Learning Hierarchical Features for Automated Extraction of Road Markings From 3-D Mobile LiDAR Point Clouds

Yongtao Yu, Jonathan Li, *Senior Member, IEEE*, Haiyan Guan, Fukai Jia, and Cheng Wang, *Member, IEEE*

***Abstract*—This paper presents a novel method for automated extraction of road markings directly from three dimensional (3-D) point clouds acquired by a mobile light detection and ranging (LiDAR) system. First, road surface points are segmented from a raw point cloud using a curb-based approach. Then, road markings are directly extracted from road surface points through multisegment thresholding and spatial density filtering. Finally, seven specific types of road markings are further accurately delineated through a combination of Euclidean distance clustering, voxel-based normalized cut segmentation, large-size marking classification based on trajectory and curb-lines, and small-size marking classification based on deep learning, and principal component analysis (PCA). Quantitative evaluations indicate that the proposed method achieves an average completeness, correctness, and F-measure of 0.93, 0.92, and 0.93, respectively. Comparative studies also demonstrate that the proposed method achieves better performance and accuracy than those of the two existing methods.**

***Index Terms*—Deep learning, mobile light detection and ranging (LiDAR), point cloud, road marking, three dimensional (3-D) extraction.**

I. Introduction

INFORMATION about the current road surface conditions and features, traffic flows, and surrounding environments is one of the most important inputs to the intelligent transportation systems (ITS). Road markings, which are highly reflective objects painted on road surfaces, function to manage and control traffic activities. Detecting and monitoring road markings assist in regulating driving behaviors and reducing traffic accidents. Traditional methods [1], [2] for extracting road markings are based on digital images and videos. Due to illumination conditions, occlusions, distortions, incompleteness, viewpoints, and lack of geospatial information, it is challenging to effectively recognize and extract different types of road markings from images and videos acquired by vehicle-borne optical imaging systems.

Vehicle-borne mobile light detection and ranging (LiDAR) systems have exhibited outstanding advantages over the traditional optical imaging techniques. By using laser scanners, mobile LiDAR systems are able to acquire highly dense and accurate three-dimensional (3-D) point cloud data at a normal urban driving speed day and night. Due to direct views of road surfaces and intensities backscattered by different targets, mobile LiDAR systems have become a promising and cost-effective means for rapid inventory mapping of large-area urban road surfaces. However, automated extraction of road markings from large amounts of 3-D mobile LiDAR point clouds with huge concavo-convex features and varying point densities is a very challenging task. Most of the existing studies simplified such a task by converting 3-D point clouds into two-dimensional (2-D) geo-referenced feature imagery [3], [4] so that image processing techniques could apply. In [3], 3-D point clouds were first rasterized into 2-D geo-referenced feature images through inverse distance weighted (IDW) interpolation. Then, road marking points were automatically extracted using a two-step filter which considered the reflectance and elevation properties of different targets. Finally, shape-based method was applied to extract rectangular-shaped markings. Similarly, a point-density-based multiple thresholding method was proposed in [4] to extract any types of road markings using 2-D geo-referenced feature imagery. In this method, a 2-D geo-referenced feature image was first partitioned into a set of bins according to the point density distribution function. Then, each of the bins was thresholded separately to extract road markings. Finally, morphological operations were applied to remove noise in the extracted road markings. However, such a conversion may lead to incompleteness and incorrectness in feature extraction.

In [5], a fusion approach was presented for detecting and tracking lane markings by fusing the point cloud data from laser scanners and the navigation data from the inertial measurement unit (IMU) and the differential global positioning system (DGPS). A morphological method was proposed in [6] for extracting lane markings, arrow markings, and zebra crossings. In this method, peak detectors were used to detect lane markings; template matching methods were considered to

Manuscript received March 29, 2014; revised July 01, 2014; accepted August 06, 2014. This work was supported in part by the discovery grant from the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant 111368 and in part by the general grant from the Natural Science Foundation of China (NSFC) under Grant 41471379. *(Corresponding author: Jonathan Li.)*

Y. Yu, F. Jia, and C. Wang are with the Fujian Key Laboratory of Sensing and Computing for Smart Cities (SCSC), School of Information Science and Engineering, Xiamen University, Xiamen 361005, China (e-mail: allennessy.yu@gmail.com).

J. Li is with the Key Laboratory of Underwater Acoustic Communication and Marine Information Technology, Xiamen University, Xiamen 361005, 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).

H. Guan is with the Department of Geography and Environmental Management, 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/JSTARS.2014.2347276

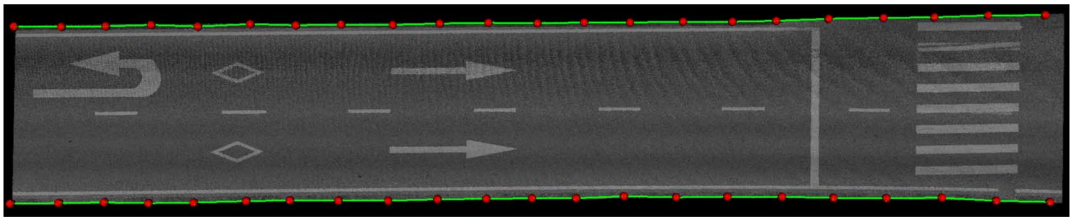


Fig. 1. Segmented road surface along with the ascertained curb points (red dots) and fitted curb-lines (green lines).

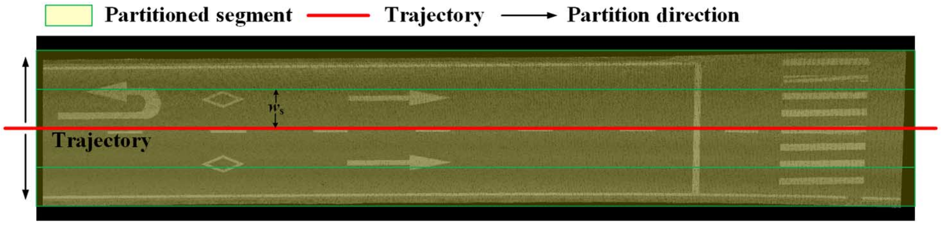


Fig. 2. Multisegment partition model.

detect arrow markings; morphological indicators were applied to detect zebra crossings. Considering the reflectivity difference between road surface and road markings, an adaptive thresholding method [7] was developed to extract lane markings through adaptively minimizing the within-class variance of the reflectivity values. In addition, a profile-based intensity analysis method [8] was adopted to directly extract 3-D lane markings from point clouds. In this method, the raw point clouds were first divided into slices along the trajectory. Then, road surfaces were detected according to the geometric properties of curbs, barriers, and border lines. Finally, lane markings were extracted through locating the intensity peaks on each scan line.

In this paper, we propose an automated method for road marking extraction directly from 3-D mobile LiDAR point clouds. The contributions of this paper are as follows: 1) a trajectory-based multisegment thresholding method; 2) a spatial density filtering method; 3) a trajectory-curb-line-based road marking classification method; and 4) a deep learning and principal component analysis (PCA)-based road marking classification method.

This paper is organized as follows. Section II describes the road marking extraction method. Section III presents the road marking classification method. Section IV reports and discusses the experimental results. Concluding remarks are given in Section V.

II. Road Marking Extraction Method

A. Road Surface Segmentation

The 3-D geospatial information of road surfaces, roadside buildings, trees, and other street-scene objects can be collected simultaneously by a mobile LiDAR system. However, our focus is only on road surfaces. Therefore, in order to reduce the quantity of the point cloud data to be processed and the time complexity of our proposed method, road surfaces should be segmented from the raw point cloud rapidly and accurately. In our previous study [4], we developed a curb-based road surface segmentation approach. This approach first partitions the point cloud into a set of data blocks along the trajectory, which is recorded by the on-board navigation system. Instead of processing an entire data block, a profile is generated perpendicularly to the trajectory within each data block. Then, through profile analysis, curb points are located within each profile by selecting the points on opposite sides of the trajectory with specific elevation gradients constrained by the predefined thresholds. Finally, the curb points ascertained from all profiles are fitted into curb-lines. Since curbs indicate the boundaries of the road, road surface points can be easily segmented from the raw point cloud based on the curb-lines. This approach operates very effectively and achieves high accuracy in road surface segmentation. Therefore, this curb-based road surface segmentation approach [4] is used in this study for preparing the road surface point cloud for road marking extraction and classification. The segmented road surface along with the ascertained curb points and fitted curb-lines are shown in Fig. 1.

B. Multisegment Thresholding

Road markings are highly reflective objects painted on road surfaces; therefore, they show higher intensities than their surrounding road surface in the point cloud (see Fig. 1). However, the reflected laser pulse intensities also depend on the incident angles of the emitted laser beams and the ranges between the measured target and the scanner center. Generally, the reflected laser pulse intensity decreases with the increase of incident angles and ranges. As a result, the road markings farther away from the scanner center exhibit relatively lower intensities than those of the road markings nearer to the scanner center. Therefore, the intensities of road markings distribute unevenly and fluctuate strongly in the road surface point cloud.

