Gallen, while the other one is obtained from a farm located in a rural area. The sizes of these two point clouds are around 3.7 million and 7.5 million points, respectively. Compared with optical based photogrammetric techniques (e.g., 3D reconstruction using aerial and spaceborne images), laser scanning can avoid influences of shadows and changing illuminations resulting from sunlight conditions and relief the uncertainties of matching points caused by the low texture areas in the images.

The size of voxels, the resolution of seeds, the radius of local affinity graph, and the parameters of MCL algorithm are factors that can significantly affect the final performance of our proposed method in the overall workflow. To be specific, the size of voxels will limit the representation of details in the final segments. While the resolution of seeds can directly influence on the preservation of edges and boundaries. The radius of graph will determine the degree of accuracy for the final segments, which should be larger than the resolution of seed. If the radius of graph is smaller than the seeds resolution, oversegmentations may happens because there will be no "connected supervoxels" for merging clusters. The parameters of MCL (e.g., particle size) will control the size and smoothness of resulting segments.

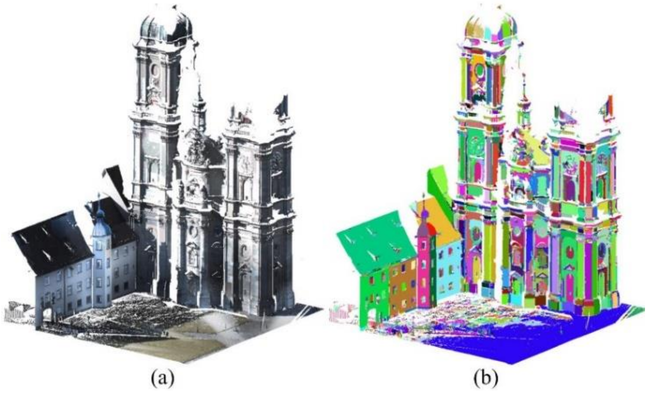


Figure 3. The segmentation results of Cathedral of St. Gallen. (a) RGB textured original point cloud. (b) Segmented point cloud.

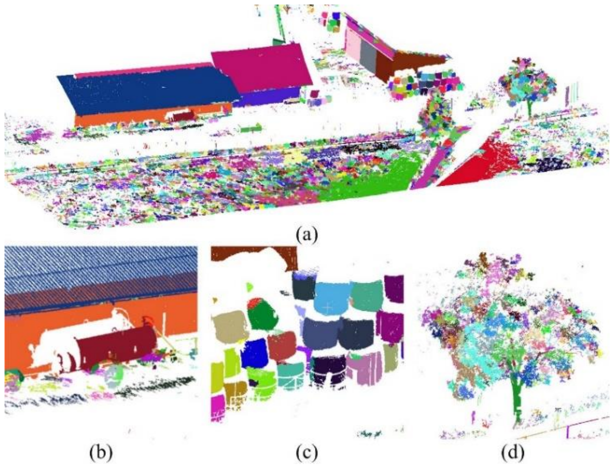
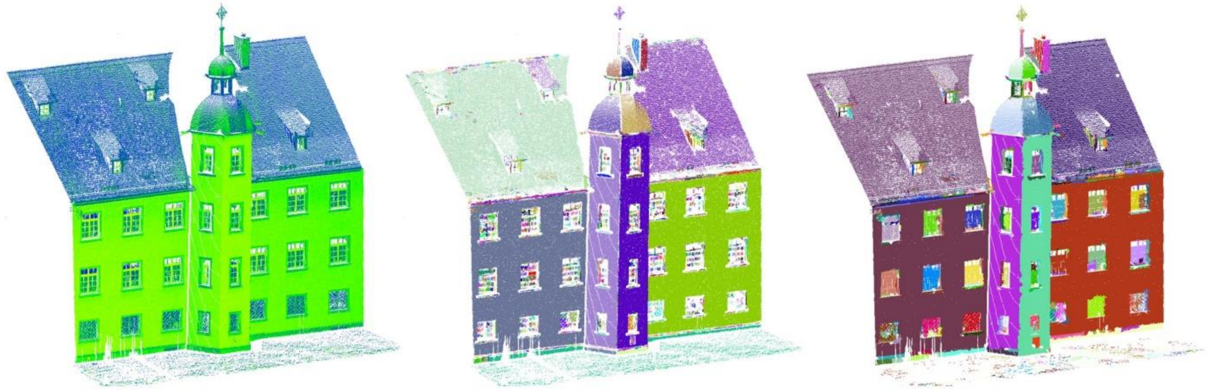


Figure 4. The segmentation results of the farm. (a) Segmented point cloud. (b)-(d) Details of some segments.



(a) (b) (c)

Figure 5. (a) The original point cloud. Segmentation results of using (b) region growing based method and (c) our method.

