Rapid Localization and Extraction of Street Light Poles in Mobile LiDAR Point Clouds: A Supervoxel-Based Approach

Fan Wu, Chenglu Wen, *Member, IEEE*, Yulan Guo, *Member, IEEE*, Jingjing Wang, Yongtao Yu, Cheng Wang, *Senior Member, IEEE*, and Jonathan Li, *Senior Member, IEEE*

***Abstract*—This paper presents a supervoxel-based approach for automated localization and extraction of street light poles in point clouds acquired by a mobile LiDAR system. The method consists of five steps: preprocessing, localization, segmentation, feature extraction, and classification. First, the raw point clouds are divided into segments along the trajectory, the ground points are removed, and the remaining points are segmented into supervoxels. Then, a robust localization method is proposed to accurately identify the pole-like objects. Next, a localization-guided segmentation method is proposed to obtain pole-like objects. Subsequently, the pole features are classified using the support vector machine and random forests. The proposed approach was evaluated on three datasets with 1,055 street light poles and 701 million points. Experimental results show that our localization method achieved an average recall value of 98.8%. A comparative study proved that our method is more robust and efficient than other existing methods for localization and extraction of street light poles.**

***Index Terms*—Point clouds, supervoxel, localization, segmentation, classification.**

I. Introduction

STREET light poles play an important role in transportation systems as they provide light for pedestrians, vehicles and road surveillance cameras during the night [1]. They can significantly reduce the frequency of nighttime crashes, especially for roadway intersections [2]. Geospatial information of street light poles has been used in many applications including urban planning, environmental impact assessment, Intelligent Transportation Systems (ITS), disaster management [3]. For ITS related applications, the information of street light poles can be used for road infrastructure maintenance, road safety analyses, advanced driver assistance, semantic mapping, and smart city applications. For instance, for a driver assistance system, the position of street light poles can be used to improve the stability of road tracking [4]. Therefore, regular inventory and maintenance of street light poles is important. Due to the large number of street light poles on road, traditional manual survey methods are extremely time-consuming. A rapid and robust method is highly needed to obtain street light pole information.

In recent decades, several methods have been proposed to extract street light poles from images [5]. However, image based methods have several significant limitations, including the lack of geospatial and reflectivity intensity information, the image distortions caused by camera lens, and the image change due to illumination variations. Consequently, traditional optical imaging-based systems might not be able to understand a road scene correctly in a poor condition.

Recently, mobile LiDAR systems have been rapidly developed. 3D points and reflectivity intensity of road scenes can be acquired by LiDAR sensors equipped on a vehicle [6]. Compared to optical imaging-based systems, mobile LiDAR systems can obtain accurate geospatial and reflectivity intensity information. Besides, they are more robust to illumination variations and image distortions. Compared to airborne LiDAR systems, mobile LiDAR systems have a better view of steep terrain and the sides of structures. Moreover, the density of point clouds acquired by mobile LiDAR systems is also higher than airborne LiDAR systems. Therefore, mobile LiDAR systems are more suitable for road scene understanding and have been used in several ITS applications [7]–[10]. Due to these reasons, extraction of street light poles in mobile LiDAR point clouds has attracted increasing number of attention in the last few years [3], [11]–[13]. An example of street light pole extraction in point clouds is shown in Fig. 1.

Point cloud segmentation is a key step for street light pole extraction in point clouds. Nowadays, although several point cloud segmentation methods for pole-like objects have been

Manuscript received June 21, 2015; revised October 9, 2015, January 29, 2016, and April 6, 2016; accepted May 7, 2016. Date of publication June 22, 2016; date of current version February 1, 2017. This work was supported in part by the Natural Science Foundation of China under Grants 61401382, 41471379, and 61471371. The Associate Editor for this paper was F.-Y. Wang. *(Corresponding authors: Chenglu Wen and Jonathan Li.)*

F. Wu, C. Wen, and C. Wang are with Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen 361005, China, and also with the Fujian Collaborative Innovation Center for Big Data Applications in Governments, Fuzhou 350003, China (e-mail: clwen@xmu.edu.cn).

Y. Guo is with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China.

J. Wang is with the Data Mining Group, School of Information Science and Engineering, Xiamen University, Xiamen 361005, China.

Y. Yu is with the Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian 223003, China.

J. Li is with the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen 361005, China; with Fujian Collaborative Innovation Center for Big Data Applications in Governments, Fuzhou 350003, China; and also with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: junli@xmu.edu.cn; junli@uwaterloo.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2016.2565698

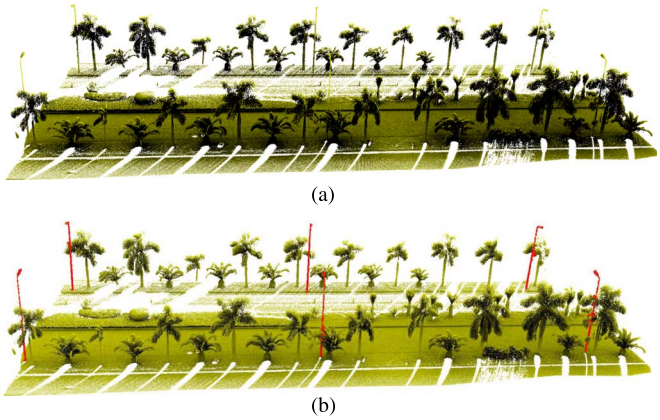


Fig. 1. Example of street light pole extraction in point clouds. (a) Input point cloud. (b) Street light pole extraction results in point clouds.

proposed [3], [11]–[15], accurate and rapid pole-like object segmentation in complex scenes is still an unsolved problem. Both under-segmentation and over-segmentation will deteriorate the performance of existing street light pole extraction methods for point clouds [3], [11]–[13].

In this paper, we propose a novel guided approach for automatic street light pole extraction from 3D point clouds acquired by a mobile LiDAR system. The ground points are first removed and non-ground points are then segmented into supervoxels. Once the localization of pole-like objects is obtained, an efficient prior information guided segmentation method is proposed to acquire pole-like objects. Next, two types of features are generated to describe the pole-like objects. Finally, Support Vector Machine (SVM) and random forests are employed for object classification. The contributions of this paper are three-folds:

- We propose an efficient, robust and automatic method to locate pole-like objects. It consists of three parts. First, the initial localization map is generated by projecting points on to a plane with a designed function. A "ball falling" algorithm is then proposed to improve the robustness of localization with respect to point density variations. Finally, the accurate positions of pole-like objects is detected. Our localization method achieves good performance on scenes with occlusion and density variations.
- We propose a guided segmentation method. Using the prior information of localization, the supervoxels close to a detected position are more likely to be determined as a pole-like object located at that position. The segmentation method is also robust to occlusion, missing data, density variations.
- We propose a street light pole extraction method using SVM and random forests. The two classifiers depend on two types of features obtained from guided segmentation results. The method is also robust to occlusion.

The remainder of this paper is organized as follow. Section II introduces the related work on street light pole extraction and patches generation in point clouds. Section III describes our method. Section IV evaluates our method and analyses the experiment results. Section V gives the concluding results.

II. Related Work

A. Labeling of Street Light Poles in Images and Point Clouds

Abdalla *et al.* [5] used a sequence of image filters and a cross correlation operation to extract street light poles in images. However, the method is not fully automatic.

Existing methods for street light pole extraction in point clouds are mainly based on shape features [11], [12], prior knowledge [13], and shape template matching [3]. A segmentation method was presented to extract street light poles using k-nearest neighbors, Principal Component Analysis (PCA), and contextual/shape features [11]. Eigenvalue analysis and linear features were also used to extract street light poles [12]. The methods in [11], [12] used k-nearest neighbors to cluster points, which is not robust to occlusion. Hu *et al.* [13] used the Density of Projection Points (DoPP) to extract street light poles based on the heights of street light poles. The method has to manually determine an optimal height threshold. The Normalized Cut (Ncut) [16], [17] and the shape template matching methods were also proposed [3] to extract street light poles.

Aforementioned approaches for street light pole extraction in mobile LiDAR point clouds are limited by various factors including occlusion, clutters, incomplete data, density variations of point clouds, and high computational complexities. Particularly, for scenes with occlusion and clutters [as shown in Fig. 1(a)], most existing methods are not sufficiently robust. That is because the performance of these street light pole extraction methods highly relies on their point cloud segmentation results. However, due to the lack of prior information, the segmentation results are usually inaccurate. As a result, feature description can be biased and classification performance will be declined. Consequently, their street light pole extraction performance in occluded scenes is unstable.

In summary, the major challenge for street light pole extraction in point clouds is how to improve the efficiency and robustness with occlusion, clutter, and incomplete data.

B. Studies on 3D Patch Extraction

Several existing point cloud processing algorithms are too time-consuming as the data are treated in a point-wise manner. In order to accelerate the point cloud processing algorithms, over-segmentation methods have been proposed [18]–[22]. These methods are similar to the superpixel approaches that have been widely used in image processing [23], [24]. Using an over-segmentation method, a point cloud can first be divided into a number of patches and the processing can then be operated in a patch-wise manner (rather than a point-wise manner). Since the number of patches is much smaller than the number of points in a point cloud, the efficiency of point cloud processing can be significantly improved. Consequently, patch is treated as the basic processing unit in this paper.

