

Road Boundaries Detection Based on Local Normal Saliency From Mobile Laser Scanning Data

Hanyun Wang, Huan Luo, Chenglu Wen, *Member, IEEE*, Jun Cheng, Peng Li, Yiping Chen, Cheng Wang, *Member, IEEE*, and Jonathan Li, *Senior Member, IEEE*

Abstract—The accurate extraction of roads is a prerequisite for the automatic extraction of other road features. This letter describes a method for detecting road boundaries from mobile laser scanning (MLS) point clouds in an urban environment. The key idea of our method is directly constructing a saliency map on 3-D unorganized point clouds to extract road boundaries. The method consists of four major steps, i.e., road partition with the assistance of the vehicle trajectory, salient map construction and salient points extraction, curb detection and curb lowest points extraction, and road boundaries fitting. The performance of the proposed method is evaluated on the point clouds of an urban scene collected by a RIEGL VMX-450 MLS system. The completeness, correctness, and quality of the extracted road boundaries are 95.41%, 99.35%, and 94.81%, respectively. Experimental results demonstrate that our method is feasible for detecting road boundaries in MLS point clouds.

Index Terms—Mobile laser scanning (MLS), point cloud, road boundary, saliency map.

I. INTRODUCTION

ROADS, as an important urban structure, provide a continuous surface spanning an entire city and provide contextual cues for recognizing important scene structures (e.g., road signs, lamp posts, road markings, and cars). Consequently, the accurate extraction of roads is an important issue in road surveying, automatic drive, and autonomous vehicle navigation.

A mobile laser scanning (MLS) system is developed as a cost-effective solution for large-scale 3-D data acquisition. Some methods have been proposed to detect road boundaries from MLS point clouds [1], [2]. However, these methods are either sensitive to the point density variation and overlap with green spaces or have a high computational complexity. Zhou and Vosselman [3] proposed a sigmoidal function to fit the

points near a detected curbstone, but their method requires computing the digital terrain model first. Some methods also utilize existing geographic information system maps to extract roads from MLS point clouds. A 2-D road map acquired online provides the topology and geometry of a road. Boyko and Funkhouser [4] utilized a 2-D road map acquired from OpenStreetMap to partition a road into patches. However, the 2-D map is required to be registered with 3-D point clouds. To deal with a large-scale point cloud, in our previous work, we utilized the vehicle trajectory to partition the point cloud into data blocks [5]. Because model parameters are estimated through the Random Sample Consensus (RANSAC) algorithm, this method is sensitive to the ratio of road surface points in each data block. Guan *et al.* [6] utilized the trajectory to extract a set of profile examples and only detected two points in each profile. Based on a very sparse point set, the extracted road boundaries are not robust to the change in the road direction. Except for 3-D laser scanning systems, some methods utilize 2-D Light Detection and Ranging (LIDAR) to deal with road detection problems. Han *et al.* [7] utilized Markov chain propagation models to conduct road detection using a downward-looking 2-D LIDAR sensor. Another 2-D-LIDAR-sensor-based method uses multiple Kalman filters to integrate the sensor measurements for road boundary detection [8].

Curbs are important cues identifying the boundary of a roadway [9]. We assume that the lowest points of road curbs constitute road boundaries that separate pedestrian pavements or other green spaces from road surfaces. In this letter, we propose a local-normal-saliency-based method to extract road boundaries from MLS point clouds in an urban environment. Fig. 1 shows the workflow of our method for automatic road boundaries detection. Our method includes four steps: 1) road partition, where MLS point clouds are partitioned into a number of overlapping data blocks with the assistance of the vehicle trajectory; 2) saliency map construction and salient points extraction, where, for each data block, the saliency map of a 3-D unorganized point cloud is computed and salient points are extracted adaptively; 3) curb detection and curb lowest points extraction, where road curbs are detected and the lowest points of curbs are extracted from salient points; and 4) road boundary fitting, where the curb lowest points of adjacent data blocks are merged and further fitted into smooth curves through a curve fitting algorithm.