In order to reduce the impact of intensity unevenness on the extraction of road markings, we adopt a multisegment thresholding strategy. Instead of processing the entire road surface point cloud, we first partition it into a set of point cloud blocks along the road direction with a length of L_{b}. Then, each of the point cloud blocks is divided into a multisegment structure based on the trajectory. As shown in Fig. 2, centered at the trajectory, a point cloud block is partitioned into a set of

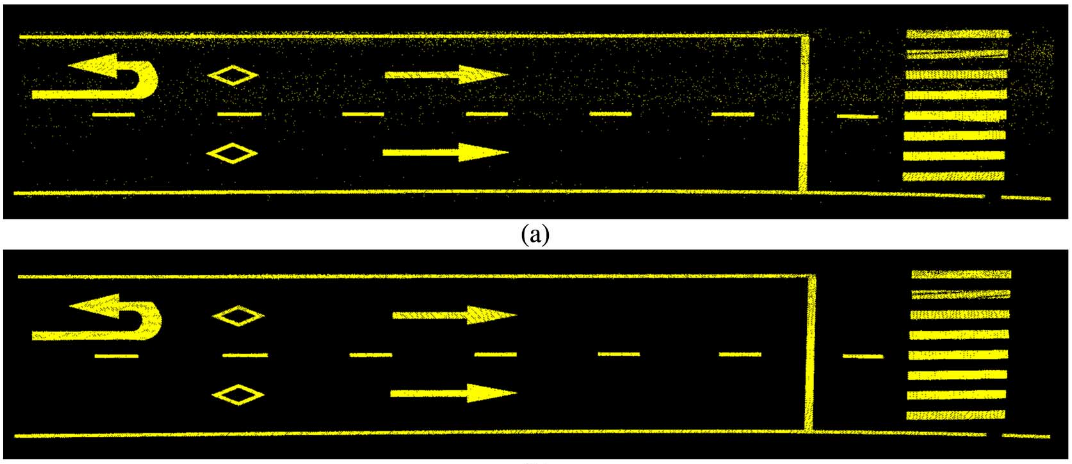


Fig. 3. Extracted road markings (a) after multisegment thresholding and (b) after spatial density filtering.

segments with a width w_s in opposite directions. Finally, all the segments are thresholded separately to extract road markings. Such a partition strategy can effectively smooth the intensity variations caused by the incident angle changes of laser beams.

In this paper, each of the partitioned segments is thresholded separately using the Otsu's thresholding algorithm [9]. Otsu's thresholding algorithm aims to find an optimal threshold that can maximize the ratio of between-class variance to within-class variance. In order to apply the Otsu's thresholding algorithm, the intensities of the points within each segment are first normalized into the gray level range of [0, 255]. Then, the between-class variance is defined as [9]

$$\sigma_{\mathrm{B}}^2 = \omega_{\mathrm{R}}\omega_{\mathrm{M}}(\mu_{\mathrm{M}} - \mu_{\mathrm{R}})^2 \tag{1}$$

where ω_{R} and ω_{M} are the probabilities of the occurrences of road surface and road markings, respectively; μ_{R} and μ_{M} are the corresponding mean levels. Finally, the optimal intensity threshold is selected as follows:

$$G^* = \arg \max_{0 \le G \le 255} \sigma_{\mathrm{B}}^2(G). \tag{2}$$

After the threshold is calculated, the points with intensities higher than G^* are regarded as road markings, whereas the others are treated as the road surface. By using the Otsu's thresholding algorithm, road markings in all segments are automatically and separately extracted. The extracted road markings are shown in Fig. 3(a).

C. Spatial Density Filtering

Caused by the characteristics of the backscattered reflectance and the material properties of the bituminous roads, some parts of the road surface exhibit similar intensities to those of road markings. Therefore, as shown in Fig. 3(a), the extracted road markings contain considerable noise after multisegment thresholding. However, compared to road marking points, these noise distribute dispersedly and irregularly. To effectively remove these noise from the extracted road markings, we develop a spatial density filter. This filter can efficiently distinguish road marking points from noise through calculating the spatial density of each point. The spatial density of a point $p(x, y, z)$ is defined as follows:

$$SD(p) = 1 + \sum_{p_i \in N(p)} \exp\left(-\frac{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}{(d_{\mathrm{N}}/3)^2}\right) \tag{3}$$

where $N(p)$ denotes the local neighborhood of point p, d_{N} is the size of the neighborhood, and $p_i(x_i, y_i, z_i)$ is a point within the neighborhood. By such a definition, the spatial density of a point depends on two factors: 1) the number of points within its neighborhood and 2) the distribution properties of these points. The more the points located near to a point, the higher the spatial density of this point. Therefore, the spatial density of the noise is lower than those of the road marking points. After the spatial densities of the points are computed, the points with spatial densities lower than a threshold ρ_{SD} are regarded as noise and further filtered out. Fig. 3(b) shows the extracted road markings after spatial density filtering.

III. Road Marking Classification Method

After road markings are extracted from the road surface point cloud, a classification method is applied to delineate the extracted road markings into specific categories. The road marking classification method includes: 1) Euclidean distance clustering for grouping unorganized road marking points into individual road markings; 2) voxel-based normalized cut segmentation for segmenting road marking clusters containing more than one type of road markings; 3) large-size marking classification based on trajectory and curb-lines; 4) deep learning-based small-size marking classification; and 5) PCA-based rectangular-shaped marking classification.

A. Euclidean Distance Clustering

Since the extracted road marking points are isolated and unorganized, they should be grouped into clusters, representing individual road markings, before carrying out classification. In

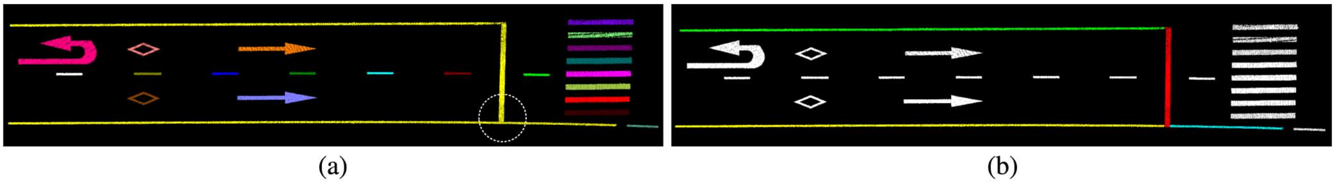


Fig. 4. (a) Clustered individual road markings and (b) segmentation result using voxel-based normalized cut segmentation.

this paper, we adopt a Euclidean distance clustering approach, which clusters points based on the Euclidean distances to their neighbors. Theoretically, an unclustered point is embedded into a specific cluster if and only if its shortest Euclidean distance to the points in this cluster lies below a threshold d_c. Otherwise, a new cluster is created for this point. Fig. 4(a) shows the clustered individual road markings using the Euclidean distance clustering approach. Different colors represent different road marking clusters.

B. Voxel-Based Normalized Cut Segmentation

As shown by the white dashed circle in Fig. 4(a), this road marking cluster actually contains two types of road markings: stop line and boundary line. Since they are orthographic and overlapped with each other, Euclidean distance clustering cannot separate them. To effectively segment such road marking clusters containing multiple types of road markings, we develop a voxel-based normalized cut segmentation method. First, a bounding box is calculated for each road marking cluster. Then, large-size road marking clusters are determined by selecting the clusters whose diagonal lengths of their bounding boxes exceed a predefined threshold D_b.

Considering road fluctuations and the efficiency of the segmentation method, a large-size road marking cluster that contains multiple types of road markings is partitioned into a voxel structure with a spacing of v_s. Then, these voxels are constructed into a fully connected weighted graph $\boldsymbol{G} = \{\boldsymbol{V}, \boldsymbol{E}\}$, where $\boldsymbol{V}$ takes the voxels as the nodes and $\boldsymbol{E}$ is formed between each pair of the nodes. In order to assign a meaningful weight to an edge for measuring the similarity between a pair of voxels, the point nearest to the voxel center is selected in each voxel for computing voxel features. The selected points are stored in $\boldsymbol{P}_\mathrm{voxel}$. Then, for a point $q \in \boldsymbol{P}_\mathrm{voxel}$, a covariance matrix is computed for it using its k-nearest neighbors $(p_1, p_2, \ldots p_k)$

$$\mathbf{C}_q = \frac{1}{k}\sum_{i=1}^{k}(p_i - q)\cdot(p_i - q)^T. \tag{4}$$

Through eigenvalue decomposition on the covariance matrix, the distribution feature of q is set to be the eigenvector associated with the largest eigenvalue. Denote the distribution feature of q as $\boldsymbol{v}_q$. The weight on the edge connecting a pair of nodes $(i, j) \in \boldsymbol{V}^2$ is defined as follows: [shown at bottom of the page (5)]

where $q_i, q_j \in \boldsymbol{P}_\mathrm{voxel}$; $\boldsymbol{v}_{q_i}$ and $\boldsymbol{v}_{q_j}$ are their corresponding distribution features; σ_D and σ_A are the standard deviations; and d_s is a distance constraint determining the maximal valid distance between two voxels.