In [18], the partition of a point cloud was achieved by an octree [25]. Local patches were then extracted according to the leaves of the octree. The number of local patches is related to the number of whole points and the size of octree leaves. A major limitation of this method is that the interior shape structure is discarded. The methods [19]–[21] simply extended

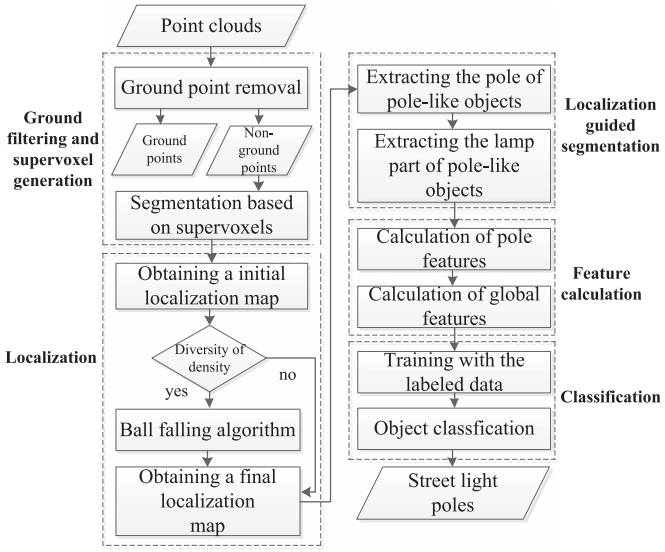


Fig. 2. Flowchart of the street light pole extraction pipeline.

the 2D superpixel segmentation algorithms to the domain of 3D volumes. These approaches cannot work on 3D point clouds because they are limited to data with a structured lattice (e.g., 2D images, 3D volumetric data). Papon *et al.* [22] proposed a Voxel Cloud Connectivity Segmentation (VCCS) method to take full advantage of 3D geometry information. The points with similar normals, colors, and Fast Point Feature Histograms (FPFHs) [26] are clustered into a supervoxel. Since VCCS can greatly reduce the point cloud processing time [27], it is used in our work for 3D patch extraction.

III. The Proposed Method

As shown in Fig. 2, our method consists of five main stages: (1) Ground points are removed and non-ground points are segmented into supervoxels (Section III-A); (2) The characters of street light poles are used to locate pole-like objects (Section III-B); (3) Pole-like objects are obtained by a guided segmentation method (Section III-C); (4) Two types of features are calculated for pole-like objects (Section III-D); and (5) street light poles are recognized by SVM and random forests (Section III-E).

The workflow of the proposed method is shown in Fig. 2 and the result for each step of the proposed approach is shown in Fig. 3.

A. Ground Filtering and Supervoxel Generation

1) Ground Filtering: First, the raw point clouds are vertically divided into segments along the trajectory of a vehicle to achieve high efficiency of the filtering process [9], [28]. Then, the Random Sample Consensus (RANSAC) algorithm [29] is applied to remove the points which are close to the ground points. The average height of ground, h_g, is estimated through initial plane fitting. In each RANSAC iteration, the plane function is obtained. Then, if the distance from an un-classified point to the plane is less than a predefined threshold d_t, this point is classified as ground points.

For each iteration, this process is repeated for all unclassified points. The iteration continues until the height of one point in the iteration point set is higher than $h_g + 1$ or the number of points generated by RANSAC remains unchanged. An illustration of the ground filtering results is shown in Fig. 3(b). Once the ground points are removed, the non-ground points with similar features are then segmented into a supervoxel through VCCS [22].

The points in each supervoxel are then projected onto the XY plane to obtain the area of the convex hull [30] with cross product. Next, several features are calculated for each supervoxel. These features include (1) the number of points; (2) the highest point; (3) the lowest point; (4) the barycenter of points; (5) the area of the convex hull of the projected points; (6) the bounding box. Each supervoxel is assigned a specific ID while the ID of an unclassified supervoxel is assigned to -1.

B. Localization

There are two distinctive characters for street light poles. (1) Street light poles are usually higher than other road facilities; (2) the area of cross section of a street light pole is nearly the same from bottom to up and is smaller than that of common urban trees. Inspired by these observations, a localization method is proposed based on the height and the 2D projection image of a point cloud. The localization method includes three parts, i.e., initial localization map generation, "ball falling," and position detection. The localization result is then used as prior information for segmentation. An illustration of the localization map generation process is shown in Fig. 4. For better visualization, the localization map is inverted to generate Figs. 4(d)–(f). That is, each pixel in Figs. 4(d)–(f) is calculated by subtracting the pixel value in the localization map from the maximum pixel value.

Initial localization map generation

(1) Grid Filtering: The raw point clouds are firstly divided into grids along the x and y axes [see Fig. 4(a) and (b)]. The maximum value along the z axis, z', in a grid is obtained. If z' is within a pre-defined range $(h_{\text{low}}, h_{\text{high}})$, the grid is kept. Otherwise, it is removed.

(2) Projection: Equation (1) is used to obtain $f(z)$ for point $p(x, y, z)$ in each remaining grid. The function $f(z)$ is defined as follows:

$$f(z) = \frac{1}{\exp(-(z - \frac{h_{\text{lamp}}}{2})) + 1} \tag{1}$$

where h_{lamp} is the height of the street light pole and can be obtained by manual measurement in point clouds. Then, the sum of $f(z)$ values for all the points $p(x, y, z)$ belonging to a grid is calculated. The maximum value of these sums in all grids is denoted by f'.

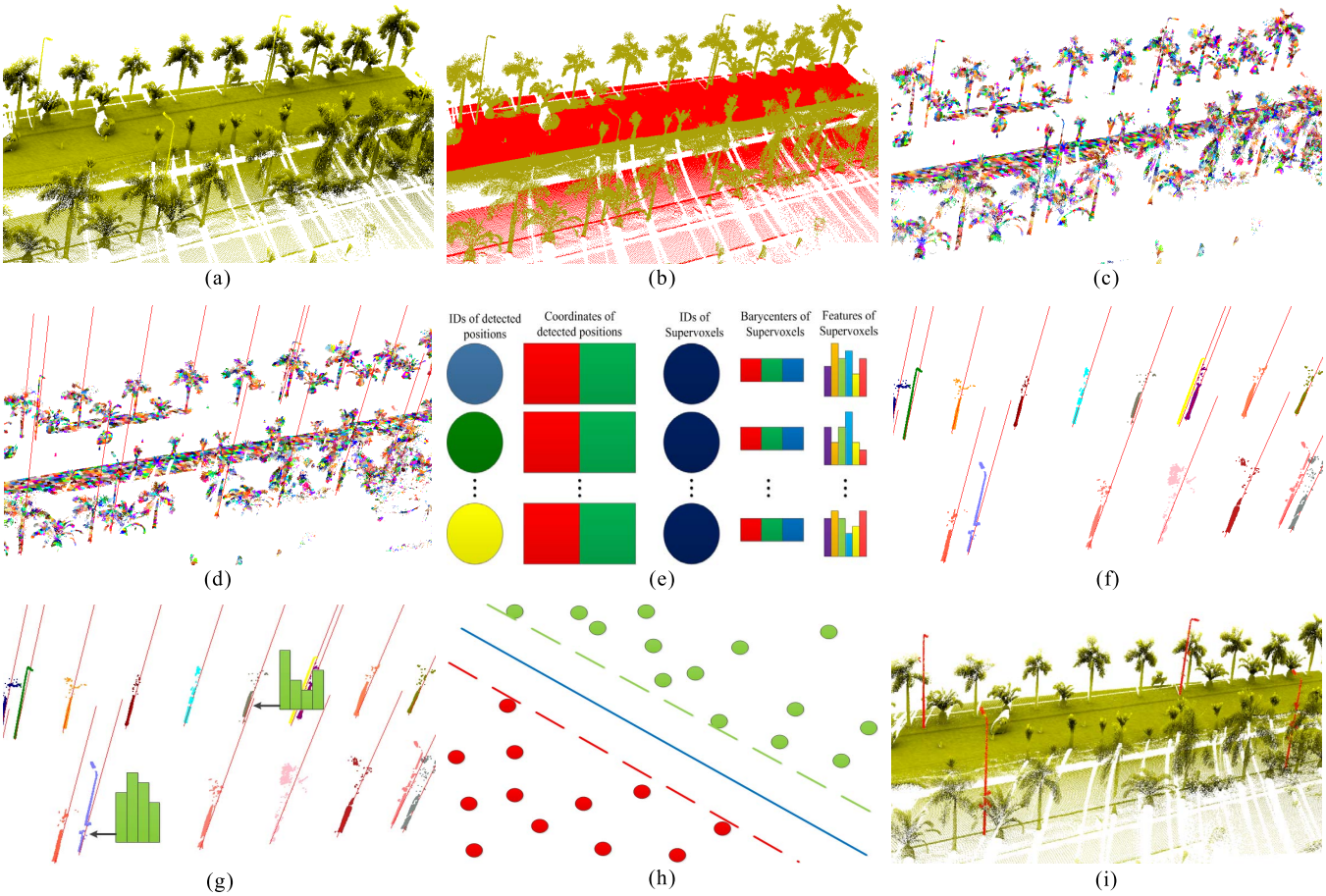


Fig. 3. Illustration of the results for each step of the street light pole extraction method. (a) Raw point clouds. (b) Filtered ground points. (c) Supervoxels of non-ground points. (d) Localization. (e) Obtained information about detected positions and supervoxels. (f) Segmentation. (g) Feature extraction. (h) Classifier traning. (i) Classification.