II. METHOD

A. Trajectory-Based Road Partition

MLS systems provide both 3-D laser scanning point clouds and the vehicle trajectory. The trajectory represents the location

Manuscript received March 15, 2015; revised May 12, 2015; accepted June 22, 2015. Date of publication July 9, 2015; date of current version August 7, 2015. This work was supported by the National Science Foundation of China under Project 61371144. (*Corresponding author: Chenglu Wen.*)

H. Wang, P. Li, and Y. Chen are with the School of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China.

H. Luo, C. Wen, J. Cheng, and C. Wang are with the Fujian Key Laboratory of Sensing and Computing for Smart City and the School of Information Science and Engineering, Xiamen University, Xiamen 361005, China (e-mail: clwen@xmu.edu.cn).

J. Li is with the Key Laboratory of Underwater Acoustic Communication and Marine Information Technology (MOE), Xiamen University, Xiamen 361005, China, and also with the Waterloo Laboratory for GeoSpatial Technology and Remote Sensing (GeoSTARS Laboratory), Department of Geography and Environmental Management, Faculty of Environment, University of Waterloo, Waterloo, ON N2L 3G1, Canada.

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/LGRS.2015.2449074

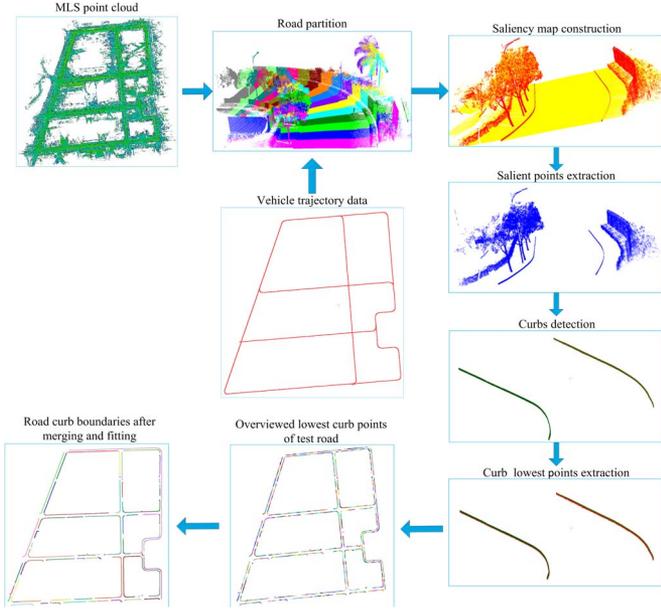


Fig. 1. Workflow of the road boundaries detection algorithm.

and direction of a mobile mapping van on a road and indirectly indicates the road location and direction since the scanned data and the trajectory are transformed into a unified coordinate system. With the assistance of the trajectory, the whole point cloud of the test area is divided into a number of overlapping data blocks that ensure the continuity of road boundaries. Specifically, by selecting trajectory points at a constant time interval T_s , a set of trajectory points with an approximate constant distance interval is generated. These points are used to partition the whole point cloud into overlapping data blocks. Considering the road surface rising and falling, the slope of the road surface in each data block is constrained to change slowly, and this constraint is formulized as

$$\frac{t_{ez} - t_{sz}}{\sqrt{(t_{ex} - t_{sx})^2 + (t_{ey} - t_{sy})^2}} - \frac{t_{mz} - t_{sz}}{\sqrt{(t_{mx} - t_{sx})^2 + (t_{my} - t_{sy})^2}} < S_{thr} \quad (1)$$

where (t_{sx}, t_{sy}, t_{sz}) , (t_{mx}, t_{my}, t_{mz}) , and (t_{ex}, t_{ey}, t_{ez}) are the start point, middle point, and endpoints of the trajectory points in each data block, respectively. The time interval of the middle point to both the start point and the endpoint is $T_s/2$. If the slope difference between a line connecting the start and end trajectory points and a line connecting the start and middle trajectory points is larger than the given threshold S_{thr} , the data block is divided into two subblocks according to the middle trajectory point.

