EXTRACTION OF STREET TREES FROM MOBILE LASER SCANNING POINT CLOUDS BASED ON SUBDIVIDED DIMENSIONAL FEATURES

Pengdi Huang[1], Yiping Chen[1], Jonathan Li[1, 2], Yongtao Yu[1], Cheng Wang[1], Hongshan Nie[3]*

[1]Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, Fujian 361005, China (ypchenhk@gmail.com)
[2]GeoSTARS Lab, Department of Geography and Environmental Management, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada
[3]Hunan Intelligent Things Technology Co.,Ltd., Changsha, Hunan 410000, China

ABSTRACT

This paper proposes a method for automated extraction of street trees in a typical urban environment from 3D point cloud data acquired by the mobile laser scanning system. First, the algorithm utilizes the voxel-based method to remove the ground points from the scene. Second, the Euclidean distance clustering is adopted to cluster points into individual objects. The eigenvalues of neighborhood covariance matrix and the corresponding normalized centroid distance are computed for each point to obtain the subdivided dimensional features. Finally, the statistical component features and horizontal information are calculated for object detection. The experiment results show the feasibility of the proposed algorithm.

Index Terms— Mobile laser scanning, 3D point clouds, dimensional features, street tree detection, centroid-distance

1. INTRODUCTION

With the improvement of urban greening requirements, a large amount of vegetation is planted on both sides of a street such as landscape trees (called “street trees” in this paper). These trees are carefully positioned so as not to block street luminaries, and not influence utility lines above or below the ground. However, street trees still raise security problems. The climate and external forces make the weak vegetation a potential safety hazard. For example, a falling branch could injure a pedestrian or indiscriminate growth of the street trees could cover a sign pole, creating an obstructed view. Laser scanning technology has been promoted and widely applied to surveying and mapping fields in recent years. In particular, a mobile laser scanning system, with high accuracy, acquisition speed, and density of data sets, provides a systematic way to inventory urban vegetation. Therefore, research and the application value of tree extraction from point clouds are attracting more and more attention.

The inventory of urban street trees has been mostly accomplished by aerial images [1]. For point clouds, some researchers also launched targeted works of tree detection on 3D data sets. For forest inventory, a two-step method, which includes cluster searching and point density analysis, was presented in [2] to detect trees in a horizontal cross section . The following two approaches were presented in [3] to segment tree regions: (1) computing the eigenvalues of the covariance matrix for cylindrical and spherical environments, and (2) using the features of the echoes without neighborhood information. In [4], a method for tree detection was designed to first calculate the local geometrical features to locate foliage points, then to cluster similar points for retrieving individual trunks based on 2D morphological operations. In [5], the matched RGB imagery color values acquired by digital cameras were used as segmentation clues. Then, an elevation layer of points was divided after segmentation, followed by use of a discriminant rule to segment individual trees. A novel descriptor for simultaneously modeling the local and global geometric structures of a shape was proposed in [6] for roadside tree retrieval.

In previous studies, the main thought was to design descriptors or features (e.g., color, shape, local geometrical descriptor) relative to the prominent part (e.g., foliage, trunk) of the tree, with the result relying on the performance of the descriptor in each small computing unit.

Rather than emphasizing the local characteristics of trees, the purpose of this paper is to develop a new workflow with a focus on objective geometrical component proportions (See [7]) for automated detection of street trees in the urban road environment. The paper presents the results on street tree detection obtained using 3D point cloud data acquired by a RIGEL VMX-450 system.

2. OVERVIEW

This paper aims at deriving a street vegetation map layer, as well as single tree extraction, from point cloud data in an efficient way. The process of our method is shown as follows.

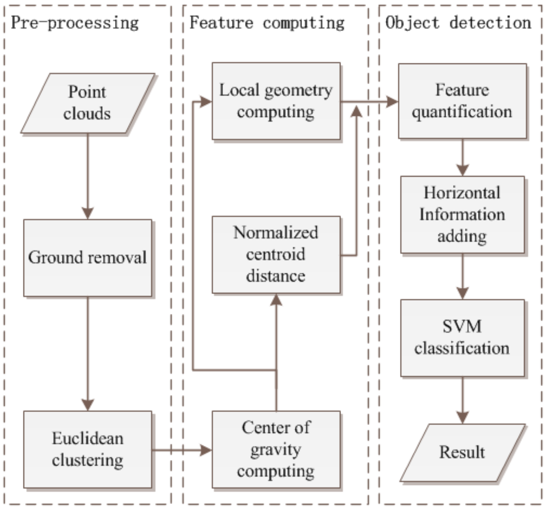


Fig.1. Workflow of street tree extraction.

In the pre-processing stage, we firstly segment the objects located above the ground by ground removal and Euclidean distance clustering so that the canopy and tree branches are not easily separated.