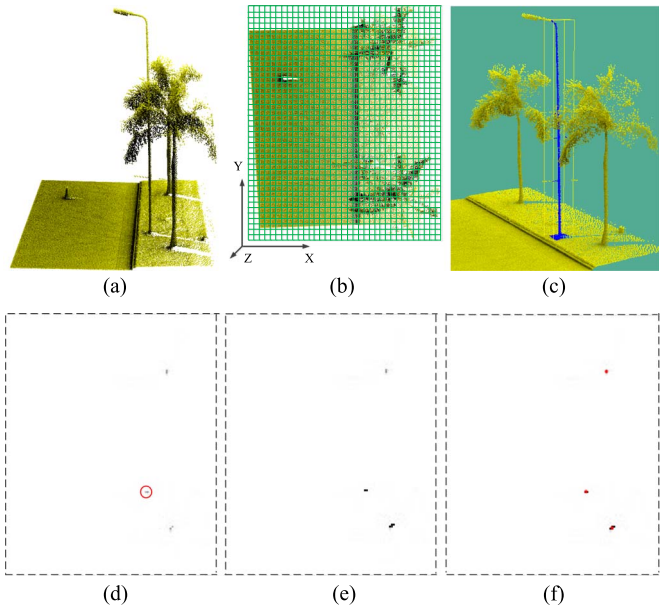


Fig. 4. Illustration of the localization map generation process. (a) Raw point clouds. (b) Gridded point clouds. (c) One of the grids. (d) Initial localization maps. Red circle denotes the pixel of the grid shown in figure (c). (e) Final localization map. (f) Positions of detected street light poles.

(3) Coping with density variations: It is obvious that f' is related to the density of point clouds. Considering a system with two laser scanners mounted on the top of a vehicle and worked towards two different directions [as shown in Fig. 9], if only one scanner is used, the density of the point cloud acquired from the left side will be different from that acquired from the right side. Then, f' has to be modified as $f' = \alpha \cdot f'$ (where $\alpha = 0.67$ in this paper). If two scanners are employed, the density of point clouds acquired from two sides will be almost the same, f' does not need to be modified in this case. It is obvious that the number of points for point clouds acquired from two scanners is about two times of that acquired from one scanner. Therefore, in order to reduce the number of points for processing, only the point cloud acquired from one scanner was used in our experiments.

(4) Initial localization map generation: When f' is obtained, the initial localization map F can be normalized by f'. Finally, F is replaced by the high-frequency components in F by the removal of low-frequency components in F [see Fig. 4(d)]. The process for initial localization map generation is shown in Algorithm 1. In Algorithm 1, g_k represents the k-th grid in G [see Fig. 4(c)] and f_k represents the pixel of g_k in F [shown in red circle in Fig. 4(d)].

Algorithm 1 Initial localization map generation

Input: A set of grids $g_k \subset G$. The number of point clouds, N_{scan}
Output: An initial localization map F.
1: $F \leftarrow \mathbf{0}$ //Localization map initialization
2: **for all** $g_k \subset G$ **do**
3: $z' \leftarrow \max_{p(x,y,z)\in g_k} z$
4: **if** $h_{\text{low}} \leq z' \leq h_{\text{high}}$ **then**
5: $f_k \leftarrow \sum_{p(x,y,z)\in g_k} f(z)$ //filtering grids by height
6: **end if**
7: **end for**
8: $f' \leftarrow \max_{f_k \in F} f_k$
9: **if** $N_{\text{scan}} = 1$ **then**
10: $f' \leftarrow \alpha \cdot f'$ // coping with density variations
11: **end if**
12: **for all** $f_k \in F$ **do**
13: $f_k \leftarrow \min(M \cdot \frac{f_k}{f'}, M)$ //M is the intensity of upper limit and equals 255
14: **end for**
15: $F \leftarrow F - Gaussian(F, r_g)$ //r_g is the radius of Gaussian blur function
16: return F

Due to the diversity in point density, the number of points for street light poles with a low point density is smaller than the number of those with a high density. Therefore, it is difficult to locate all street light poles in point clouds acquired from a laser scanner by a simple threshold without detecting few false positives. Consequently, a "ball falling" algorithm is proposed.

The ball falling algorithm

(1) Falling condition: In each grid, if the number of points in the grid, $N(g_k)$, is larger than a predefined value and the largest z in the grid, $H(g_k)$, is larger than a predefined value, a ball with the radius r_b is firstly placed at the highest position h_{ball} in grid g_k [see Fig. 5(a)], the ball then falls down to reach the ground in g_k. When the center of the ball reaches the ground, the ball falling process stops [see Fig. 5(d)].

(2) Seeking required supervoxels: During the ball falling process, if the number of barycenters of supervoxels S within the range of the ball radius [see Fig. 5(b)] is larger than a predefined value e_c and the area of convex hull of the projection points of these supervoxels is less than a predefined threshold s_t [see Fig. 5(c)], the grid is considered as the position of a potential street light pole.

(3) Adding corresponding pixels: Therefore, the pixel f_k of g_k should be added [see Fig. 4(e)]. The added pixel is related to the number of supervoxels found by the ball falling process. The details of "ball falling" algorithm is shown in Algorithm 2. Where, $N(\cdot)$ is the number of elements in a set, $H(\cdot)$ is the largest z value of the points in a grid.

Position detection

When the ball falling algorithm is performed for all grids, a pixel intensity threshold p_t is set to obtain connected areas [31]. Finally, each center of a connected area is regarded as a detected position [see Figs. 3(d) and 4(f)] and each detected position is assigned an unique ID [see Fig. 3(e)].

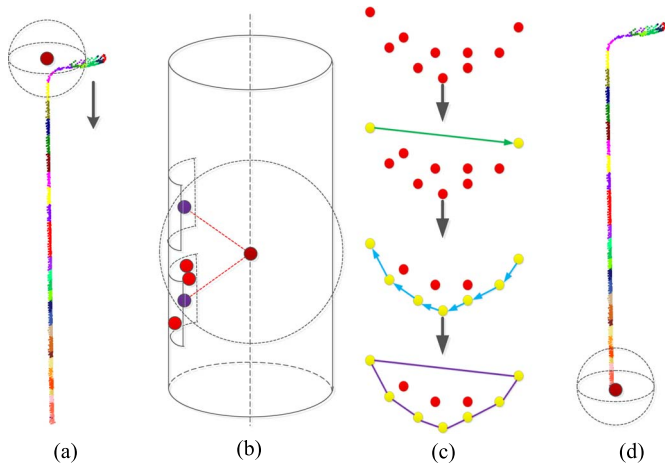


Fig. 5. Ball falling algorithm. (a) Initialization. (b) Seeking supervoxels and calculating average distance from the barycenters to the detected position. (c) Convex hull calculation. (d) End condition.

Algorithm 2 The "ball falling" algorithm

Input: The initial localization map F. The gridded point clouds G.
Output: A final localization map F.
1: **for all** $f_k \in F$ **do**
2: **if** $N(g_k) > n_g$ and $H(g_k) > h_g + h_o$ **then**
3: $num_e \leftarrow 0$
4: $sum_d \leftarrow 0$
5: $h_{ball} \leftarrow GetHighestPosition(g_k)$
6: **while** $h_{\text{ball}} > h_g$ **do**
7: $S \leftarrow GetLessAreaSupervoxel(h_{ball})$
8: **if** $N(S) \geq e_c$ **then**
9: $sum_d \leftarrow sum_d + GetAverageDistance(S)$
10: $num_e \leftarrow num_e + 1$
11: **end if**
12: $h_{\text{ball}} \leftarrow h_{\text{ball}} - 1$
13: **end while**
14: $f_{\text{add}} \leftarrow 0$
15: **if** $num_e > n_f$ **then**
16: $f_{\text{add}} \leftarrow (1 - \frac{sum_d}{num_e \cdot r_b}) \cdot num_e \cdot p_t$
17: **end if**
18: $f_k \leftarrow \min(f_k + f_{\text{add}}, M)$
19: **end if**
20: **end for**
21: return F

C. Localization Guided Segmentation

There are two steps in our segmentation methods. Supervoxel classification in the two steps is performed in descending order of the intensity of the corresponding pixel in F of detected positions [see Fig. 4(e)].

Guided segmentation

(1) Extracting the pole of a pole-like object

The first step is to obtain the pole of pole-like objects. According to the second character of street light poles, three conditions are designed to classify supervoxels.

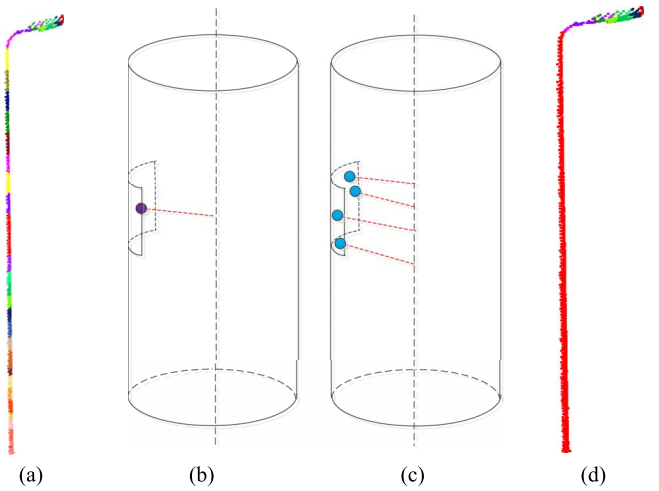


Fig. 6. Pole extraction. (a) Supervoxels of polelike objects. (b) Distance between the barycenter of a supervoxel and a detected position. (c) Distance between a point of a supervoxel and a detected position. (d) Extraction result.

Barycenter condition: It is common that street light poles are vertical to the ground and the z coordinates of ground points in a grid are almost the same. Therefore, the distance from the gravity center g of a supervoxel s to a detected position l_k, can be calculated:

$$dis(g, l_k) = \sqrt{(x_g - x_{l_k})^2 + (y_g - y_{l_k})^2} \tag{2}$$

(x_g, y_g) and (x_{l_k}, y_{l_k}) are the plane coordinates of g and l_k, respectively. $dis(g, l_k)$ should be less than d_g [see Fig. 6(b)].