B. Saliency Map Construction and Salient Points Extraction

In this section, we propose a method to construct a saliency map in a bottom-up manner. Compared with other saliency map construction methods [10], our method directly constructs a saliency map on 3-D unorganized point clouds. The saliency is measured by projecting the distance of each point's normal vector to the point cloud's dominant normal vector into a hyperbolic tangent function space. Given a point cloud, the

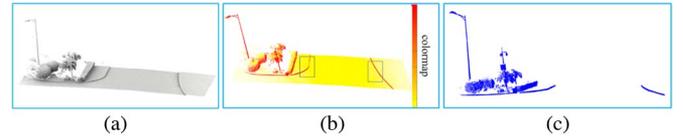


Fig. 2. Illustration of saliency map construction and salient points extraction. (a) Raw point cloud. (b) Constructed saliency map. (c) Extracted salient points.

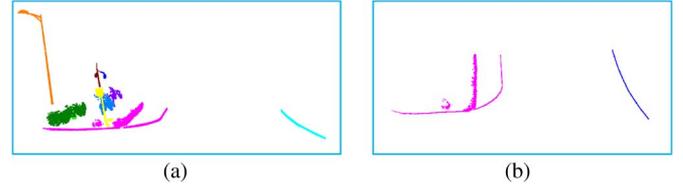


Fig. 3. Illustration of curbs extraction. (a) Salient points segmentation. (b) Curbs extraction.

dominant normal vector is estimated by separating the point cloud normal vectors into k clusters through a K -means clustering method [11]. The centroid of the largest cluster is treated as the dominant normal vector (denoted as \mathbf{n}_m) of this point cloud. For each point p_i , the Euclidean distance d_i of its normal vector \mathbf{n}_i to the dominant normal vector \mathbf{n}_m is computed. Then, the saliency of point p_i is obtained by projecting d_i into a hyperbolic tangent function space [12] by

$$S(d_i) = \frac{1 + \tanh(\lambda(d_i - c))}{2} = \frac{1}{2} \left(1 + \frac{e^{\lambda(d_i - c)} - e^{-\lambda(d_i - c)}}{e^{\lambda(d_i - c)} + e^{-\lambda(d_i - c)}} \right) \quad (2)$$

where c represents the geometric centroid of the hyperbolic tangent function, and λ controls the sharpness of the shape of curve. The aim of this nonlinear transformation is to ensure a large enough saliency difference between salient and nonsalient points.

In order to estimate c adaptively, we first construct a histogram consisting of a set of bins. Each bin counts the number of points whose scalar quantity d_i falls into a given range of values. Let $\{N_k, k = 1, 2, \dots, K\}$ represent the number of points falling into K bins. The probability of a point falling into the k th bin is formulized as

$$\hat{p}_k = \frac{N_k}{N}, \quad k = 1, 2, \dots, K. \quad (3)$$

Then, the geometric centroid is estimated by

$$c = \frac{\sum_{k=1}^K \left(\log\left(\frac{1}{\hat{p}_k}\right) * h_k \right)}{\sum_{k=1}^K \log\left(\frac{1}{\hat{p}_k}\right)} \quad (4)$$

where h_k denotes the mean of the scalar quantity corresponding to the k th bin. According to (4), the scalar quantity with fewer points is given a larger weight, whereas the scalar quantity with more points is given a smaller weight. After projection based on the estimated geometric centroid, the difference of saliency between the salient and nonsalient points is enlarged.