Normalized cut segmentation [10] aims to partition the graph G into two disjoint voxel groups A and B by maximizing the similarity within each voxel group and minimizing the similarity between voxel groups. The cost function is defined as [10]

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)} \tag{6}$$

where $cut(A, B) = \sum_{u\in A, v\in B} w_{uv}$ denotes the sum of weights between voxel groups A and B; $assoc(A, V) = \sum_{u\in A, v\in V} w_{uv}$ represents the sum of weights on the edges ending in voxel group A; $assoc(B, V) = \sum_{u\in B, v\in V} w_{uv}$ represents the sum of weights on the edges ending in voxel group B. The minimization of $Ncut(A, B)$ is achieved by solving the corresponding generalized eigenvalue problem [10]

$$(\mathbf{D} - \mathbf{W})y = \lambda\mathbf{D}y \tag{7}$$

where $\mathbf{W}(i, j) = w_{ij}$ and $\mathbf{D}$ is a diagonal matrix with $\mathbf{D}(i, i) = \sum_m w_{im}$. Finally, the road marking cluster is segmented into different parts using the eigenvector associated with the second smallest eigenvalue. Fig. 4(b) shows the segmentation result.

C. Large-Size Marking Classification

Once the large-size road marking clusters containing multiple types of road markings are segmented, the large-size markings are first classified into three types: centerline, boundary line, and stop line. To this end, we develop a trajectory-curb-line-based method, which uses trajectory and curb-lines to classify large-size markings. Trajectory is recorded by the onboard navigation system and curb-lines are fitted from the extracted curb points in Section II-A. Therefore, these data are available to be used. Since, 1) trajectory indicates the direction of the road and 2) stop lines are usually painted orthographic to the direction of the road, stop lines can be easily distinguished from centerlines and boundary lines based on the trajectory. In addition, 1) curb-lines indicate the boundaries of the road; 2) centerlines and boundary lines are painted along the road;

$$w_{ij} = \begin{cases} \exp\left(-\frac{\|q_i - q_j\|_2^2}{\sigma_\mathrm{D}^2}\right)\cdot\exp\left(-\frac{\arccos^2\left(\boldsymbol{v}_{q_i}\cdot\boldsymbol{v}_{q_j}\right)}{\sigma_\mathrm{A}^2}\right), & \text{if } \|q_i - q_j\|_2 \le d_\mathrm{s} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

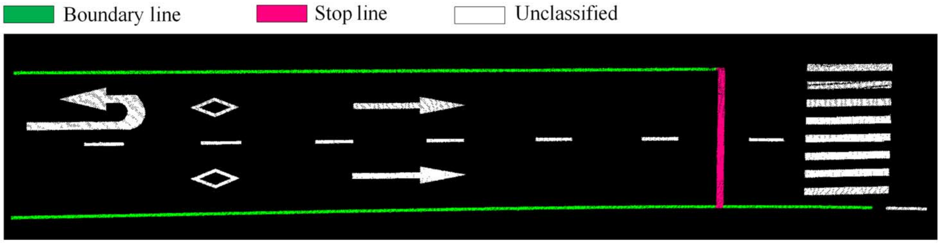


Fig. 5. Classification result of large-size markings.

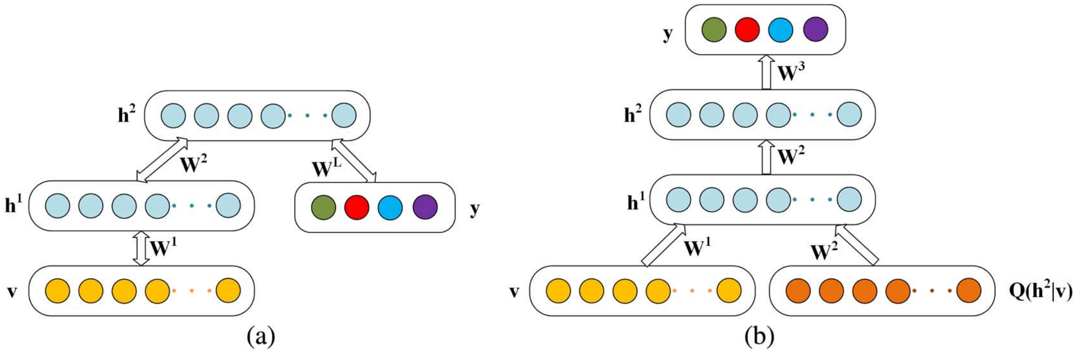


Fig. 6. (a) Jointly trained DBM model and (b) fine-tuned multilayer classifier.

and 3) centerlines are usually located between two boundary lines and farther away from the curb-lines than the boundary lines; therefore, centerlines and boundary lines can be easily distinguished based on the curb-lines.

For each large-size marking, we uniformly sample N_{L} points and compute their distribution features. If the percentage of the sampling points, whose distribution features are orthographic to the direction of the trajectory, lies above 95%, this marking is classified as stop line. If the percentage of the sampling points, whose distribution features are along the direction of the trajectory and shortest distances to the left and right curb-lines are below d_{L}, lies above 95%, this marking is classified as boundary line. Otherwise, if the percentage of the sampling points, whose distribution features are along the direction of the trajectory and shortest distances to the left and right curb-lines exceed d_{L}, lies above 95%, this marking is classified as centerline. Fig. 5 shows the classification result for large-size markings.

D. Small-Size Marking Classification

Recently, deep learning models [11]–[13] have been attractive for their high performance in learning hierarchical features from high-dimensional unlabelled data. By learning multilevel feature representations, deep learning models have been proved to be an effective tool for rapid object-oriented classification problems. Deep Boltzmann Machines (DBMs) [13], [14] have been an important breakthrough in the requirement for powerful deep learning models. A DBM is a layer-wise extension of the Restricted Boltzmann Machines (RBMs) [15], [16] with multiple hidden layers.

To efficiently classify small-size markings, we first jointly learn a two-layer DBM [see Fig. 6(a)] using a set of manually selected and labeled training samples. Let $\boldsymbol{v}$ be a vector of binary visible units that represent a road marking image; let $\boldsymbol{y}$ be a binary label vector that contains just one 1; and let $\boldsymbol{h}^1$ and $\boldsymbol{h}^2$ be the lower and higher layer binary hidden units, respectively. Since only four types of small-size markings are considered: rectangular-shaped marking, pedestrian warning marking, arrow marking, and other markings, $\boldsymbol{y}$ can only take the values of $[1,0,0,0]^T$, $[0,1,0,0]^T$, $[0,0,1,0]^T$, and $[0,0,0,1]^T$. The energy of the state $\{\boldsymbol{v},\boldsymbol{y},\boldsymbol{h}^1,\boldsymbol{h}^2\}$ is

$$E(\boldsymbol{v},\boldsymbol{y},\boldsymbol{h}^1,\boldsymbol{h}^2;\theta)=-\boldsymbol{v}^T\mathbf{W}^1\boldsymbol{h}^1-(\boldsymbol{h}^1)^T\mathbf{W}^2\boldsymbol{h}^2-\boldsymbol{y}^T\mathbf{W}^L\boldsymbol{h}^2 \tag{8}$$

where $\theta=\{\mathbf{W}^1,\mathbf{W}^2,\mathbf{W}^L\}$ are the model parameters. $\mathbf{W}^1$, $\mathbf{W}^2$, and $\mathbf{W}^L$ represent visible-to-hidden, hidden-to-hidden, and label-to-hidden symmetric interaction terms, respectively. The probability that the model assigns to a vector $\boldsymbol{v}$ with a label $\boldsymbol{y}$ is

$$P(\boldsymbol{v},\boldsymbol{y};\theta)=\frac{1}{Z(\theta)}\sum_{\boldsymbol{h}^1}\sum_{\boldsymbol{h}^2}\exp(-E(\boldsymbol{v},\boldsymbol{y},\boldsymbol{h}^1,\boldsymbol{h}^2;\theta)) \tag{9}$$

where $Z(\theta)=\sum_{\boldsymbol{v}}\sum_{\boldsymbol{y}}\sum_{\boldsymbol{h}^1}\sum_{\boldsymbol{h}^2}\exp(-E(\boldsymbol{v},\boldsymbol{y},\boldsymbol{h}^1,\boldsymbol{h}^2;\theta))$ is the partition function [13]. The conditional distributions over the visible, label, and two sets of hidden units are

$$p(h_j^1=1|\boldsymbol{v},\boldsymbol{h}^2)=g\left(\sum_i W_{ij}^1 v_i+\sum_m W_{jm}^2 h_m^2\right) \tag{10}$$

$$p(h_m^2=1|\boldsymbol{h}^1,\boldsymbol{y})=g\left(\sum_j W_{jm}^2 h_j^1+\sum_k W_{km}^L y_k\right) \tag{11}$$

$$p(v_i=1|\boldsymbol{h}^1)=g\left(\sum_j W_{ij}^1 h_j^1\right) \tag{12}$$

$$p(y_k|\boldsymbol{h}^2)=\frac{\exp\left(\sum_m W_{km}^L h_m^2\right)}{\sum_s \exp\left(\sum_m W_{sm}^L h_m^2\right)} \tag{13}$$

where $g(x)=1/(1+\exp(-x))$ is the logistic function [14].