Ratio condition: The distance from a point p_s in s to l_k can also be obtained by [Eq. (2)] and it should be less than a predefined value d_{in} [see Fig. 6(c)]. The ratio of points satisfying this distance requirement should be more than λ.

Overall condition: A set of supervoxels s_k satisfying above two conditions can be acquired. If the number of supervoxels in s_k, $N(s_k)$, is more than a predefined value n_l and the number of points in s_k, $N_p(s_k)$, is more than a predefined value n_p, the ID of l_k is assigned to the supervoxels in s_k [see Fig. 6(d)].

(2) Extracting the lamp part of a pole-like objects

This step is to obtain the lamp part of a street light pole. It is observed that if an unclassified supervoxel is close to and over the peak of the pole obtained in step 1, and the supervoxel is within the range of the position of the pole, the supervoxel is likely to belong to the pole. Motivated by this observation, step 2 is presented as follows.

Adding initial seed supervoxels: First, the peak point $p_{\mathrm{pe}}(x_{\mathrm{pe}}, y_{\mathrm{pe}}, z_{\mathrm{pe}})$ can be obtained for some classified supervoxels, s_k, at a detected position l_k. Next, the supervoxels s_{seed} whose barycenters are within a distance of r_e from p_{pe} can be obtained [see Fig. 7(a)]. s_{seed} are then added into a queue Q in the order of ascending distances between their barycenters to p_{pe}. The element on the top of the queue is considered as a seed supervoxel u and is removed from Q.

Expanding: Then, the neighboring unclassified supervoxels whose barycenters are within the distance of r_e from the barycenter of the seed supervoxel can be obtained [see Fig. 7(b)]. There are three conditions for adding a neighboring supervoxel s_n of a seed supervoxel into the queue. (1) The z coordinate of the highest point in the seed supervoxel is less than the z coordinate of the lowest point in s_n plus l_g [see Fig. 7(c)]. (2) The z coordinate of the highest point in s_n should be less than the estimated ground coordinate h_g plus h_{lamp} [see Fig. 7(d)]. (3) Eight corner points of the bounding box of s_n can be obtained. Afterwards, the largest distances from these points to the position l_k [calculated by Eq. (2)] should be less than d_s [see Fig. 7(e)].

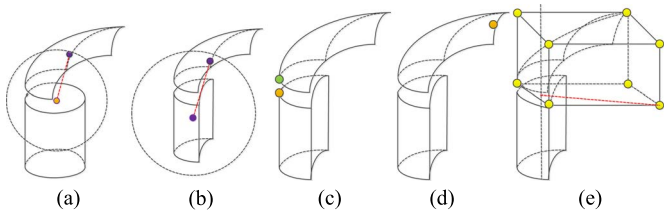


Fig. 7. Process for extracting lamp part of a street light pole. (a) Adding initial seed supervoxels. (b) Seeking neighbor supervoxels. (c) Relationship with height. (d) Height limit. (e) Offset distance.

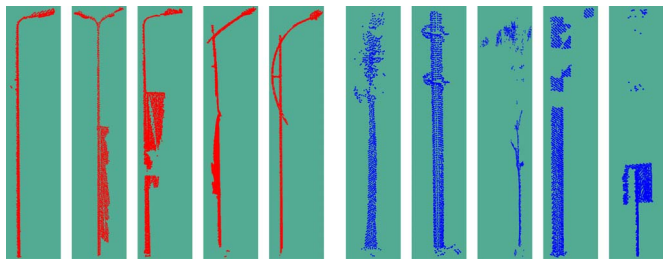


Fig. 8. Training examples. (a) Positive training examples. (b) Negative training examples.

Supervoxel classification: The neighboring unclassified supervoxels s' satisfying the three conditions are classified into s_k.

Terminal condition: Then, the first element of the queue becomes a new seed supervoxel. This process continues until the queue is empty. Each detected positions are checked by the guided segmentation. The segmentation process is shown in Algorithm 3. The segmentation result is shown in Fig. 3(f). Next, the highest point $p_{\mathrm{su}}(x_{\mathrm{su}}, y_{\mathrm{su}}, z_{\mathrm{su}})$ of an object can be obtained. Then, the vector v_h is calculated by $v_h = (x_{\mathrm{su}} - x_{\mathrm{pe}}, y_{\mathrm{su}} - y_{\mathrm{pe}}, z_{\mathrm{su}} - z_{\mathrm{pe}})$. Finally, the angle between the vector v_h and the ground vector $(0, 0, 1)$ is calculated and further used for classification.

D. Feature Extraction

It is demonstrated by several existing feature descriptors that improved performance can be achieved by integrating local point features and global geometry features [8]. The distinctive and robust performance of point-wise features is usually lower than that of object part based features. Therefore, we empirically designed two types of features to represent a street light pole: pole features and global features. The global features include the information of the street light lamp but the pole features do not. Since the upper parts of street light poles and

Fig. 9. Experimental setup for point cloud acquisition. (a) RIEGL VMX-450 system. (b) System configurations. (c) Trajectory of point cloud acquisition (presented by yellow lines).

trees are obviously different from their poles, the combination of pole features and global features is sufficient to discriminate street light poles from trees.

Algorithm 3 Location guided segmentation

Input: A set of detected positions L. A set of unclassified supervoxels, S.
Output: A set of classified supervoxels S.
for all $l_k \in L$ **do**
  $s_k \leftarrow FindRangeSupervoxel(l_k, d_g)$
  $s_k \leftarrow RatioNear(s_k, l_k, d_{in}, \lambda)$
  if $N(s_k) > n_l$ and $N_p(s_k) > n_p$ **then**
    $AssignID(s_k, k)$ //k is assigned to the ID of s_k
  end if
end for
for all $l_k \in L$ **do**
  $Q \leftarrow \emptyset$
  $p_{\text{pe}} \leftarrow Peak(s_k)$
  $s_{\text{seed}} \leftarrow FindRangeSort(p_{\text{pe}}, r_e)$
  $Enqueue(Q, s_{\text{seed}})$
  while $Q \neq \emptyset$ **do**
    $u \leftarrow Dequeue(Q)$
    $s' \leftarrow FindRequiredGrowing(l_k, u, r_e, l_d, d_s)$
    $Enqueue(Q, s')$
    $s_k \leftarrow s_k \cup s'$ //k is assigned to the ID of s'
  end while
end for

The pole features also can be used to describe the features of other pole-like objects, such as traffic signs and telegraph poles. It means that using pole features only is insufficient to discriminate street light poles from other pole-like objects with the same size. However, since our feature description method uses both pole features and global features, it is able to discriminate street light pole from other pole-like objects as they have different global shapes.

Pole features. Once the first step of segmentation is completed, the following nine features are computed to describe the pole: (1) the height; (2) the average height; (3) the standard deviation of height; (4) the average area of the convex hull for the projected points in a supervoxel; (5) the standard deviation of the area of the convex hull for projected points in a supervoxel; (6) the area of the convex hull for the projected points of the whole object; (7) the estimated volume; (8) the number of pole points; and (9) the number of the supervoxels whose area of the convex hull for their projected points is less than s_t, where s_t is the standard area of the convex hull for the projected points of a street light pole.

TABLE I
STATISTICS AND GROUND TRUTH OF THE DATASET

	Statistics				Ground Truth	
Dataset	Type	Size (GB)	Length (km)	#Points (million)	#Light (Poles)	Overlapping
RRS	Urban	15.8	1.3	403	644	435
ICEC	Urban	9.3	7.2	220	271	180
YDT	Suburb	2.94	2.0	78	140	61

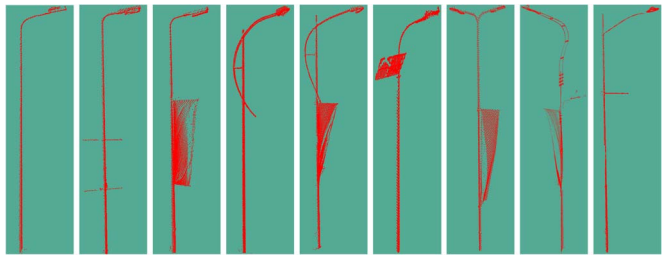


Fig. 10. Illustration of nine types of street light poles presented in our test datasets.