In the feature computing stage, we calculate the normalized centroid distance and the local geometrical features of off-ground objects. Then a feature criterion is derived and each point judged according to corresponding geometrical behaviors.

The street tree detection includes feature quantity and object recognition. We carry out the geometry component statistics proportion for each object. Then, the Support Vector Machine (SVM) is adopted as a classifier. Besides, the horizontal distribution information of an object is added to help detect street trees.

3. METHOD

3.1. Pre-processing

3.1.1. Ground removal

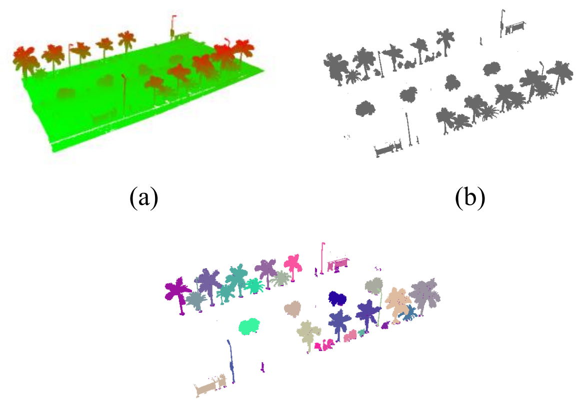
Computing the dimensional feature of all points acquired by the mobile LIDAR system requires a large amount of calculation. To reduce the amount of calculation, we adopt the ground removal algorithm in [8] to separate ground points and off-ground points.

Essentially, the ground removal algorithm is a region growing judging criterion. The data is subdivided into voxels as the basic calculation unit. The voxel, whose z value lies below an assigned ground threshold, $Z_{ground}=10\ m$, is considered as a suspected ground voxel. Then, we grow the suspected ground voxel upward to a height greater than $H_{off\text{-}ground}=1.6$ m. The growth rule judges the connectivity of the nine neighboring voxels above a query voxel. If the voxel fails to touch $H_{off\text{-}ground}$, the points which are included in this voxel are regarded as ground points and further removed. As shown in Fig. 2(b), ground points are removed and off-ground points are reserved for further clustering.

3.1.2. Clustering

After ground removal, Euclidean distance clustering is adopted to cluster points into individual objects from unorganized point clouds. Considering the distance between two objects, we use a search radius, $d=0.2\ m$, to implement clustering. After clustering, those objects consisting only of a few points are not regarded as trees and further abandoned. Due to the error presented by the ground filter, points on the bottom of each object under $h=0.1\ m$ are removed.

Clustering operation achieves the connection between the trunk and canopy of a street tree; thus, parts of street trees need not be further located. As shown in Fig. 2(c), the off-ground objects are successfully labeled by color coding.



(a) (b)

(c)

Fig.2. Illustration of ground removal: (a) raw data, (b) off-ground points, and (c) labelled off-ground objects.

3.2. Features extraction

Considering the different geometric detail and structural complexity of objects on point clouds, we improve the adaptation of the local geometrical descriptor. A saliency geometrical feature to describe the local point distribution by using principal component analysis was adopted in [9]. Let the covariance matrix of one 3D spatial point set be:

$$C=\frac{1}{n}\sum_{i=1}^{n}(X_i-\overline{X})(X_i-\overline{X})^T \tag{1}$$

where $X_i=(x_i,y_i,z_i)$ and $i\in[1,n]$; n represents the number of points around the query point. $\overline{X}=\frac{1}{n}\sum_{i=1}^{n}X_i$ indicates the center of gravity of the neighboring point set, which is additionally helpful to our algorithm. Then the covariance matrix, *C,* is decomposed to calculate the three eigenvalues $(\lambda_0,\lambda_1,\lambda_2)$, where $\lambda_0\geq\lambda_1\geq\lambda_2$. For street trees in point clouds, the volumetric structure is produced when the laser through the canopy of vegetation and the eigenvalues of inside points are followed with $\lambda_0 \approx \lambda_1 \approx \lambda_2$.

In fact, some quantitative criteria have been designed to measure local geometrical features based on eigenvalues of the covariance matrix. In [10], the artificial neural network was adopted to train and classify raw points into planar, linear or scatter points. Particularly, [7] presented a simple judging formula to discriminate those three dimensional behaviors as follows:

$$V_q = \operatorname{argmax}((\sigma_0 - \sigma_1)_N, (\sigma_1 - \sigma_2)_N, \sigma_{2N}) \tag{2}$$