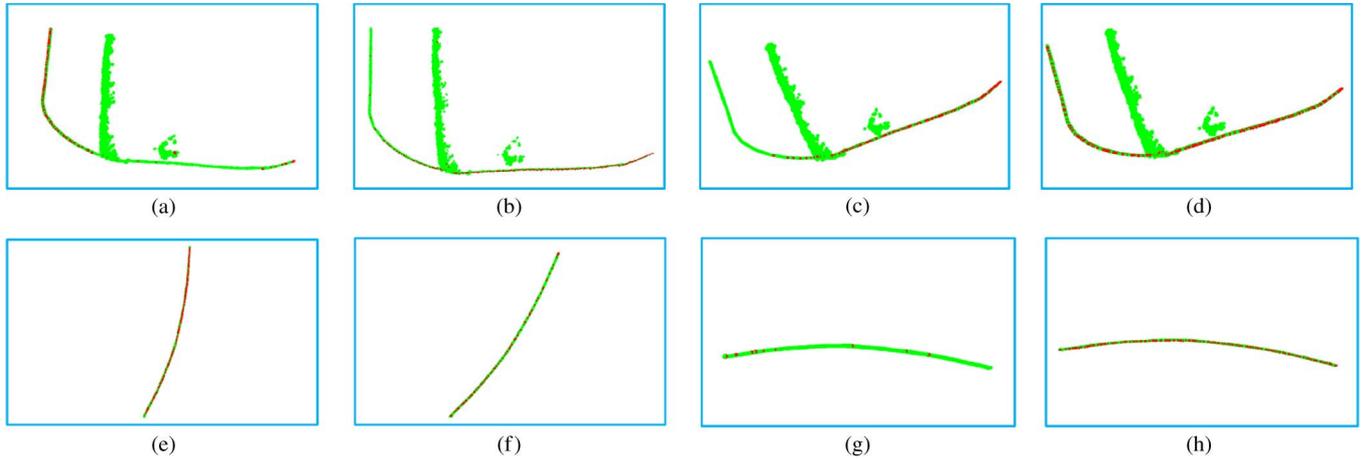


Fig. 4. Bottom view of the extraction of curb lowest points. The green points represent the curb points, and the red points represent the extracted lowest points. The first row shows the extracted results of the left curb, and the second row shows the extracted results of the right curb.

The salient points are extracted by classifying the points into salient and nonsalient classes. The classification is mathematically formulated as

$$p_i = \begin{cases} 1, & \text{if } S(d_i) \geq S(c) \\ 0, & \text{if } S(d_i) < S(c) \end{cases} \quad (5)$$

where 1 represents that point p_i is a salient point, and 0 represents that point p_i is a nonsalient point. The salient points are extracted after removing all nonsalient points from the point cloud. Fig. 2(a)–(c) shows the original point cloud, the constructed saliency map, and the extracted salient points, respectively.

C. Curb Detection and Curb Lowest Points Extraction

The salient points extracted in the previous step contain objects such as trees, road curbs, street lamps, and vehicles. In this step, the salient points are first segmented into clusters through a region growing segmentation algorithm [13]. Fig. 3(a) shows the segmentation results, and different colors represent different clusters. Then, curbs are detected based on their elevation, horizontal length, and distance to the trajectory. The elevation criterion separates curbs from objects such as trees, cars, and high fences. The horizontal length criterion separates curbs from objects such as small low fences. The distance to the trajectory separates curbs from objects such as long low fences. These criteria are mathematically formulated as

$$v_i \in \begin{cases} \text{curb}, & \text{if } (L_i > L_T) \ \& \ (E_{\min} < E_i < (E_{\min} + E_{\text{thr}})) \\ & \ \& \ (d_i^t = \min \{d_j^t\}) \\ \text{noncurb}, & \text{otherwise} \end{cases} \quad (6)$$

where L_i denotes the horizontal length of cluster i , L_T is the given horizontal length threshold, E_i denotes the maximal elevation of the points in cluster i , E_{\min} denotes the minimal elevation of the point cloud, E_{thr} denotes the elevation threshold, d_i^t denotes the distance of cluster i to the trajectory line, and d_j^t denotes the distance of clusters to the trajectory line in the same side as cluster i . According to these criteria, the candidate curbs are extracted. Fig. 3(b) shows the extracted curb clusters.