Global features. Once the second step of segmentation is completed, the following ten features are calculated to describe the whole object: (1) the height; (2) the average height; (3) the standard deviation of height; (4) the pixel intensity in the corresponding location map (Section III-B); (5) the area of the convex hull for all projected points; (6) the estimated volume; (7) the height difference between the barycenter and geometry center; (8) the number of points; (9) the number of neighboring supervoxels with barycenters within a distance of

TABLE II
PARAMETERS USED IN THE PREPROCESSING AND LOCALIZATION STAGE AND LOCALIZATION RESULTS

	Parameters Used in the Preprocessing Stage			Parameters Used in the Localization Stage										Quantitative Evaluation of Localization	
Dataset	d_t	d_c	r_s	h_{high}	h_{low}	h_{lamp}	r_g	s_t	n_g	h_o	p_t	n_f	r_b	e_c	#Ground Truth	#Location Detected(%)
RSR	0.03	0.1	0.05	13	6	12	3	0.05	80	7	30	3	1.1	2	644	639 (99.2%)
ICEC	0.03	0.1	0.05	13	6	12	3	0.02	80	5	30	5	1.1	2	271	266 (98.1%)
YDT	0.03	0.1	0.05	13	6	11	3	0.02	80	4	30	5	1.1	2	140	139 (99.2%)

TABLE III
PARAMETERS USED IN SEGMENTATION

Dataset	d_g	d_{in}	λ	n_l	n_p	r_e	l_g	d_s
RSR	1.0	0.5	0.5	5	50	1.8	0.1	1.8
ICEC	1.0	0.5	0.5	5	50	2.6	0.1	2.6
YDT	1.0	0.5	0.5	5	50	2.8	0.1	2.8

TABLE IV
EVALUATION RESULT ACHIEVED BY OUR SVM-BASED METHOD

Dataset		recall	precision	F1-measure
RRS	mean	94.6%	97.5%	96.0%
	std	1.2×10^{-3}	7.7×10^{-4}	7.9×10^{-4}
ICEC	mean	95.9%	95.0%	95.4%
	std	2.2×10^{-3}	2.3×10^{-3}	1.8×10^{-3}
YDT	mean	98.3%	98.6%	98.4%
	std	3.5×10^{-3}	5.0×10^{-5}	1.8×10^{-3}

TABLE V
EVALUATION RESULT ACHIEVED BY OUR RANDOM FOREST-BASED METHOD

Dataset		recall	precision	F1-measure
RRS	mean	94.6%	99.2%	96.9%
	std	2.1×10^{-3}	2.3×10^{-3}	9.4×10^{-4}
ICEC	mean	95.2%	98.9%	97.0%
	std	3.7×10^{-3}	1.5×10^{-3}	1.7×10^{-3}
YDT	mean	97.9%	99.4%	98.6%
	std	1.1×10^{-16}	5.9×10^{-3}	2.9×10^{-3}

1 m from p_{pe}; and (10) the angle between the vector v_h and the vector $(0, 0, 1)$.

E. Classification

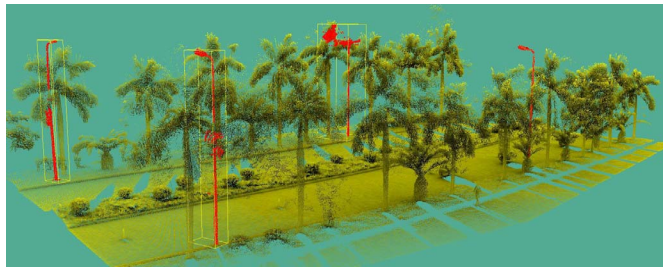
In the final stage, the street light poles are extracted by classifying all the pole-like objects. The feature vectors have to be normalized before classification. Both SVM and random forests in the weka tool kit [32] are employed to classify objects. SVM and random forests are trained with manually labeled objects. Several training examples in the three datasets are shown in Fig. 8. Finally, the trained classifiers are used to classify the test datasets.

IV. RESULTS AND DISCUSSION

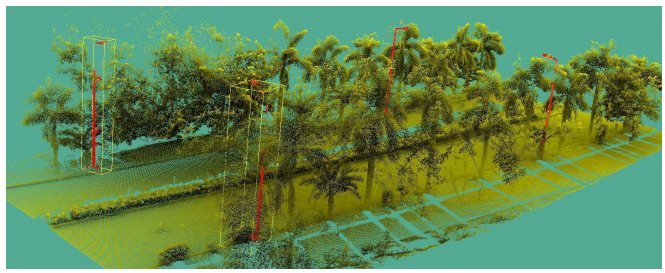
A. Mobile LiDAR Datasets

The mobile LiDAR point cloud data were acquired by a RIEGL VMX-450 system mounted on a vehicle [as shown in Fig. 9(a)]. The vehicle was drove at a speed of 40–50 km/h in the City of Xiamen during point cloud acquisition. The data collection process started from the Siming Campus of Xiamen University (XMU), then to the International Conference and Exhibition Center (ICEC) and arrived at the Yun Ding Tunnel (YDT) [as shown in Fig. 9(c)]. Three different scenes were selected for our experimental studies.

Scene 1. The first dataset was collected from the Ring Road South (RRS). The RRS road is a typical urban road with four lanes in two directions. Two types of trees were located next to each other with a distance of about 4.08 m. Therefore, the street light poles can be easily occluded by their neighboring trees [see Fig. 12(a)]. In the RRS dataset, more than two thirds of street light poles are occluded by trees.



(a)



(b)

Fig. 11. Failed cases for street light pole detection. (a) Street light poles with over-segmentation. (b) Street light poles with significantly incomplete data.

Scene 2. The second dataset was collected from ICEC and a part of RRS. There are two types of roads in the ICEC dataset: roads with a single lane, and roads with four lanes. On the roads with a single lane, the average distance between neighboring trees is about 5.51 m [see Fig. 12(c)]. On the roads with four lanes, a half of street light poles were among the trees with a neighboring distance of 9.75 m. The canopy of these trees is so large that the street light poles are almost surrounded by these trees [see Fig. 12(d)]. Nearly two thirds of street light poles are highly occluded in this dataset.

Scene 3. The dataset was collected from YDT. The roads in the YDT dataset have two lanes. The average distance between neighboring trees in this dataset is about 6.27 m, and nearly a half of street light poles are occluded [see Fig. 12(b)].

More details of the three datasets are shown in Table I. In our datasets, there are nine types of street light poles (see Fig. 10). It is clear that our test dataset is very challenging for street light pole extraction.

B. Street Light Pole Extraction

First, the raw point clouds were divided into segments. Next, the non-ground points were segmented into supervoxels. The

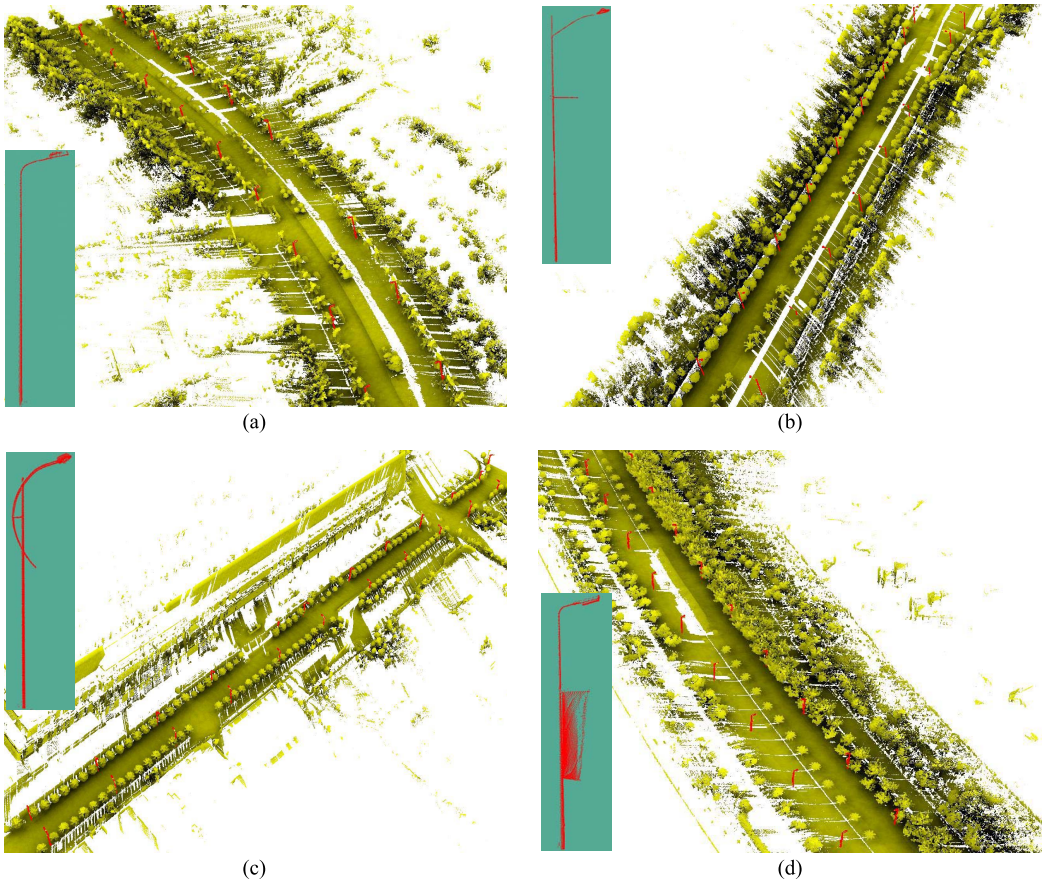


Fig. 12. Illustrations of street light pole extraction examples on the three datasets. (a) Street light pole extraction on the RSR dataset. (b) Street light pole extraction on the YDT dataset. (c) and (d) Street light pole extraction of ICEC dataset.

size of each supervoxel is r_s, and it should be less than the diameter of a street light pole. Then, the positions of pole-like objects were obtained by our localization method. The length of a grid is d_c. The parameters used in the preprocessing and localization stage, and localization results are presented in Table II. The localization results shows the number of street light poles that have been detected as pole-like objects. Afterwards, the proposed guided segmentation was used to segment pole-like objects. The segmentation parameters used in the three datasets are shown in Table III.

Then, two types of features were calculated for each segmented object. To quantitatively assess the accuracy and correctness of our classification method, three metrics including recall, precision, and F_1-measure are employed. The recall represents the percentage of true positives in the ground truth, the precision represents the percentage of true positives in the extracted result, and the F_1-measure is a combination of the two metrics. They are calculated as follows:

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4}$$

$$F_1 - \text{measure} = \frac{2 \cdot \text{recall} \cdot \text{precision}}{\text{recall} + \text{precision}} \tag{5}$$

where TP, FN, and FP denote the number of true positives, false negatives, and false positives, respectively. The correctly detected street light poles are considered as true positives and the undetected street light poles are considered as false negatives. The objects which are falsely detected as street light poles are considered as false positives. SVM and random forests were used to classify the objects. The radius basis function was used as the kernel for SVM, and the parameter cost for SVM was set to 6.0, 5.0, and 1.0 for the RRS, ICEC, and YDT datasets, respectively. The number of trees used in the random forests was set to 100.