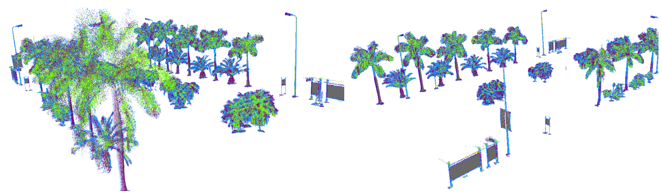
where V_q indicates the local geometry behaviour of the query point; the subscript N means the normalization, and $\sigma_i = \sqrt{\lambda_i}$. In addition, one salience feature, which calculates the Euclidean distance between the query point and the center of gravity of a neighboring point set for point cloud registration, was proposed in [11]. Inspired by those above methods, we decided to use the normalized centroid distance to subdivide three dimensional features. In this paper, we distinguish the local geometrical behaviors of a single point as five categories containing linear, edge, planar, cambered and scatter points. Particularly, the normalized centroid distance, *d*, is defined as the ratio of the distance between the center of gravity of a neighboring point set and the query point, q , to the searching radius *R* as follow:

$$d = \left\| X_q - \overline{X} \right\|_2 / R \tag{3}$$

where $X_q = (x_q, y_q, z_q)$ denotes the Cartesian coordinate of the query point. This equation measures the location of the query point in its neighborhood. If the query point is far removed from the neighborhood center, it is possibly an edge or convex behavior with a large value of *d*.

After computing the normalized centroid distance, *d*, of each point neighborhood, the descriptor is rebuilt as a vector $(\lambda_0, \lambda_1, \lambda_2, d)$. In the case of the situation that dimensional features are sensitive to the edge of an object, we prefer λ rather than σ and classify the point as V_q based on the above judging formula. To adapt to different geometric behaviors, the threshold d_L =0.15 and d_P =0.2 are empirically set to subdivide linear points into linear and edge points, planar points into planar and cambered points, respectively.

In general, it is intuitively plausible that tree objects contain more scatter and cambered structures than man-made objects, which have a neat body structure. As shown in Fig. 4, our proposed descriptor describes the street trees in greater geometric detail. Moreover, the scatter points (green) are distributed mainly in crown as shown in Fig. 3 (a). On the contrary, as shown in Fig. 3 (b), bus stops are presented as facades that retain a large number of planar points (grey) and fewer cambered points (purple). The edge points of objects are correctly detected and colored blue; linear points are colored yellow and distributed in the slender linear structure only.



(a) (b)

Fig.3. Illustrations of (a) and (b) local geometrical feature distributions on various objects

3.3. Tree detection

For feature quantization of objects, we firstly use the above criteria to determine the geometrical categories of each off-ground point. Then, considering the strong stability of the overall composition, the proportion of five geometrical behaviors of one object are counted as a feature vector. Thus, one five-dimensional vector is matched with one unidentified object. Moreover, the shapes of street trees vary with changing species. To span the difference in the wide gap of various street trees, two principal direction standard deviations of object horizontal projection (indicating the horizontal distribution of objects) are added as auxiliary features.

For classifying off-ground objects, the SVM algorithm is adopted as a classifier to recognize the objects as street trees and non-tree objects, respectively. LibSVM [12], the algorithm implementation platform, is applied with RBF kernel function and select appropriate parameters in C++ codes. Moreover, the ground truth labeling and feature normalization have been made before using SVM.

4. RESULTS AND DISCUSSION

The experimental datasets used in this paper were acquired by a RIGEL VMX-450 system. The location of the point cloud data we adopted was acquired on the Island Ring Road in the City of Xiamen, in which there are a variety of tropical trees on both sides of the road. The diversity and arrangement of the street trees is a challenge for automated detection. We used half of the data as the training dataset, the other half as the test dataset to verify the results. The detection results are summarized in Table I.

In our test dataset, the 186 clean objects and fifteen overlapping objects were extracted. Two classes of clean objects containing 112 street trees and 74 non-tree objects were manually labeled before classifying. The true positives (TP) are the number of tree objects that are correctly detected; the false positives (FP) are the number of missed

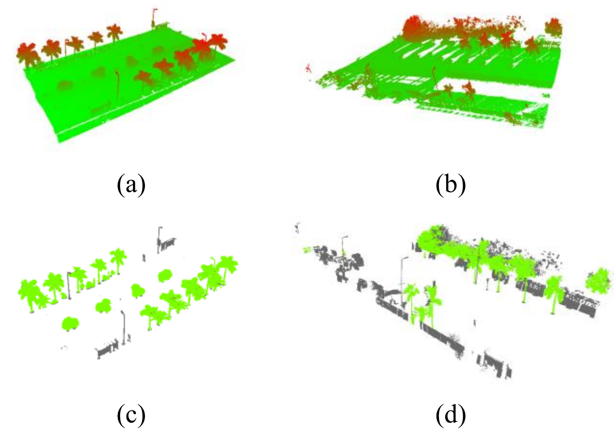


Fig.4. (a) and (b) two test datasets, (c) and (d) detected street trees.

trees after detection. Similarly, true negatives (TN) and false negatives (FN) measure the corresponding results of non-tree objects.