As described in [1], curbs display either an elevation jump or a gradual elevation change between roads and curb surfaces in real-world urban scenes. Yang *et al.* [1] utilized two adjacent moving windows to extract the upper and lower points of curb surfaces. However, their method is not robust when a curb is overlapped with a green space. To extract the lowest points of curbs, a projection-based method is proposed in this letter. Specifically, we first project the points of curbs into the X – Z and Y – Z planes both in the original and principal component analysis (PCA)-based projective point clouds, and then, we extract the lowest points of the curb in these four planes. In the X – Z and Y – Z planes of the original point cloud, all the lowest points in these two planes are treated as the lowest points of curbs. In Fig. 4(a), (b), (e), and (f), the extracted lowest points are shown in red. The projective point cloud is obtained by transforming the original point cloud according to the principal components of the horizontal distribution of the curb. The principal components essentially are the eigenvectors of the covariance matrix of the horizontal components of a curb, which are formulated as

$$\mathbf{CV} = \mathbf{E}\mathbf{V} \quad \mathbf{C} = \text{cov}(\mathbf{H}) \quad \mathbf{H} = \begin{bmatrix} x_1, & y_1 \\ x_2, & y_2 \\ \vdots & \vdots \\ x_n, & y_n \end{bmatrix} \quad (7)$$

where \mathbf{C} is the covariance matrix of curb horizontal components \mathbf{H} , \mathbf{E} is the diagonal matrix of the eigenvalues, and \mathbf{V} are the two orthogonal eigenvectors or principal components. Then, the projective point cloud is obtained through the following transformation:

$$\begin{bmatrix} x'_i \\ y'_i \\ z'_i \end{bmatrix} = \begin{bmatrix} \mathbf{V} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \quad (8)$$

where (x_i, y_i, z_i) are the point coordinates in the original point cloud space, and (x'_i, y'_i, z'_i) are the transformed point coordinates. After transformation, the horizontal coordinates are changed, whereas the elevation is still the same. The lowest points extraction described in the original point cloud is

repeated in the projective point cloud, as shown in Fig. 4(c), (d), (g), and (h). Finally, all of the extracted lowest points are merged together.

D. Road Boundaries Fitting

In our method, a whole scene is divided into many overlapping data blocks. For each data block, the extraction procedure of the curb lowest points is repeated. Then, the curb lowest points of the adjacent data blocks are merged together. Two adjacent lowest point sets are only merged if the distance between these two point sets is smaller than a given threshold m_{thr} . Finally, a curve fitting algorithm is used to fill the small gap and generate smooth curb boundaries [14].

III. EXPERIMENTS AND DISCUSSION

A. Test Data

The test data were collected using a RIEGL VMX-450 MLS system. The VMX-450 system integrates two RIEGL VQ-450 laser scanners (400 lines/s, 1.1 million measurements/s, 8-mm accuracy, and 5-mm precision), as well as an inertial measurement unit and global navigation satellite system equipment. The test data, with about 331 million points, were acquired in a factory area of 828 m \times 792 m in Xiamen, China. The total length of the road in the test data is approximately 5 km, and the total length of road boundaries is 9292.79 m. Furthermore, the test scene contains numerous crossroads, buildings, grass strips, low fences, and cars, which contribute to the complexity of road boundaries detection.

B. Extraction of Road Curb Boundaries

The parameters used in the proposed method are listed in Table I. Our proposed algorithm was implemented on a computer with an Intel Core i3 3.3-GHz processor and random access memory of 16 GB. The averaged running time of extracting the curb lowest points for each data block is about 10 s. The whole time of the proposed algorithm is about 4605 s. In order to visually check the quality of road curb boundaries extraction, we overlaid the fitted results of the curb lowest points on the MLS point clouds, as shown in Fig. 5. Some typical road scenes are enlarged for detailed inspection. In Fig. 5, we observe that our proposed method performs well for road boundary extraction.