Finally, a 10-fold cross validation approach was used to obtain classification and extraction results [32]. Particularly, each dataset was approximately and randomly divided into ten subsets. Each subset was obtained by stratified sampling and had a similar data distribution. The nine subsets were used for training and the remaining one subset was used for test. For instance, there were 3303 samples for training and 367 samples for test in each experiment on the RRS dataset. Totally, ten different groups of training and test data were then obtained following this approach, resulting in ten classification results.

For each 10-fold cross validation experiment, the number of true positives and the number of false positives in the 10-fold experiments are counted. Then, the number of false negatives is obtained by extracting the number of true positives from the ground-truth number of street light poles. Subsequently, the results for a 10-fold cross validation experiment can be achieved. The 10-fold cross validation experiment was repeated for ten times using different dataset partitions. Finally, the

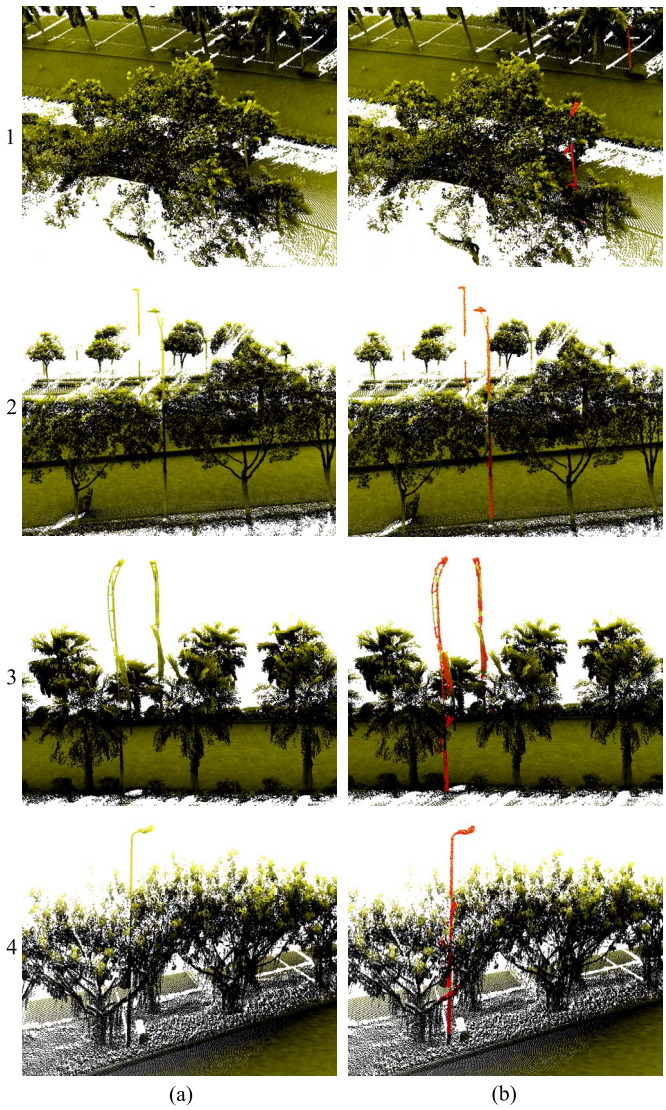


Fig. 13. Illustration of street light pole extraction results in point clouds with missing data and occlusions. (a) Original point clouds. (b) Street light pole extraction results.

mean and stand deviation (std) of recall, precision, and F_1-measure for the ten 10-fold cross validation experiments were calculated. Note that, the standard deviation of the three metrics can be used to measure the robustness of our method.

Street light pole extraction results achieved on the three datasets by the two classifiers are shown in Tables IV and V, respectively. Our algorithm with SVM achieved an average recall, precision, and F_1-measure of 96.3%, 97.0%, and 96.6%, respectively. Our algorithm with random forests obtained an average recall, precision, and F_1-measure of 95.9%, 99.2%, and 97.5%, respectively. As shown in Tables IV and V, most of the street light poles were correctly extracted in each dataset using our algorithms. Besides, the standard deviations for all of the three metrics are smaller than 0.01. It is clearly shown that our algorithm is very stable and robust.

Due to the complexity of experimental data, many street light poles are surrounded by trees. Furthermore, some street light poles are surrounded and even lower than trees. The two kinds

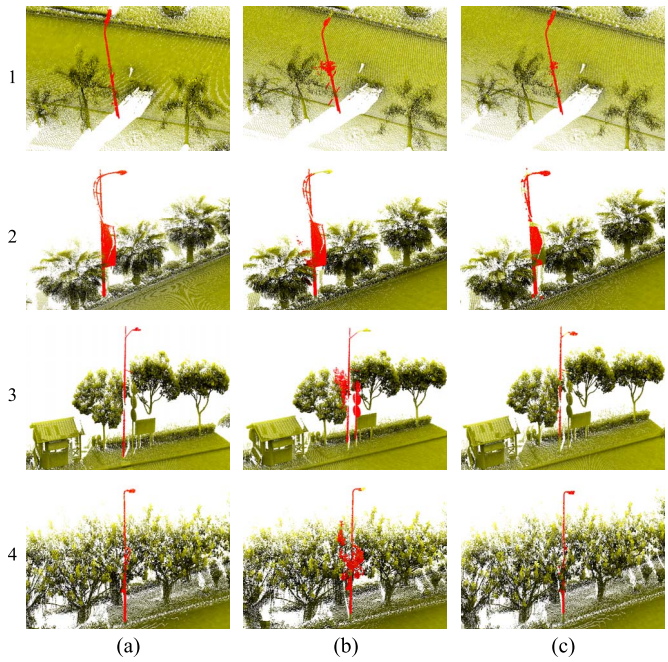


Fig. 14. Comparative segmentation results. (a) Ground truth. (b) Results of the min cut segmentation method. (c) Results of our guided segmentation.

of street light poles might result in over-segmentation results [see Fig. 11(a)]. On the other hand, some street light poles are far away from scanners and occluded by other objects, resulting in seriously incomplete data [see Fig. 11(b)]. These factors will introduce street light pole extraction failures, and finally make the recall results worse than the pole-like object localization results for street light poles.

The street light pole extraction examples on the three datasets are shown in Figs. 12 and 13. Since our localization algorithm is robust to occlusion (see Section IV-D2), the proposed guided based segmentation method achieves good performance (see Section IV-C1). Besides, since the pole and whole object are significantly different, two types of features for object description are suitable for classification. On the other hand, the SVM and random forests are well-investigated classifiers. Consequently, the proposed method achieved promising results even on point clouds with significant occlusions (as shown in Fig. 13).

C. Comparative Studies

1) Segmentation Performance: In order to evaluate the performance of our guided segmentation, the min cut algorithm [14] was used for comparison. Several complex scenes in the three datasets were used for the comparative experiments. These results are shown in Fig. 14. The min cut method was carried out by manually selecting a point in a street light pole as a seed point. It achieved promising results in our test examples, but it failed in highly occluded environment [see Figs. 14 (3) and (4)]. Since this approach is a greedy algorithm. It is impossible to select a distance threshold to cluster foreground points in the connected component without clustering background points in occluded scenes. The method is sensitive to those

TABLE VI
COMPARISON OF EVALUATION RESULTS ACHIEVED ON THE YDT DATASET

Dataset	Algorithm	Quantitative Evaluation		
		Recall	Precision	F_1-measure
YDT	DoPP [13]	**98.6%**	84.1%	90.8%
	Shape template matching [3]	86.4%	93.0%	89.5%
	Our method with random forests	97.9%	**99.4%**	**98.6%**

points between the foreground and background. Our guided segmentation method takes advantage of the prior information of detected positions. Therefore, in our first step of segmentation, it can achieve accurate results and capture most points in street light poles [see Fig. 6(d)]. In summary, our guided segmentation method is more suitable for the segmentation of street light poles than the min cut algorithm.

2) Classification Performance: Our algorithm using random forests was further compared to Hu's DoPP method [13] and Yu's shape template matching method [3]. The YDT dataset was used in the comparative experiment. The key step for the DoPP method is to set an appropriate height threshold. If the largest z value in a grid is above the pre-defined threshold, the points in the grid are considered as belonging to a street light pole. Otherwise, these points are removed. The threshold has to be manually determined and some other objects higher than street light poles can easily be recognized as street light poles, resulting in a large number of false positives (as shown in Table VI). The Ncut method used in [3] needs manual determination for object segmentation. Besides, this method does not use any prior information of positions and the segmentation performance in occlusion is relatively poor. Based on the good segmentation performance and the two types of well-designed features, random forests can achieve highly distinctive classification results. Since the DoPP and the shape template matching methods do not require any training stages, each method was tested on the whole test data in a time. In contrast, our methods require model training, the whole test data were tested in 10 times using 1/10 of the data for each time. Since the overall test data are the same for all these methods, the comparison is fair. The comparative results are presented in Table VI. From theory analyses and experimental results, it is clear that our algorithm outperforms the two existing algorithms.