TABLE I. Detected result of street trees

Clean				Overlapping	F1-score
TP	FP	TN	FN		
109	3	63	11	15	89.53%

In Table I, overlapping means that, within a cluster, the street tree targets overlap non-tree targets. Hence, a separate result discussion is required for the overlapping object. The result of detecting overlapping objects as trees (quantity *n*) adds to both the number of true positives (TP) and false negatives (FN). Particularly, the results show that, the fifteen overlapping objects were all detected as trees, so *n*=15. And the evaluation equation for the entire data was built as follow.

$$\begin{aligned} &\text{Precision: } P = (TP+n)/(TP+n+FP) \\ &\text{Recall: } R = (TP+n)/(TP+FN+2n) \\ &\text{F1-score: } F = 2PR/(P+R) \end{aligned} \tag{4}$$

Finally, the precision (*P*) of 97.64%, the recall (*R*) of 82.67% and the F1-score (*F*) of 89.53% were obtained from our test data set. The typical visual results are shown in Figs. 4. The street trees (shown in green in Fig. 4(c) and (d)) were effectively extracted from the point cloud data. Conversely, road surface, buildings, or cars (presented in grey) were separated.

5. CONCLUDING REMARKS

Extraction and detection of street trees on point clouds acquired by mobile laser scanning systems facilitate the digital management and inventory of urban street trees. We improve the dimensional features to further subdivide for classifying through increasing the geometric detail and discrimination to the street trees. Besides, the geometry proportion feature was developed to adapt to the diversity and variety of street tree shapes. The results of our proposed method show the feasibility of our method to detect street trees. However, it is difficult to separate the objects when the street trees and non-tree objects are too close. In addition, some street trees miss their structural features because of withering leaves or incomplete scanning. In future work, the explicit segmentation of objects overlapping with street trees will be further studied for more complicated scenes.

REFERENCES

[1] C. Iovan, D. Boldo, and M. Cord, “Detection, characterization, and modeling vegetation in urban areas from high-resolution aerial imagery”, *IEEE J. Sel. Top. Appl. Earth Obs*. Remote Sens., vol.1, no. 3, pp. 206–213, 2008.

[2] A. Bienert, S. Scheller, and E. Keane, et al, “Tree detection and diameter estimations by analysis of forest terrestrial laser scanner point clouds,” *ISPRS Workshop on Laser Scanning and SilviLaser 2007*, Espoo, Finland, pp. 50-55, 2007.

[3] H. Gross, B. Jutzi, and U. Thoennessen, “Segmentation of tree regions using data of a full-waveform laser,” *International archives of photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(part 3), W49A, 2007.

[4] F. Monnier, B. Vallet, and B. Soheilian, “Trees detection from laser point clouds acquired in dense urban areas by a mobile mapping system,” *ISPRS Annals*, vol. I-3, pp. 245-250, 2012.

[5] R. Zhong, J. Wei, and W. Su, et al, “A method for extracting trees from vehicle-borne laser scanning data,” *Math. Comput. Modell.* , vol. 58, no.3, pp.733-742, 2013.

[6] Y. Yu, J. Li, and J. Yu, et al, “Pairwise three-dimensional shape context for partial object matching and retrieval on mobile laser scanning data,” *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 5, pp. 1019-1023, 2014.

[7] J. Demantké, C. Mallet, and N. David, et al, “Dimensionality based scale selection in 3D lidar point clouds,” *ISPRS Archives*, 38(5/W12), pp. 97-102, 2011.

[8] Y. Yu, J. Li, and H. Guan, et al, “Semiautomated extraction of street light poles from mobile LiDAR point-clouds”, *IEEE Trans. Geosci. Remote Sens.*, vol.53, no.3, pp.1374-1386, 2015.

[9] J. F. Lalonde, N. Vandapel, and D. F. Huber, et al, “Natural terrain classification using three - dimensional ladar data for ground robot mobility,” *J. Field Rob.*, vol. 23, no. 10, pp. 839-861, 2006.

[10] R. Zhang, G. Li, and M. Li, et al, “Classification of LIDAR point clouds based on PCA-BP algorithm,” *Bulletin of Surveying and Mapping*, no. 7, pp. 23-26, 2014.

[11] C. Basdogan, and A. C. Oztireli, “A new feature-based method for robust and efficient rigid-body registration of overlapping point clouds,” *The Visual Computer*, vol. 24, pp. 679-688, 2008.

[12] C. C. Chang, and C. J. Lin, “LIBSVM: a library for support vector machines,” *ACM Trans. Intell. Syst. Technol.*, vol.2, no.3, pp. 1-27, 2011.