In Fig. 3, we illustrate the segmentation result of the Cathedral of St. Gallen. It can be seen from the result that our method achieves good segments for the planar, cylindrical, and curvature surfaces of objects in a complex outdoor scene. However, there is still many oversegmented and undersegmented parts due to the missing points caused by the occlusions. In Fig. 4a, the segmentation of the farm located in a rural area is displayed. Several detailed views of segments are zoomed in Figs. 4b-d. Similarly, for the parts with enough points, our method can perform well even for the objects having a complex background environment. However, for the parts with very sparse or uneven distributed points, like the grass and the crowns of trees, oversegmentations may occur.

In addition to qualitative results, a quantitative evaluation of the segmentation results is conducted as well, by comparing the segments against ground truth using two standard metrics: the precision and the recall [14], which are calculated via the true positive (TP), the true negative (TN), the false negative (FN), and false positive (FP). In theory, the higher the precision and the recall values are, the better the segmentation result is. Here, the ground truth we used is segmented manually, and the criteria of manual segmentation is based on the perceptual laws mentioned in Section IIIB using human visual inspection.

To make a comparison, the renowned smoothness based region growing [11] method is used as reference, which is implemented by the Point Cloud Library (PCL) [26]. A part of point cloud from Cathedral St. Gall is selected as test dataset, having around 0.65 million points (see Fig. 5a). The segmentation results using two different methods are displayed in Figs. 5b and 5c. Here, the radius of normal vector estimation for the region growing method is set as 0.05 m. While the size of voxels, the resolution of seeds, and the radius of graph used in our method are set 0.05 m, 0.25 m, and 0.6 m, respectively. Apparently, both of the two methods show acceptable segmentation results. Since the region growing based method uses the smoothness as the criteria, it can perform very well in the planar or smooth surfaces. However, the rough surfaces can significantly affect its performance. For example, the points of windows are segmented into broken small patches. In contrast, our method can better segment the rough surfaces and nonplanar objects, with the help of the perceptual laws. Moreover, the use of voxel based structure can also help us to suppress the influence of noise and outliers.

Quantitative evaluation results are given in Table II, where segmentation results of 10 major objects (4 wall surfaces, 4 dome surfaces, and 2 roof surfaces) are assessed. Seen from Table II, we can find that our method shows better performance than the region growing in precision values, but slightly inferior to the region growing method in the recall values. Such phenomenon can be explained by two aspects. One is that the voxel based structure will decrease the resolution of the segmentation. Especially for the edges and corners of the object, the use of voxels limits the ability of finding very accurate segmenting boundaries. In fact, the segmenting boundary found by our method is approximated by the edges of voxels belonging to the same segment. The other one is that we have tested merely the objects with simple geometric shapes in this validation. For the performance of our method in dealing with more complex objects, it still needs to be verified in our future work. In addition, as for the efficiency, for segmenting the test dataset, the computation time of our method and the region growing method is 19.2 s and 184.6 s, respectively, implemented in C++ on a computer with an Intel i7-4710MQ CPU. It reveals that our method is much faster than the region growing, with almost a same level of performance achieved.

TABLE II. QUANTITATIVE EVALUATION RESULTS

Criterions	**Precision**		**Recall**	
Method **Object**	*Region growing*	*Our method*	*Region growing*	*Our method*
Wall 1	0.8847	0.9570	0.9984	0.9866
Wall 2	0.8828	0.9400	0.9941	0.9227
Wall 3	0.8517	0.9293	0.9968	0.9503
Wall 4	0.8656	0.9401	0.9863	0.9283
Dome 1	0.8887	0.9476	0.9975	0.9908
Dome 2	0.8992	0.9833	0.9970	0.9716
Dome 3	0.8413	0.9705	0.9636	0.9194
Dome 4	0.8267	0.9825	0.9635	0.9501
Roof 1	0.7234	0.9195	0.9980	0.9958
Roof 2	0.7403	0.9114	0.9943	0.9930
Mean	0.8404	0.9481	0.9889	0.9609

V. CONCLUSIONS

In this work, we develop a bottom-up method for segmenting point clouds of the outdoor scenes, which utilizes a hierarchical clustering structure based on the perceptual grouping laws. Our method allows a learning-free and completely automatic but parametric segmentation process. The proposed method clusters oversegmented patches of different levels in a hierarchical structure, in which different principles of the perceptual grouping laws are considered as the basic criterion. The qualitative and quantitative experimental results show that our proposed method is effective and efficient, and can achieve good segments when dealing with complex scenes and nonplanar surfaces of objects. However, our proposed algorithm tends to fail on the cloud with very sparse points density (i.e., the ground areas), which should be improved in our future work. It is also noticed that we only compared our method with the most classical region growing method. As future steps, we will conduct more experiments using the state-of-the-art segmentation method in order to find solutions of improving our method. Beside, we also plan to increase the robustness of our method and set the sizes of voxel, supervoxel, and graph adaptively.