C. Quantitative Evaluation

A completeness evaluation metric [1] was used to quantitatively evaluate our method. We compared our method with the state-of-the-art methods proposed in [1] and [6] referred to as Yang's method and Guan's method, respectively. We manually extracted the lowest points of curbs in the test data set as the ground-truth data. The distance interval between neighboring selected points is about 0.5 m. For curbs occluded by vehicles or road intersections, we counted the distance of two endpoints of occluded curbs into the length of the ground truth.

To evaluate the completeness metric of our method, we calculated the following three metrics by comparing the fitted

TABLE I
PARAMETERS AND THEIR VALUES USED IN OUR EXPERIMENTS

Parameter	Description	value
T_s	Time step of selecting trajectory points	2.5s
T_o	Overlapping time step between neighboring data blocks	0.05s
S_{thr}	The slope changing threshold when data partition	0.18 (10 deg)
k	Number of clusters of K-means in dominant normal vector estimation	5
λ	Controlling the sharpness of the shape of hyperbolic tangent function	10.0
K	Number of histogram bins when constructing the saliency map	10
L_T	The minimal horizontal length threshold of the curb	2.5m
E_{thr}	The maximal height threshold of the curb	1.0m
m_{thr}	The merging threshold for adjacent lowest point sets	1.5m

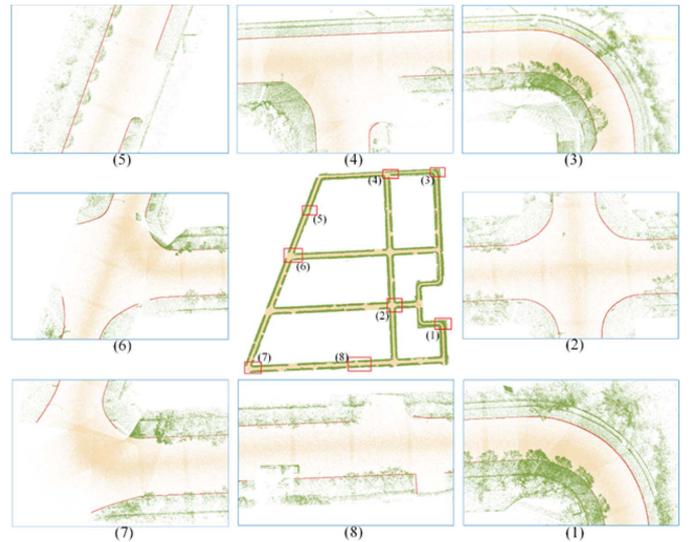


Fig. 5. Road boundary extraction results from the test data.

road boundaries with the ground-truth data:

$$\text{Completeness} : r = \frac{TP}{L_r}$$

$$\text{Correctness} : p = \frac{TP}{L_e}$$

$$\text{Quality} : q = \frac{TP}{(L_e + FN)} = \frac{TP}{(TP + FP + FN)} \quad (9)$$

As described in [1], L_r is the total length of the road boundaries in the ground-truth data, L_e is the total length of the extracted road boundaries, TP is the length of the correctly extracted boundaries, FP is the length of the extracted

TABLE II
LENGTH OF THE EXTRACTED CURB: L_e , TP , FP , AND FN

	L_r (m)	L_e (m)	TP (m)	FP (m)	FN (m)
Guan's Method	9292.79	9646.67	8859.56	787.11	433.23
Yang's Method	9292.79	9053.61	8848.18	205.43	444.61
Our Method	9292.79	8924.34	8866.31	58.03	426.48

TABLE III
COMPLETENESS MEASURE OF THE CURB EXTRACTION RESULTS

	Completeness (%)	Correctness (%)	Quality (%)
Guan's Method	95.34	91.84	87.89
Yang's Method	95.22	97.73	93.16
Our Method	95.41	99.35	94.81

boundaries that do not exist in the ground-truth data, and FN is the length of the boundaries that are not extracted but exist in the ground-truth data. Table II lists the values used for computing the completeness evaluation metric, and Table III lists the completeness evaluation results. For the test data, our method achieves a completeness of 95.41%, a correctness of 99.35%, and a quality of 94.81%. The occlusions caused by vehicles or road intersections are the major contribution to FN in the test data. There are no or fewer points in the occluded curbs because of the occlusion. Grass strips make a contribution to both FN and FP . If grass strips and curbs are too close to each other, tall grass strips make the curb detection from salient points unsuccessful, and low grass strips make a contribution to FP . The intersections of test roads also contribute to FP by making the points in the intersection area have a large normal difference from the dominant normal of the testing data block.