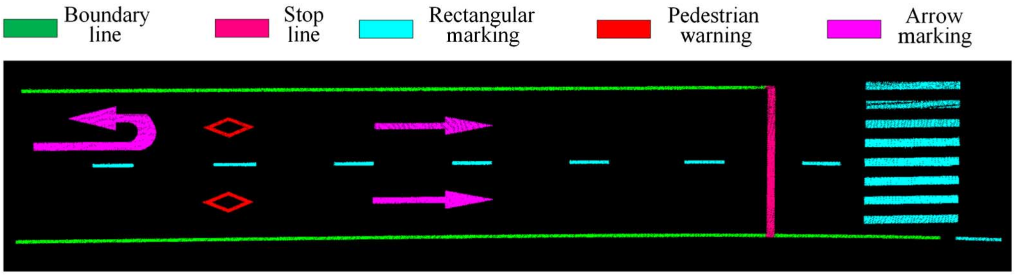


Fig. 7. Small-size marking classification result.

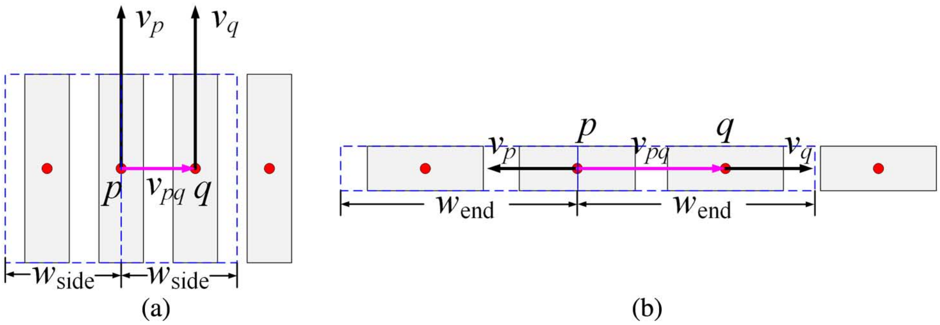


Fig. 8. Two arrangement modes: (a) side-by-side and (b) end-to-end.

Exact maximum likelihood learning in this joint DBM model is intractable. To effectively learn this model, a greedy layer-wise pre-training [17] is carried out to initialize the model parameters θ. Then, a joint learning algorithm that uses variational and stochastic approximation approaches [14] is adopted to jointly learn the model parameters.

After the joint DBM model is learned, the stochastic activities of the binary features in each hidden layer are replaced by deterministic, real-valued probabilities to construct a multilayer classifier [see Fig. 6(b)]. For each input $\boldsymbol{v}$, the mean-field inference [13] is used to obtain an approximate posterior distribution $Q(\boldsymbol{h}^2|\boldsymbol{v})$. The marginal $q(h_m^2 = 1|\boldsymbol{v})$ of this approximate posterior, together with the input data, are used to create an augmented input for this multilayer classifier. Then, standard back-propagation of error derivatives [13] is carried out to fine-tune the multilayer classifier. The output $\boldsymbol{y}$ of the multilayer classifier is computed as follows:

$$(\boldsymbol{h}^2)^T = g(g(\boldsymbol{v}^T\mathbf{W}^1 + Q(\boldsymbol{h}^2|\boldsymbol{v})^T(\mathbf{W}^2)^T)\mathbf{W}^2) \qquad (14)$$

$$y_k = \frac{\exp(\sum\limits_m h_m^2 W_{mk}^3)}{\sum\limits_{s=1}^{4}\exp(\sum\limits_m h_m^2 W_{ms}^3)}, \qquad k = 1, 2, 3, 4. \qquad (15)$$

Then, the class label L^* is determined by

$$L^* = \arg\max_k y_k. \qquad (16)$$

In order to classify the small-size markings using the multilayer classifier, we first rasterize each of the small-size markings into a 2-D binary image with a size of $n \times n$. Here, $n \times n$ equals to the dimension of $\boldsymbol{v}$. Then, the binary image is used as the input to the multilayer classifier. The output $\boldsymbol{y}$ of the multilayer classifier determines the type of this road marking. That is, the values of $[1, 0, 0, 0]^T$, $[0, 1, 0, 0]^T$, $[0, 0, 1, 0]^T$, and $[0, 0, 0, 1]^T$, respectively, indicate rectangular-shaped marking, pedestrian warning marking, arrow marking, and other markings. Fig. 7 shows the classification result of small-size markings.

E. Rectangular-Shaped Marking Classification

As shown in Fig. 7, the rectangular-shaped markings contain several types of road markings such as zebra crossing, dashed centerline, and dashed boundary line. To further classify the rectangular-shaped markings, we develop a PCA-based method for judging the arrangement mode of the rectangular-shaped markings. Basically, as shown in Fig. 8, there are two kinds of arrangement modes: side-by-side and end-to-end. Zebra crossing follows the side-by-side mode, whereas dashed centerline and boundary line follow the end-to-end mode.

For a rectangular-shaped marking P, we first calculate the coordinates of its centroid on the XY plane as follows:

$$p = \frac{1}{N_P}\sum_{i=1}^{N_P} p_i^{\text{XY}}, \quad p_i \in P \qquad (17)$$

where N_P is the number of points in marking P and p_i^{XY} denotes the coordinates of point p_i on the XY plane. Then, a covariance matrix is computed for p as follows:

$$\mathbf{C}_p = \frac{1}{N_P}\sum_{i=1}^{N_P}\left(p_i^{\text{XY}} - p\right)\cdot\left(p_i^{\text{XY}} - p\right)^T, \quad p_i \in P. \qquad (18)$$

Through eigenvalue decomposition on $\mathbf{C}_p$, the eigenvector associated with the largest eigenvalue is selected as the

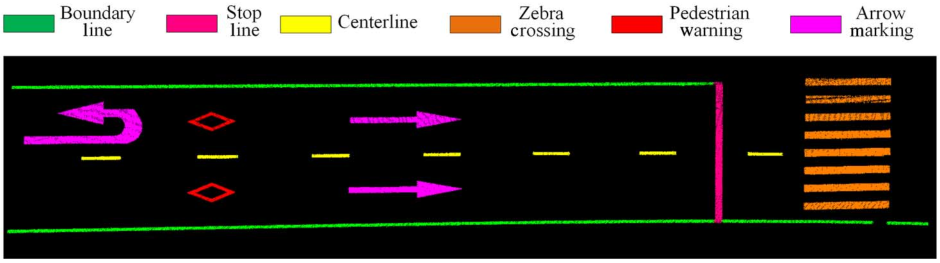
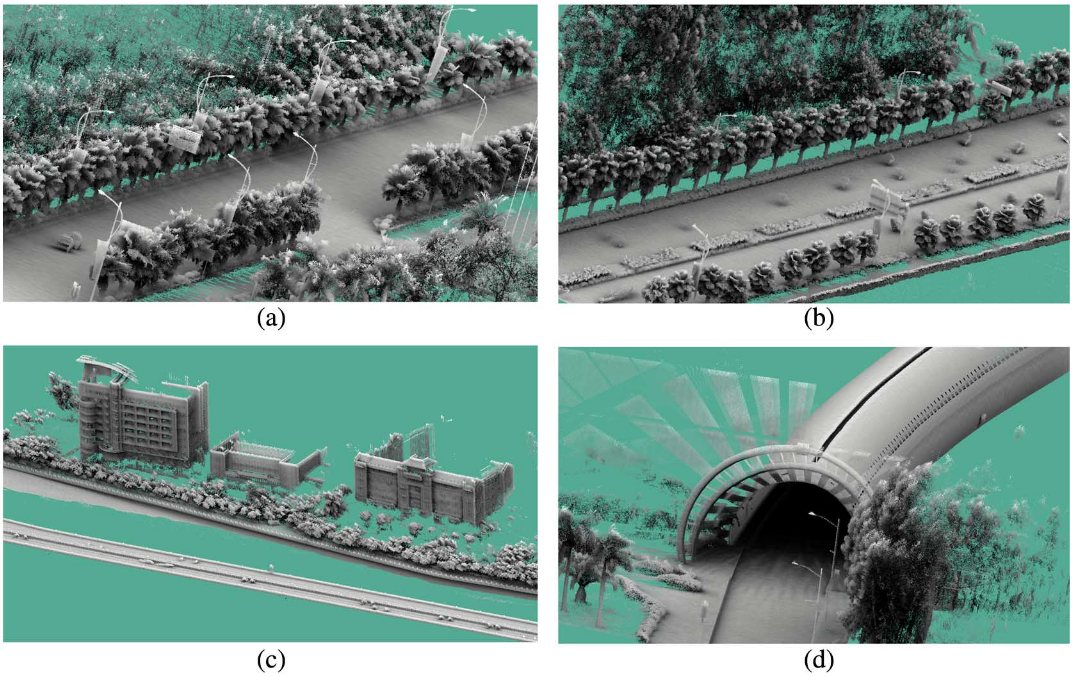


Fig. 9. Road marking classification result.