REFERENCES

[1] A. V. Vo, L. Truong-Hong, D. F. Laefer and M. Bertolotto, “Octree-based region growing for point cloud segmentation,” ISPRS J. Photogramm. Remote Sens., vol. 104, pp. 88-100, 2015.

[2] C. Wu, “Towards linear-time incremental structure from motion,” in Int. Conf. 3DV., Seattle, WA: pp. 127-134, 2013.

[3] D. H. Ballard, “Generalizing the Hough transform to detect arbitrary shapes,” Pattern recognition, vol. 13, no. 2, pp. 111-122, 1981.

[4] R. Schnabel, R. Wahl and R. Klein, “Efficient RANSAC for point‐cloud shape detection,” in Comput. graphics forum, vol. 26, no. 2, pp. 214-226, Blackwell Publishing Ltd, June. 2007.

[5] G. Vosselman, “Point cloud segmentation for urban scene classification,” in ISPRS Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci., 2013.

[6] G. Vosselman, B. G. Gorte, G. Sithole and T. Rabbani, “Recognising structure in laser scanner point clouds,” in Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., vol. 46, no. 8, pp. 33-38, 2004.

[7] F. Tarsha-Kurdi, T. Landes and P. Grussenmeyer, “Hough-transform and extended ransac algorithms for automatic detection of 3d building roof planes from lidar data,” in Proc. ISPRS Workshop Laser Scanning, vol. 36, pp. 407-412, September. 2007.

[8] T. Rabbani and F. Van Den Heuvel, “Efficient hough transform for automatic detection of cylinders in point clouds,” ISPRS WG III/3, III/4, no. 3, pp. 60-65, 2005.

[9] A. Nurunnabi, D. Belton and G. West, “Robust statistical approaches for local planar surface fitting in 3D laser scanning data,” ISPRS J. Photogramm. Remote Sens., vol. 96, pp. 106-122, 2014.

[10] D. Tóvári and N. Pfeifer, “Segmentation based robust interpolation-a new approach to laser data filtering,” in Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., vol. 36, no. 3/19, pp. 79-84, 2005.

[11] T. Rabbani and F. Van Den Heuvel and G. Vosselmann, “Segmentation of point clouds using smoothness constraint,” in Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., vol. 36, no. 5, pp. 248-253, 2006.

[12] A. Aldoma, Z. C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. B. Rusu, S. Gedikli, and M. Vincze, “Point cloud library,” IEEE Robotics & Automation Magazine, vol.1070, no. 9932/12, 2012.

[13] J. Shi and J. Malik, “Normalized cuts and image segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 8, pp. 888-905, 2000.

[14] A. Golovinskiy and T. Funkhouser, “Min-cut based segmentation of point clouds,” in IEEE 12th Int. Conf. Comput. Vis. Workshops, pp. 39-46, 2009.

[16] T. Hackel, J. D. Wegner, and K. Schindler, “Contour Detection in Unstructured 3D Point Clouds,” in Proc. CVPR 2016, pp. 1610-1618, 2016.

[17] R. B. Rusu, A. Holzbach, N. Blodow, and M. Beetz, “Fast geometric point labeling using conditional random fields,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 7-12, 2009.

[18] Y. T. Su, J. Bethel and S. Hu, “Octree-based segmentation for terrestrial LiDAR point cloud data in industrial applications,” ISPRS J. Photogramm. Remote Sens., vol. 113, pp. 59-74, 2016.

[19] A. Richtsfeld, T. Mörwald, J. Prankl, M. Zillich, and M. Vincze, “Learning of perceptual grouping for object segmentation on RGB-D data,” Journal of visual communication and image representation, vol. 25, no. 1, pp. 64-73, 2014.

[20] J. L. Brooks, “Traditional and new principles of perceptual grouping,” 2013.

[21] M. Weinmann, B. Jutzi, S. Hinz and C. Mallet, “Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers,” ISPRS J. Photogramm. Remote Sens., vol. 105, pp. 286-304, 2015.

[22] A. Dutta, J. Engels, and M. Hahn, “A Distance-Weighted Graph-Cut Method for the Segmentation of Laser Point Clouds,” in ISPRS Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci., vol. 40, no. 3, pp. 81, 2014.

[23] S. Sarkar and K. L. Boyer, “Perceptual organization in computer vision: A review and a proposal for a classificatory structure,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, no. 2, pp. 382-399, 1993.

[24] J. Papon, A. Abramov, M. Schoeler and F. Worgotter, “Voxel cloud connectivity segmentation-supervoxels for point clouds,” in Proc. CVPR, 2013, pp. 2027-2034.

[25] A. J. Enright, S. Van Dongen, C.A. Ouzounis, “An efficient algorithm for large-scale detection of protein families,” Nucleic Acids Research, vol. 30, no.7, pp.1575-1584, 2002.

[26] R. B. Rusu and S. Cousins, “3d is here: Point cloud library (pcl),” in Proc. IEEE Int. Conf. Robot. Autom., pp. 1-4, Sept. 2011.