D. Parameter Tunning

1) The Size of Supervoxels: Since our method is based on supervoxels, the size of supervoxels r_s was firstly tested. We set the supervoxel size to 0.03 m, 0.05 m, 0.1 m, 0.15 m, and 0.2 m. The F1-meausre performance of the proposed algorithm under different values of supervoxel size is shown in Fig. 15. The F1-measure performance decreased dramatically when the supervoxel size was increased from $r_s = 0.15$ m to $r_s = 0.2$ m. Less points share similar features with the increase of the size of supervoxels. As a result, less points can be used to generate local features and the performance of our guided segmentation method was decreased. Consequently, we set the supervoxel size to 0.05 m in this paper.

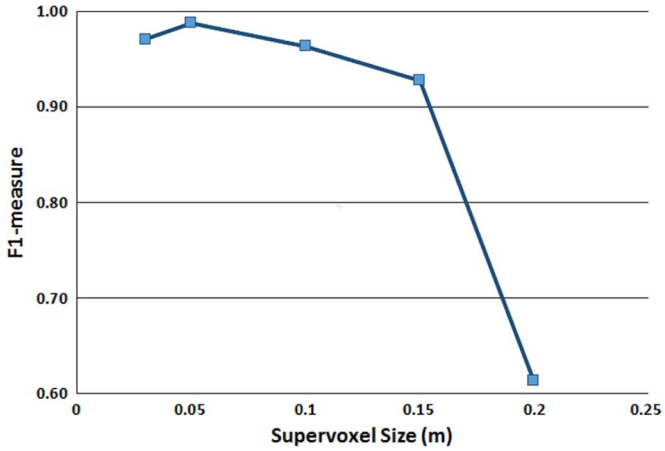


Fig. 15. F1 measure achieved by our method with different sizes of supervoxels.

2) Localization-Related Parameters: The localization step is important for our method and several parameters have been used in this step. Different values of the four important parameters (i.e., the threshold intensity p_t, the point number n_g, the effective count n_f, and the height offset h_o) were used to test the robustness of our localization algorithm, their results are shown in Fig. 16. p_t was set to 10, 20, 30, 40, 50 and n_g was set to 50, 60, 80, 100, 120. The localization detection performance achieved by our method with different values of p_t and n_g is very stable. The variation of detection rates is less than 3%. We then set n_f to 3, 5, 7, 9, 11, the results in Fig. 16(c) show that the location detected rate decreased with n_f. A set of values (1, 3, 5, 7, 9) was used for h_o, it is clear from Fig. 16(d) that the location detection rate increased with h_o. The location detection rates achieved in the four experiments are higher than 90%. It can be concluded that our method is very robust to different parameter settings.

E. Time Performance

The proposed algorithm was implemented in C++ with a single thread on a PC with an Intel Core $i5-3470H$ CPU. The CPU clock frequency is 3.20 GHz, and the memory size is 8 GB. The time costed by each stage of our method is shown in Table VII. On each dataset, more than four fifths of the total time was costed in the ground filtering and supervoxel generation stages. The remaining time was mainly costed by localization, segmentation, and feature calculation stages. The proportion of training time is minor. The total running time on the three datasets was less than one hour. Since the non-ground points are first segmented into supervoxels, the number of processing units is greatly reduced. It is fully demonstrated that our algorithm is very efficient and suitable for large-scale point cloud processing.

V. CONCLUSION

In this paper, we presented an automated and robust algorithm to extract street light poles from large-scale mobile LiDAR point clouds. Our method is robust to occlusion and

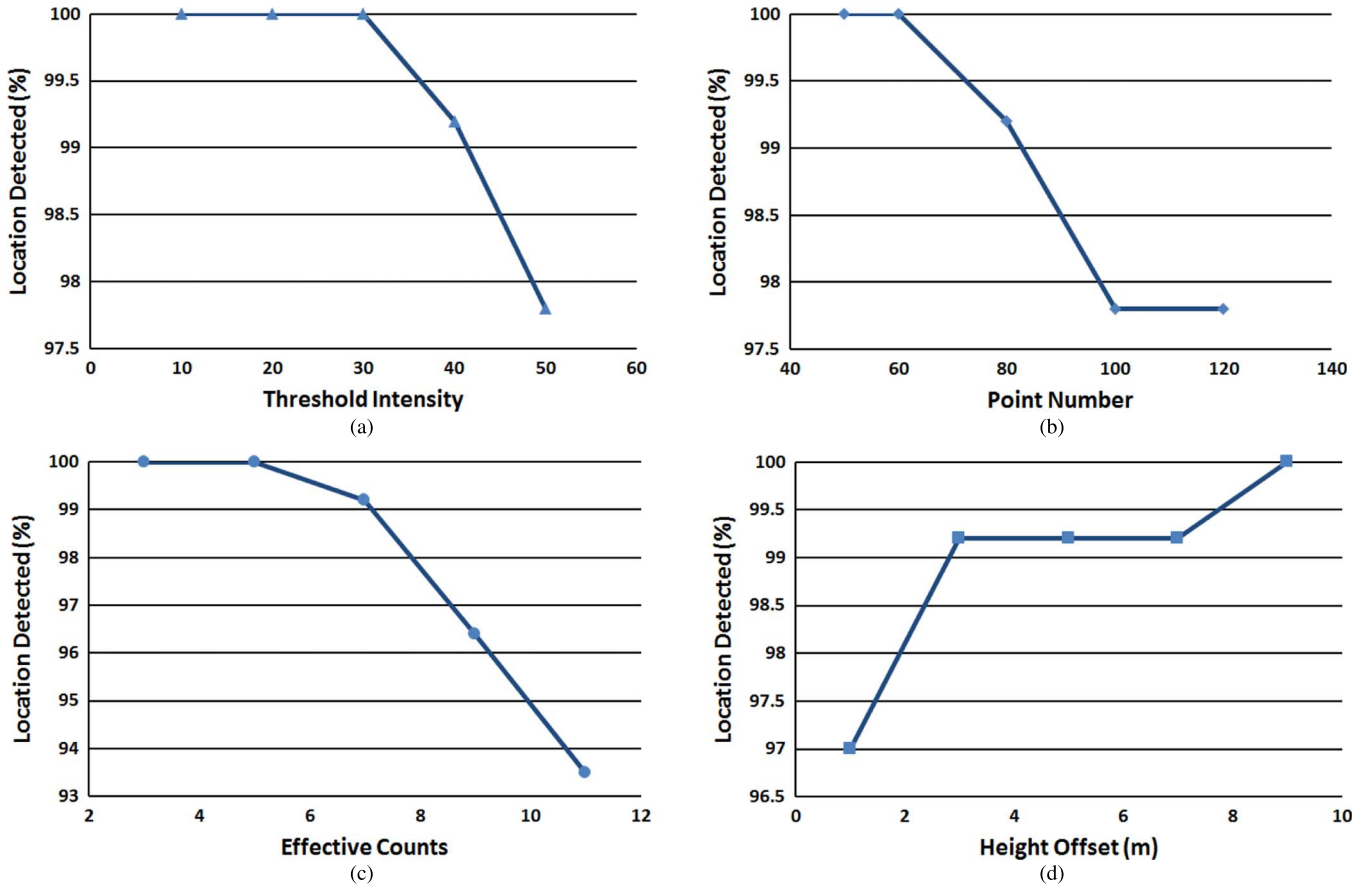


Fig. 16. Localization performance achieved by our method with different parameter settings. (a) Performance with different values of threshold intensity p_t. (b) Performance with different values of point number n_g. (c) Performance with different values of effective counts n_f. (d) Performance with different values of height offset h_o.

TABLE VII
RUNNING TIME COSTS BY EACH STAGE OF OUR METHOD

Dataset	#Points(million)	Ground Filtering(s)	Supervoxel(s)	Localization, Segmentation and Feature Calculation(s)	Training and Labeling(s)	Total Time(s)
RSR	403	775.4	747.8	275.2	8.2	1806.6
ICEC	220	416.6	441.3	172.2	2.4	1032.5
YDT	78	158.1	181.2	64.0	1.6	404.9

highly efficient. Our approach was evaluated on three datasets with significant occlusion. Experimental results and theory analyses show that our localization method is very robust. Our localization method achieves an average recall of 98.8%. It is demonstrated by experimental results and theory analyses that our guided segmentation method outperforms the min cut method for the task of street light pole segmentation. Besides, our SVM based algorithm achieves an average recall, precision, and F_1-measure of 96.3%, 97.0%, and 96.6%, respectively. Our random forest based method obtains an average recall, precision, and F_1-measure of 95.9%, 99.2%, and 97.5%, respectively. Our proposed method is therefore, superior to the Hu's DoPP method and the Yu's shape matching method. Experimental results also demonstrate that our algorithm can efficiently extract street light poles from large-scale point clouds using a normal machine. Overall, the proposed method is highly accurate, efficient and robust to occlusion. In our future work, we will run our method on more large-scale point clouds acquired from other cities and countries to further demonstrate the effectiveness of the method.

ACKNOWLEDGMENT

The authors would like to thank Mr. Michael P. McAllister for the English editing.

REFERENCES

[1] K. Takeuchi, S. Kawai, K. Shibata, and Y. Horita, "Distinction of winter road surface conditions using road surveillance camera," in *Proc. Int. Conf. ITS Telecomm.*, 2012, pp. 663–667.
[2] J. D. Bullough and M. S. Rea, "Intelligent control of roadway lighting to optimize safety benefits per overall costs," in *Proc. IEEE Conf. Intell. Transp. Syst.*, 2011, pp. 968–972.
[3] Y. Yu, J. Li, H. Guan, C. Wang, and J. Yu, "Semiautomated extraction of street light poles from mobile LiDAR point-clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1374–1386, Mar. 2015.
[4] K. Fleischer and H.-H. Nagel, "Machine-vision-based detection and tracking of stationary infrastructural objects beside inner-city roads," in *Proc. IEEE Intell. Transp. Syst.*, 2001, pp. 525–530.