(a) (b)

(c) (d)

Fig. 10. Profiles of (a) SRS dataset; (b) ICEC dataset; (c) YWB dataset; and (d) YDT dataset.

distribution feature of marking P. Denote it as v_p. v_p actually describes the orientation of the rectangular-shaped marking P. To judge the arrangement mode of marking P, we first apply the side-by-side model by searching a rectangular region of width w_{side} along both sides of marking P [see Fig. 8(a)]. If a rectangular-shaped marking Q, whose centroid on the XY plane is q and distribution feature is v_q, is obtained within the searching region and the following criterion meets, marking P is labeled as zebra crossing

$$(v_p \,//\, v_q) \wedge (v_p \perp v_{pq}) \wedge (v_q \perp v_{pq}) \Rightarrow \text{side-by-side mode} \tag{19}$$

TABLE I

DESCRIPTIONS OF THE FOUR MOBILE LIDAR DATASETS

Dataset	Number of points	Length (m)
SRS	102 818 867	977
ICEC	96 027 613	719
YWB	69 422 952	1510
YDT	109 690 724	1059

where v_{pq} denotes the direction vector starting at point p and ending at point q. The criterion in (19) defines the following three constraints to judge the side-by-side arrangement mode: 1) v_p is parallel to v_q; 2) v_p is perpendicular to v_{pq}; and 3) v_q is perpendicular to v_{pq}. Otherwise, the end-to-end model is applied by searching a rectangular region of width w_{end} along both ends of marking P [see Fig. 8(b)]. If a rectangular-shaped marking Q, whose centroid on the XY plane is q and distribution feature is v_q, is obtained within the searching region and the following criterion meets:

$$(v_p \,//\, v_q) \wedge (v_p \,//\, v_{pq}) \wedge (v_q \,//\, v_{pq}) \Rightarrow \text{end-to-end mode} \tag{20}$$

marking P is further judged by the trajectory-curb-line-based method mentioned in Section III-C to distinguish centerlines from boundary lines. The criterion in (20) defines the following three constraints to judge the end-to-end arrangement mode: 1) v_p is parallel to v_q; 2) v_p is parallel to v_{pq}; and 3) v_q is parallel to v_{pq}. When handling curved roads, the parallel and

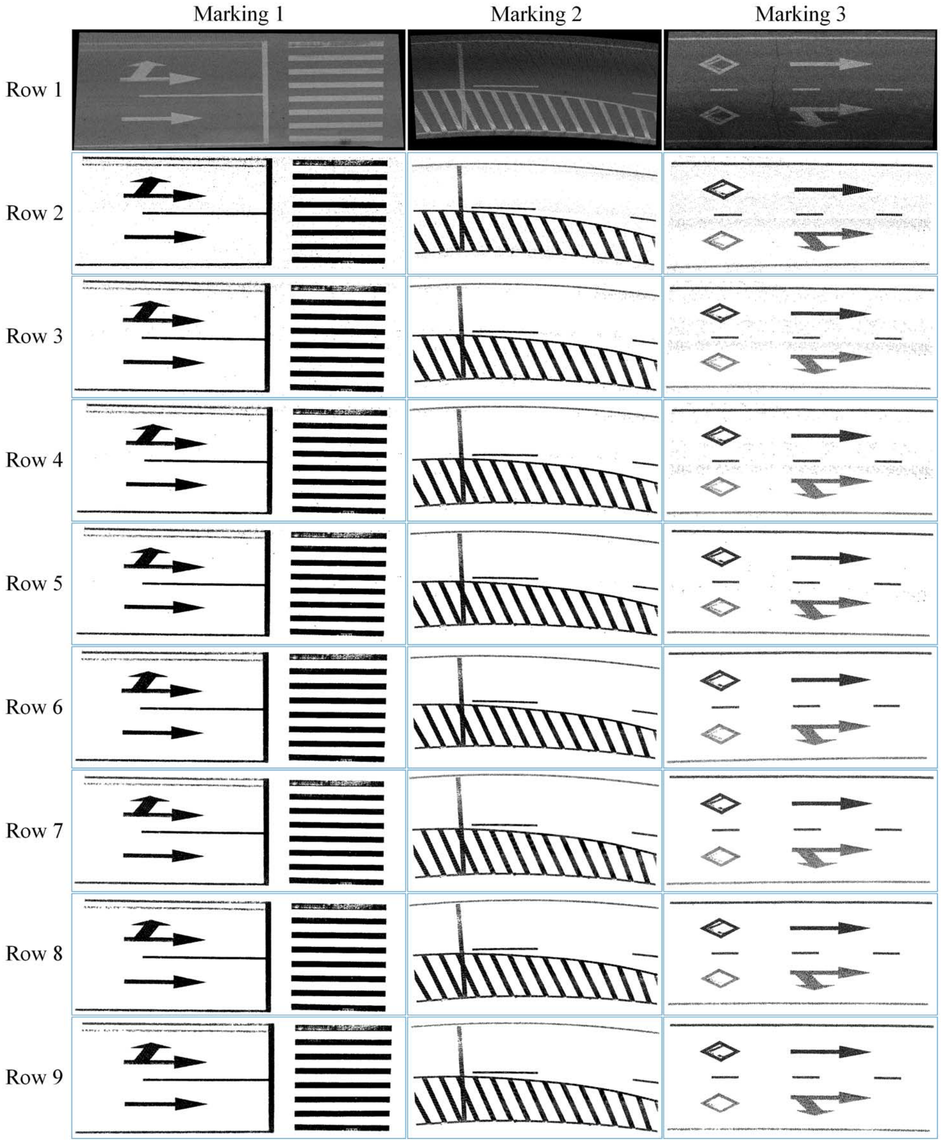


Fig. 11. Illustration of testing different configurations of ρ_{SD} on the results of noise removal. Row 1: raw point clouds. Row 2: extracted road marking points after multisegment thresholding. Rows 3–9: noise removal results through spatial density filtering with $\rho_{\mathrm{SD}} = 5$, 6, 7, 8, 9, 10, and 11, respectively.

perpendicular criteria in (19) and (20) are relaxed to have an angle deviation of 5°. The final classification result is shown in Fig. 9.

IV. Results and Discussion

A. RIEGL VMX-450 and Mobile LiDAR Datasets

The mobile LiDAR point cloud data used in this study were acquired using a RIEGL VMX-450 mobile LiDAR system in Xiamen, China. The VMX-450 system is integrated with two full-view VQ-450 laser scanners, four high-resolution digital cameras, an IMU, two global navigation satellite system (GNSS) antennas, and a wheel-mounted distance measurement indicator (DMI). The two VQ-450 laser scanners are configured with an "X" pattern for providing more coverage of the measured targets and reach a maximum range of 800 m. The mapping vehicle was driving at an average speed of 50 km/h while collecting data (point cloud, images, and trajectory). The accuracy of the scanned point cloud data is within 8 mm with a maximum effective measurement rate of 1.1 million measurements per second, and a line scan speed of up to 400 scans per second.

Fig. 10 and Table I present the four point cloud datasets, selected and used in this study, which cover the segments of 1) Siming Road South (SRS); 2) International Conference and Exhibition Center (ICEC); 3) Yanwu Bridge (YWB); and 4) Yunding Tunnel (YDT).

B. Parameter Sensitivity Analysis

The spatial density threshold ρ_{SD} for removing noise is very dependent on the characteristics of the point cloud data being used. Basically, different mobile LiDAR systems generate different maximum effective measurement rates and different line scan speeds. Therefore, the resultant point cloud data