Through Tables II and III, we observe that our method outperforms Guan's method. The reason for the better performance is that Guan's method only extracts two boundary points in each data block; thus, the sparse point set cannot accurately delineate the change in the road boundary. On the contrary, our method extracts dense boundary points in each data block, and these dense points can accurately delineate the change in the road boundary. In addition, Guan's method extracts two points in each data block whether there is a road curb or not, and this step leads to generate more FP . Our method also achieves better performance than Yang's method. This is because the performance of Yang's method is degraded when the point densities in the two sides of a road are not identical.

IV. CONCLUSION

In this letter, we have proposed a novel local-normal-saliency-based method to extract road boundaries from MLS point clouds. The proposed method directly constructs a saliency map on 3-D laser scanning point clouds and extracts road curbs according to their spatial location relationship with contextual objects. To accurately delineate a road direction change, a PCA-projection-based method is proposed to extract dense curb lowest points. Visual inspection and quantitative evaluation demonstrate the robustness and effectiveness of the proposed method in real-world urban scenes. In addition, comparative results demonstrate that our method achieves better completeness, correctness, and quality compared with state-of-the-art methods. In our future work, camera images will be incorporated to deal with the occlusions caused by objects on a road.

REFERENCES

- [1] B. Yang, L. Fang, and J. Li, "Semi-automated extraction and delineation of 3-D roads of street scene from mobile laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 79, pp. 80–93, May 2013.
- [2] P. Kumar, C. P. McElhinney, P. Lewis, and T. McCarthy, "An automated algorithm for extracting road edges from terrestrial mobile LiDAR data," *ISPRS J. Photogramm. Remote Sens.*, vol. 85, pp. 44–55, Nov. 2013.
- [3] L. Zhou and G. Vosselman, "Mapping curbstones in airborne and mobile laser scanning data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 18, pp. 293–304, Aug. 2012.
- [4] A. Boyko and T. Funkhouser, "Extracting roads from dense point clouds in large scale urban environment," *ISPRS J. Photogramm. Remote Sens.*, vol. 66, no. 6, pp. S2–S12, Dec. 2011.
- [5] H. Wang *et al.*, "Automatic road extraction from mobile laser scanning data," in *Proc. IEEE Int. Conf. CVRS*, 2012, pp. 136–139.
- [6] 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.
- [7] J. Han, D. Kim, M. Lee, and M. Sunwoo, "Enhanced road boundary and obstacle detection using a downward-looking LIDAR sensor," *IEEE Trans. Veh. Technol.*, vol. 61, no. 3, pp. 971–985, Mar. 2012.
- [8] Y. Kang, C. Roh, S.-B. Suh, and B. Song, "A Lidar-based decision-making method for road boundary detection using multiple Kalman filters," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4360–4368, Nov. 2012.
- [9] I. Stainvas and Y. Buda, "Performance evaluation for curb detection problem," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 25–30.
- [10] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 185–207, Jan. 2013.
- [11] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Hoboken, NJ, USA: Wiley, 2012.
- [12] I. J. Zucker, "The summation of series of hyperbolic functions," *SIAM J. Math. Anal.*, vol. 10, no. 1, pp. 192–206, 1979.
- [13] T. Rabbani, F. van den Heuvel, and G. Vosselman, "Segmentation of point clouds using smoothness constraint," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 36, pp. 248–253, 2006.
- [14] U. Ozertem and D. Erdogmus, "Locally defined principal curves and surfaces," *J. Mach. Learn. Res.*, vol. 12, pp. 1249–1286, Feb. 2011.