[5] M. Abdalla and S. M. Easa, "Extracting streetlight poles from orthophotos: Methodology and case study in Ontario, Canada," *J. Surveying Eng.*, vol. 133, no. 4, pp. 184–187, Nov. 2007.
[6] K. Williams, M. J. Olsen, G. V. Roe, and C. Glennie, "Synthesis of transportation applications of mobile LiDAR," *Remote Sens.*, vol. 5, no. 9, pp. 4652–4692, 2013.
[7] Y. Yu, H. Guan, and Z. Ji, "Automated detection of urban road manhole covers using mobile laser scanning data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3258–3269, Dec. 2015.
[8] Y. Yu, J. Li, H. Guan, and C. Wang, "Automated extraction of urban road facilities using mobile laser scanning data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1–15, Aug. 2015.
[9] H. Guan, J. Li, Y. Yu, M. Chapman, and C. Wang, "Automated road information extraction from mobile laser scanning data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 194–205, Feb. 2015.
[10] H. Guan, J. Li, Y. Yu, Z. Ji, and C. Wang, "Using mobile LiDAR data for rapidly updating road markings," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 5, pp. 2457–2466, Oct. 2015.
[11] H. Yokoyama, H. Date, S. Kanai, and H. Takeda, "Detection and classification of pole-like objects from mobile laser scanning data of urban environments," *Int. J. CAD/CAM*, vol. 13, no. 2, pp. 31–40, 2013.
[12] S. I. El-Halawany and D. D. Lichti, "Detection of road poles from mobile terrestrial laser scanner point cloud," in *Proc. IEEE Int. Workshop M2RSM*, 2011, pp. 1–6.
[13] Y. Hu, X. Li, J. Xie, and L. Guo, "A novel approach to extracting street lamps from vehicle-borne laser data," in *Proc. Int. Conf. Geoinformat.*, 2011, pp. 1–6.
[14] A. Golovinskiy and T. Funkhouser, "Min-cut based segmentation of point clouds," in *Proc. Int. Conf. Workshop Comput. Vision*, 2009, pp. 39–46.
[15] B. Yang and Z. Dong, "A shape-based segmentation method for mobile laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 81, pp. 19–30, 2013.
[16] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
[17] J. Reitberger, C. Schnörr, P. Krzystek, and U. Stilla, "3D segmentation of single trees exploiting full waveform LiDAR data," *ISPRS J. Photogramm. Remote Sens.*, vol. 64, no. 6, pp. 561–574, Nov. 2009.
[18] H. Wang *et al.*, "Object detection in terrestrial laser scanning point clouds based on Hough forest," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 10, pp. 1807–1811, Oct. 2014.
[19] R. Achanta *et al.*, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Oct. 2014.
[20] A. P. Moore, J. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel lattices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.
[21] O. Veksler, Y. Boykov, and P. Mehrani, "Superpixels and supervoxels in an energy optimization framework," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 211–224.
[22] J. Papon, A. Abramov, M. Schoeler, and F. Worgotter, "Voxel cloud connectivity segmentation-supervoxels for point clouds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2027–2034.
[23] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy-rate clustering: Cluster analysis via maximizing a submodular function subject to a matroid constraint," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 99–112, Jan. 2014.
[24] J. Wang and X. Wang, "VCells: Simple and efficient superpixels using edge-weighted centroidal Voronoi tessellations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1241–1247, Jun. 2012.
[25] W. T. Wong, F. Y. Shih, and T. F. Su, "Thinning algorithms based on quadtree and octree representations," *Inf. Sci.*, vol. 176, no. 10, pp. 1379–1394, 2006.
[26] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 3212–3217.
[27] H. Wang *et al.*, "3-D point cloud object detection based on supervoxel neighborhood with Hough forest framework," *IEEE J. Sel. Topics in Appl. Earth Observ. Remote Sens.*, vol. 8, no. 4, pp. 1570–1581, Apr. 2015.
[28] C. Wen, J. Li, H. Luo, and Y. Yu, "Spatial-related traffic sign inspection for inventory purposes using mobile laser scanning data," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 27–37, Jan. 2016.
[29] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
[30] A. M. Andrew, "Another efficient algorithm for convex hulls in two dimensions," *Inf. Proc. Lett.*, vol. 9, no. 5, pp. 216–219, 1979.
[31] R. Szeliski, *Computer Vision: Algorithms and Applications*. New York, NY, USA: Springer-Verlag, 2010. [Online]. Available: http://szeliski.org/Book/
[32] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*. Burlington, MA, USA: Morgan Kaufmann, 2005.

Fan Wu received the B.Eng. degree in software engineering from Fuzhou University, Fuzhou, China, in 2012. He is currently working toward the M.Sc. degree in the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, China.

He is with the Fujian Collaborative Innovation Center for Big Data Applications in Governments. His research interests include computer vision, machine learning, mobile laser scanning, and information extraction from 3-D point clouds.

Chenglu Wen (M'14) received the Ph.D. degree in mechanical engineering from China Agricultural University, Beijing, China, in 2009.

She is currently an Associate Professor with the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, China. She is also with the Fujian Collaborative Innovation Center for Big Data Applications in Governments. Her research interests are machine vision, machine learning, and point clouds data processing. She has coauthored more than 30 research papers published in refereed journals and proceedings

Prof. Wen is the Secretary of the International Society for Photogrammetry and Remote Sensing WG I/3 on Multi-Platform Multi-Sensor System Calibration (2012–2016).

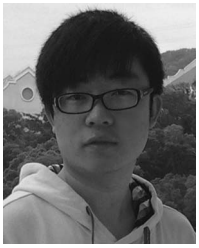
Yulan Guo (M'15) received the B.Eng. and Ph.D. degrees in information and communication engineering from National University of Defense Technology (NUDT), Changsha, China, in 2008 and 2015, respectively.

From November 2011 to November 2014, he was a Visiting Ph.D. Student with University of Western Australia, Perth, Australia. He is currently an Assistant Professor with the College of Electronic Science and Engineering, NUDT. He authored more than 30 peer reviewed journal and conference publications [including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and *International Journal of Computer Vision* (IJCV)] and one book chapter. He served as a Reviewer for more than 20 international journals (e.g., *IJCV*) and several international conferences. His research interests include computer vision and pattern recognition, particularly on 3-D feature extraction, 3-D modeling, 3-D object recognition, and 3-D face recognition.

Prof. Guo is a Member of the Association for Computing Machinery and China Computer Federation (CCF). He is also a Committee Member for the CCF Computer Vision Task Force and the Vision and Learning Seminar. He was the recipient of the NUDT Distinguished Ph.D. Thesis award in 2015 (5 out of about 300).

Jingjing Wang received the B.Sc. degree in computer science from Xiamen University, Xiamen, China, in 2013, where she is currently working toward the M.Sc. degree.

Her research interests include data mining, machine learning, parallel computing using Hadoop, and Storm.



Yongtao Yu received the B.Sc. and Ph.D. degrees in computer science and technology from Xiamen University, Xiamen, China, in 2015.

He is an Assistant Professor with the Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Jiangsu, China. He has coauthored more than 20 research papers published in refereed journals and proceedings. His research interests include pattern recognition, computer vision, machine learning, intelligent interpretation of 3-D point clouds, and remotely sensed imagery.

Prof. Yu was the recipient of the Excellent Doctoral Dissertation Award of Fujian Province in 2015.

Cheng Wang (M'11–SM'16) received the Ph.D. degree in information and communication engineering from National University of Defense Technology, Changsha, China, in 2002.

He is a Professor and an Associate Dean of the School of Information Science and Engineering, and Executive Director of Fujian Key Laboratory of Sensing and Computing for Smart City, both at Xiamen University, Xiamen, China. He is also with the Fujian Collaborative Innovation Center for Big Data Applications in Governments. His research interests include remote sensing image processing, mobile light detection and ranging data analysis, and multisensor fusion.

Dr. Wang is the Cochair of the International Society for Photogrammetry WG I/3 on Multi-Platform Multi-Sensor System Calibration (2012–2016) and the Council Member of the China Society of Image and Graphics. He has coauthored more than 80 papers in referred journals, including IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING, *International Journal of Remote Sensing*, and International Society for Photogrammetry and Remote Sensing's *Journal of Photogrammetry and Remote Sensing*.

Jonathan Li (M'00–SM'11) received the Ph.D. degree in geomatics engineering from University of Cape Town, Cape Town, South Africa.

He is currently a Professor with Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, China. He is also with the Fujian Collaborative Innovation Center for Big Data Applications in Governments. He is also the Professor and Head of the Mobile Sensing and Geodata Science Laboratory, Faculty of Environment, University of Waterloo, Waterloo, ON, Canada. He has coauthored more than 300 publications, over 100 of which were published in refereed journals, including IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (IEEE-TITS), IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, International Society for Photogrammetry and Remote Sensing's *Journal of Photogrammetry and Remote Sensing*, IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (IEEE-JSTARS), *International Journal of Remote Sensing*, *Photogrammetric Engineering & Remote Sensing*, and *The Royal Society of Edinburgh*. His research interests include information extraction from mobile light detection and ranging point clouds and from Earth observation images.

Dr. Li is the Chair of the International Society for Photogrammetry and Remote Sensing Inter Commission Working Group I/Va on Mobile Scanning and Imaging Systems (2012–2016), Chair of the International Cartographic Association Commission on Sensor-driven Mapping (2015–2019), and Associate Editor of IEEE-TITS and IEEE-JSTARS.