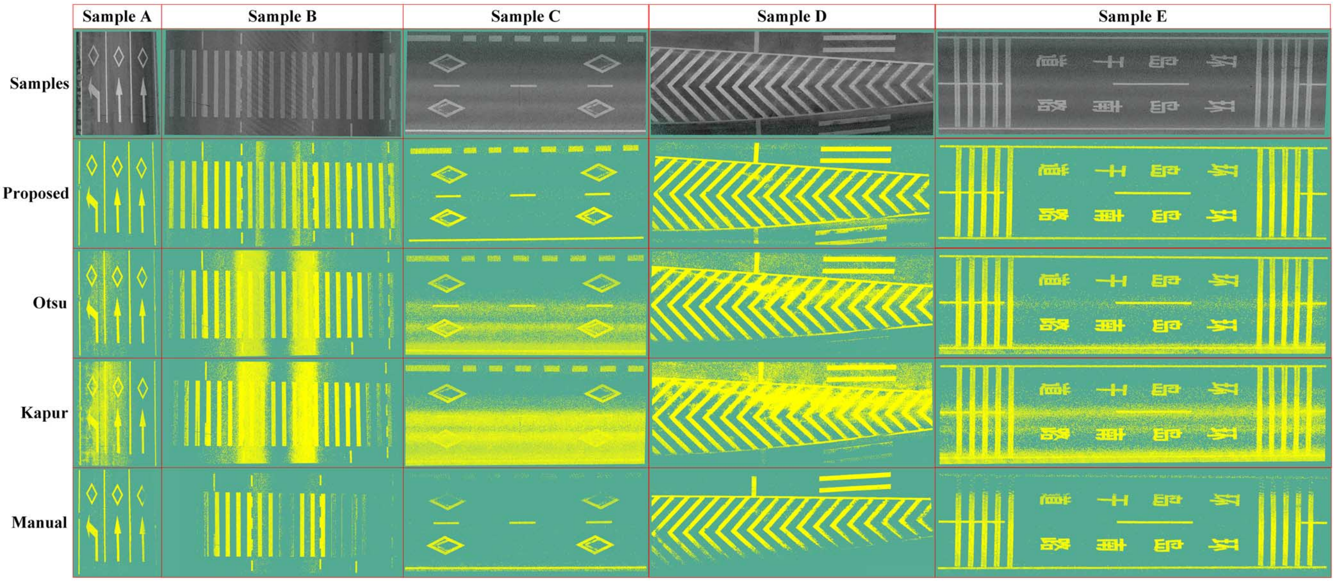


Fig. 12. Extracted road markings using different thresholding methods.

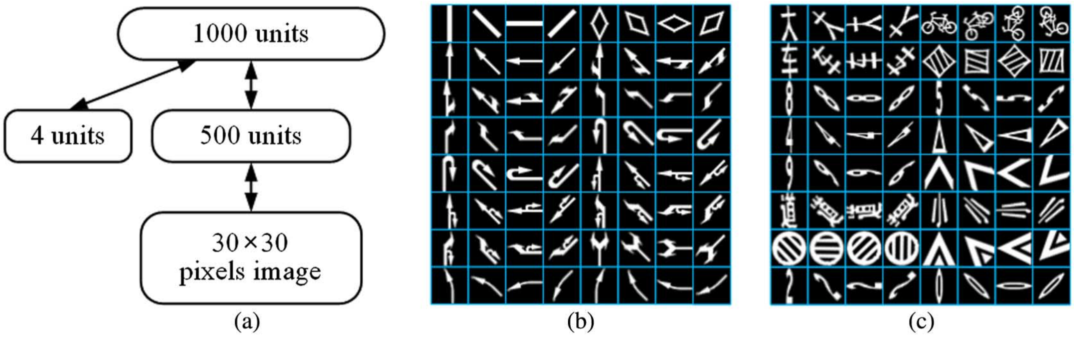


Fig. 13. (a) Architecture of the joint DBM; (b) and (c) subsets of training samples.

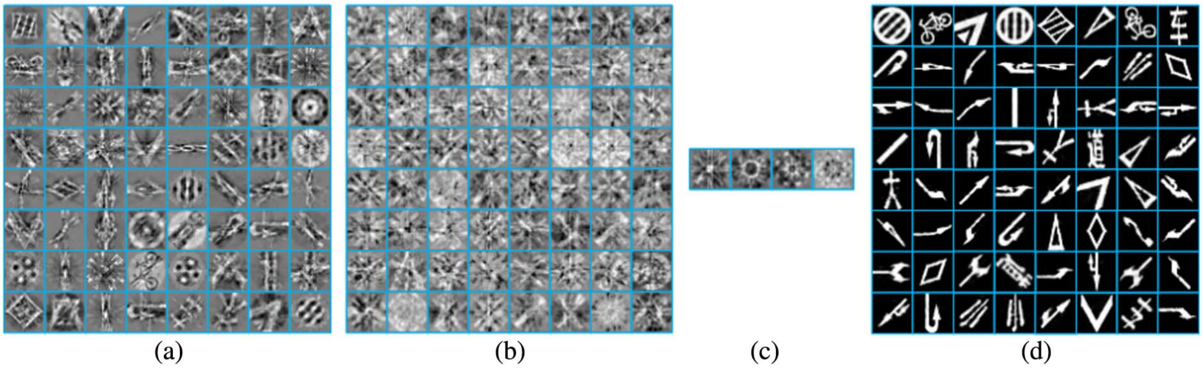


Fig. 14. (a) First-layer DBM features; (b) second-layer DBM features; (c) class-specific features; and (d) samples generated from the two-layer DBM by running the Gibbs sampler for 10 000 steps.

TABLE II
PARAMETER CONFIGURATIONS IN ROAD MARKING EXTRACTION AND CLASSIFICATION

Notation	L_b	w_s	d_N	ρ_{SD}	d_c	D_b	v_s	σ_D
Value	50 m	2 m	0.1 m	8	5 cm	8 m	0.1 m	100 m
Notation	σ_A	d_s	N_L	d_L	$n{\times}n$	w_{side}	w_{end}	
Value	π/2	50 m	100	1 m	30×30	1 m	3 m	

are of different point densities. Moreover, the driving speed of the mapping vehicle also results in point density variations. In this section, to ascertain an optimal configuration for ρ_{SD}, we tested a set of parameter configurations on noise removal. The testing results are shown in Fig. 11. As seen from the noise removal results, when ρ_{SD} is smaller than 8, noise are not well suppressed. However, when ρ_{SD} is greater than 8,

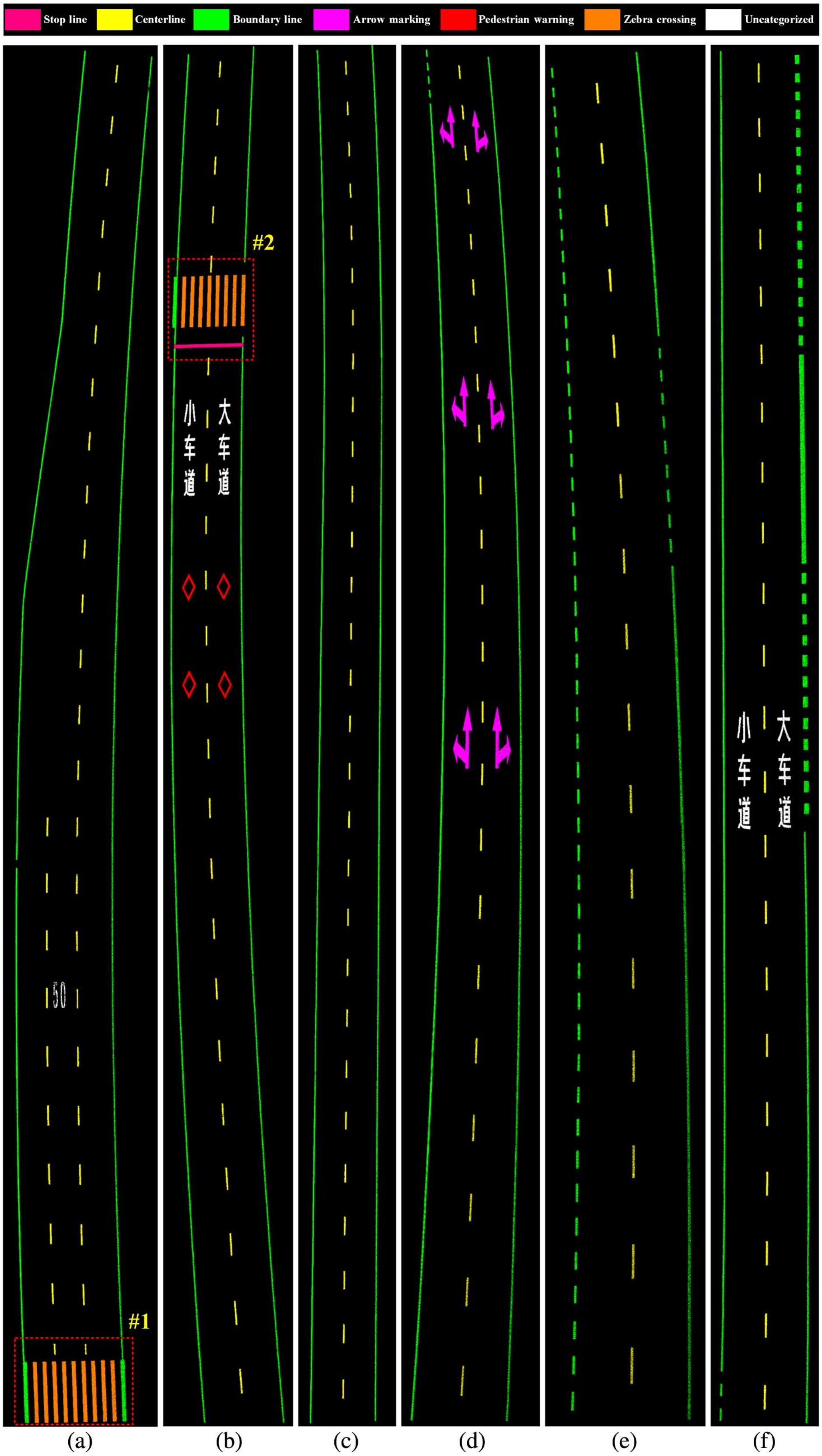


Fig. 15. (a)–(f) Road marking extraction and classification results on six road sections of the SRS dataset. Different colors represent different marking types.

some marking points are falsely regarded as noise and removed, especially at the corners and on the borders of the road markings, where the point densities are lower than those of the internal road marking points. Thus, to simultaneously suppress noise and preserve the completeness of road markings, we set $\rho_{\mathrm{SD}} = 8$ in our experiments.

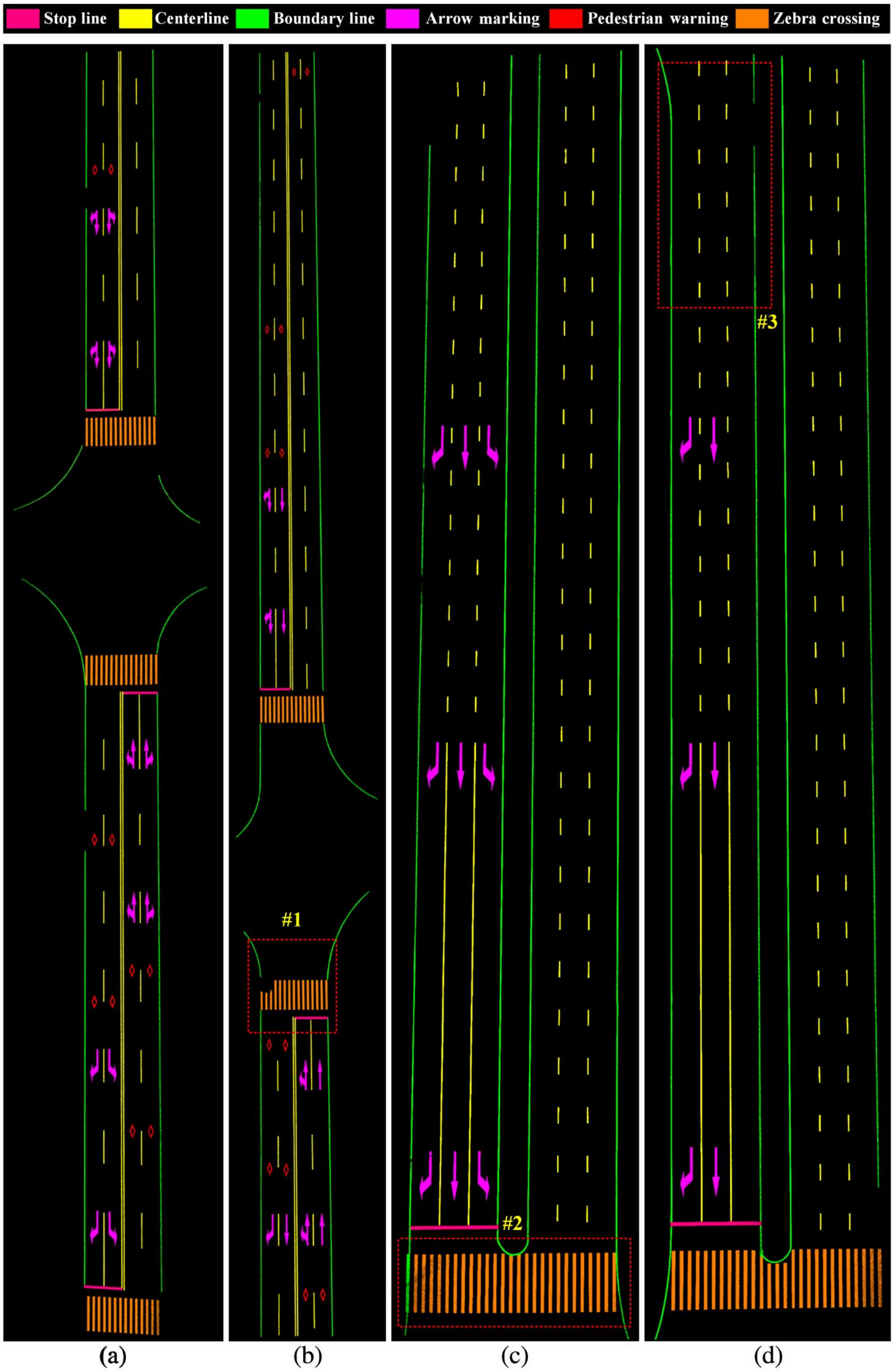


Fig. 16. (a)–(d) Road marking extraction and classification results on four road sections of the ICEC dataset. Different colors represent different marking types.

C. *Multisegment Thresholding*

To test the performance of our proposed multisegment thresholding method, we compared it with other well-known thresholding methods: Otsu method [9] and Kapur method [18]. As shown in the first row in Fig. 12, five road surface point clouds with different types of road markings were selected for comparing the performances of these methods. For both Otsu and Kapur methods, a globally optimal threshold that well separates the foreground from the background was determined for binarization. For our multisegment thresholding method, the point cloud was first partitioned into several segments according to the trajectory. Then, these segments were thresholded locally and separately. To further test the performance of a global threshold for separating road markings from the background road surface, a global threshold was manually selected for each point cloud to extract road markings with more details and less noise. The road marking extraction results are shown in the second row to the last row. As seen

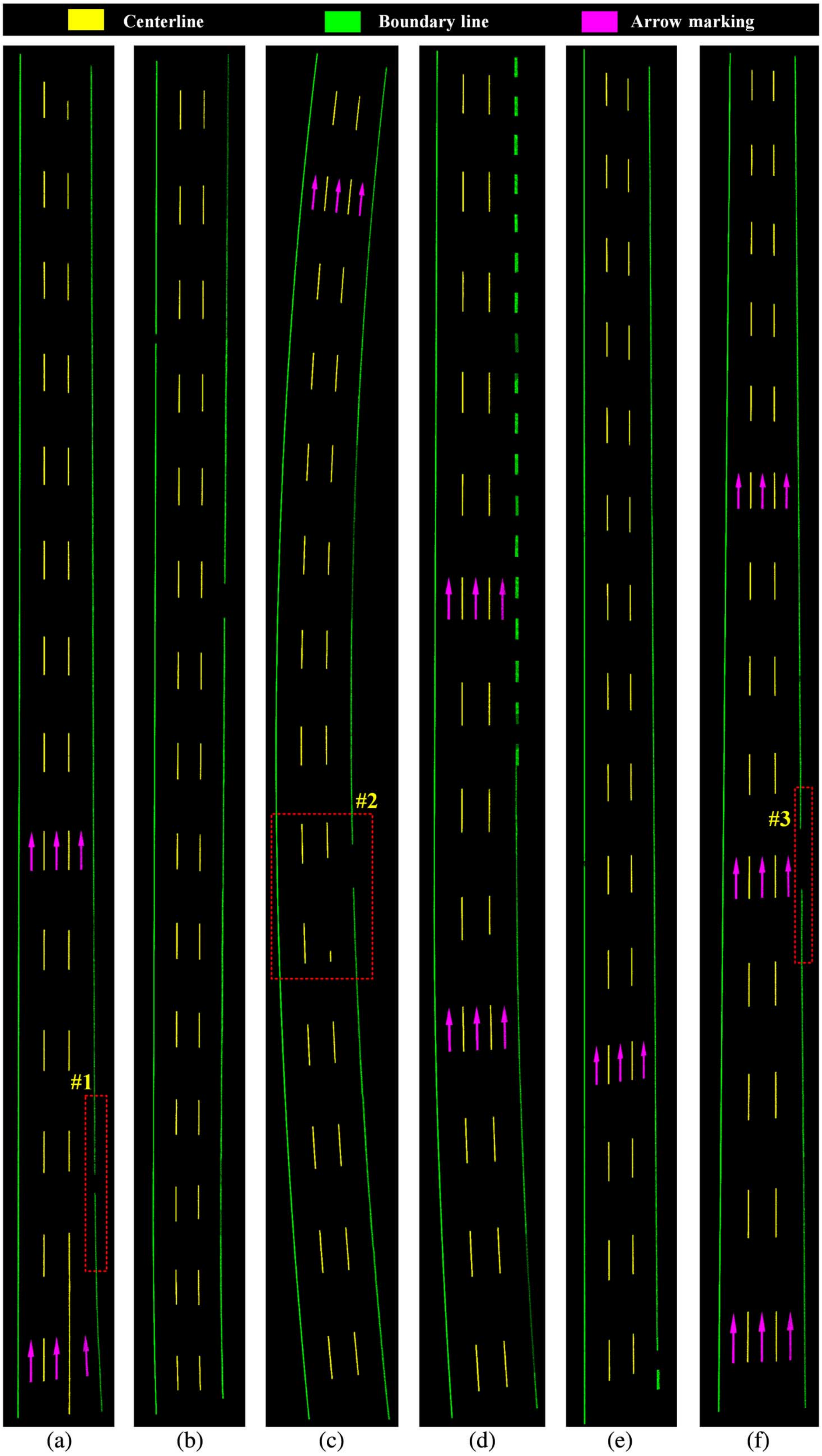


Fig. 17. (a)–(f) Road marking extraction and classification results on six road sections of the YWB dataset. Different colors represent different marking types.

from the extraction results, multisegment thresholding method extracts road markings more accurately and completely and generates less noise; whereas, by using the other three methods, the road markings are extracted incompletely or even missing, and a great portion of noise exhibits on one side of the point clouds. Therefore, the proposed multisegment thresholding

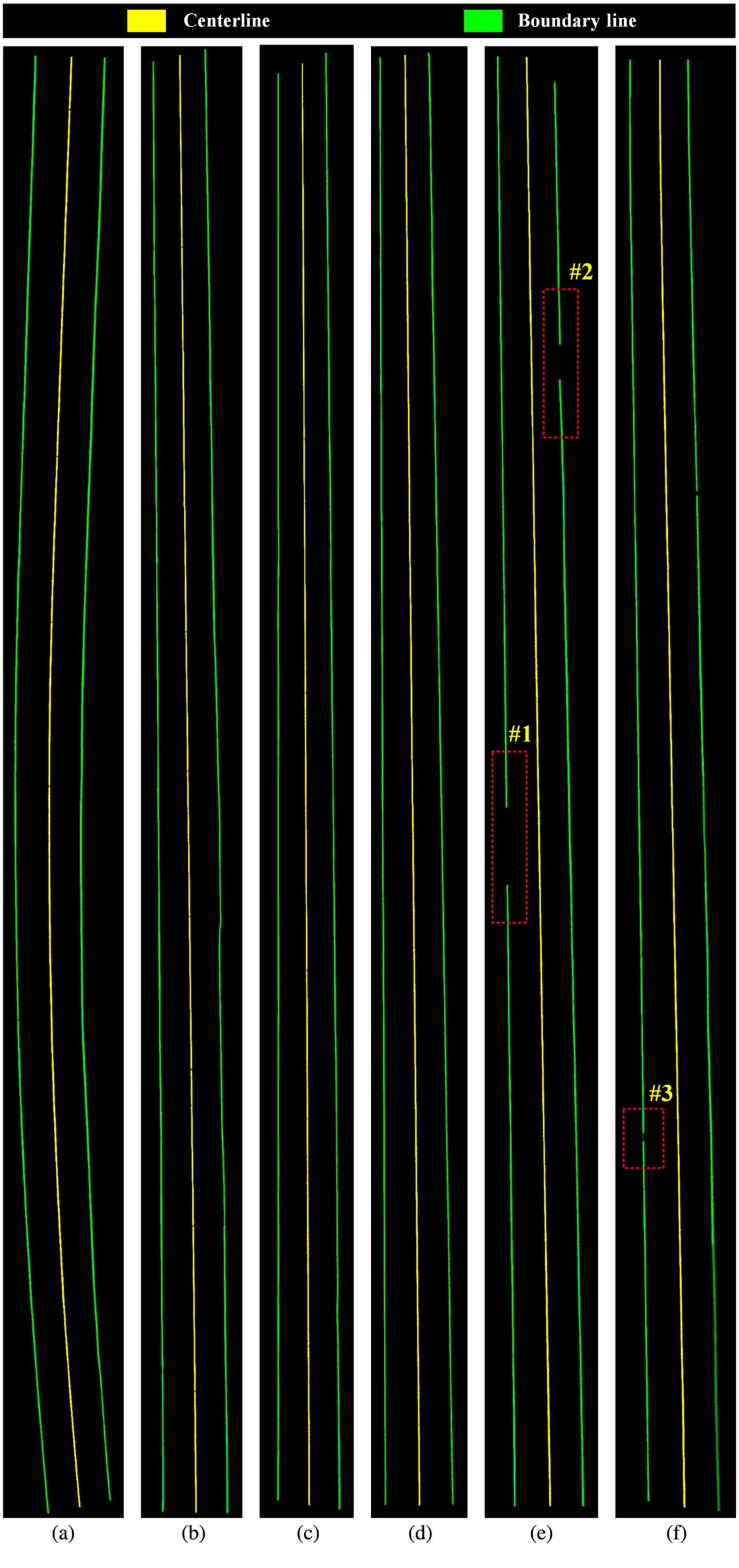


Fig. 18. (a)–(f) Road marking extraction and classification results on six road sections of the YDT dataset. Different colors represent different marking types.

TABLE III
COMPUTING TIME IN SECONDS FOR ROAD SEGMENTATION, ROAD MARKING EXTRACTION, AND CLASSIFICATION

Dataset	Road segmentation	Marking extraction	Marking classification	Total
SRS	22.79	28.39	34.33	85.51
ICEC	16.78	20.89	25.27	62.94
YWB	36.24	48.88	53.06	138.18
YDT	24.72	30.77	37.21	92.70

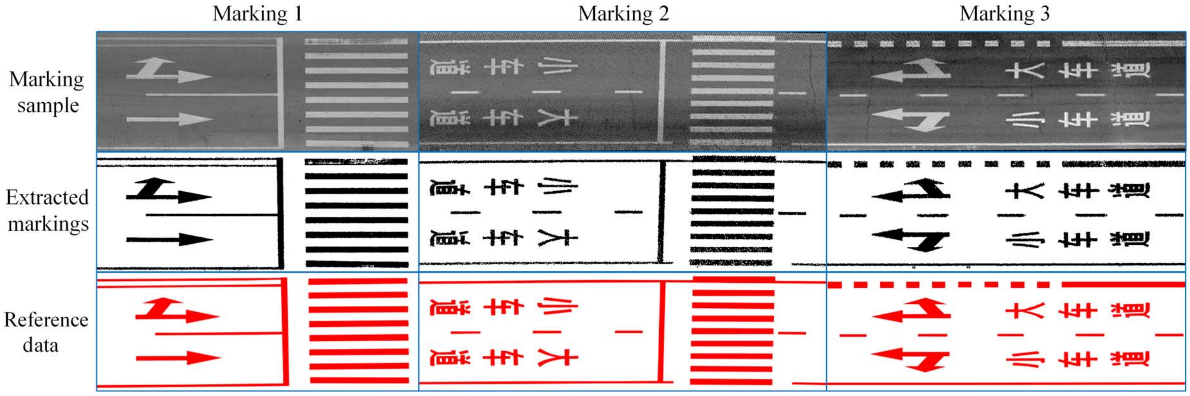


Fig. 19. Point cloud samples for quantitatively evaluating the performance of road marking extraction.

method outperforms the other three methods in completely and correctly extracting road markings.

D. Deep Learning

In order to effectively classify small-size markings, we jointly learnt a two-layer DBM [see Fig. 13(a)]. The visible input is a 30 × 30 pixels image, the first and second hidden layers contain 500 and 1000 units, respectively. Since we only consider four types of road markings including rectangular-shaped marking, pedestrian warning marking, arrow marking, and other markings, the label layer contains 4 units with "1-of-K" encoding [14]. Fig. 13(b) and (c) shows two subsets of training samples. After the joint DBM was learnt, we fine-tuned it to construct a three-layer classifier for small-size marking classification [see Fig. 6(b)]. Fig. 14(a) and (b) presents the learnt first-layer and second-layer features of the joint DBM, and Fig. 14(c) shows the class-specific features of the three-layer classifier.

To test the learning capability of the two-layer joint DBM on the labeled training samples, we fixed the label units with a specific class and randomly initialized the binary units in each layer of the DBM, and then run the Gibbs sampler according to (10), (11), and (12) for 10 000 steps. The probabilities of the binary visible units were used to construct a 30 × 30 pixels image. The randomly generated samples are shown in Fig. 14(d). Comparing the randomly generated samples with the training samples, we conclude that the two-layer joint DBM has a very good capability of learning salient and meaningful features from the labeled training samples.

E. Road Marking Extraction and Classification

The four datasets (SRS, ICEC, YWB, and YDT) mentioned in Section IV-A were used to examine the performance of the proposed road marking extraction and classification method. The parameters and their configurations are listed in Table II. Most of these parameters are determined based on prior knowledge or selected through a set of tests. Such configurations can provide good extraction and classification performance of 3-D road markings. These four datasets were first preprocessed to segment road surface point clouds from the raw mobile LiDAR point clouds using our previously developed curb-based approach [4]. Then, road marking points were extracted directly from the road surface point clouds through multisegment thresholding and spatial density filtering. Finally, the extracted road markings were further classified into seven marking types including stop line, centerline, boundary line, arrow marking, pedestrian warning marking, zebra crossing, and other markings. The road marking extraction and classification results of these four datasets are shown in Figs. 15–18, respectively.

TABLE IV
QUANTITATIVE EVALUATION RESULTS

Samples	Marking 1	Marking 2	Marking 3	Average
Completeness	0.94	0.93	0.91	0.93
Correctness	0.92	0.94	0.91	0.92
F-measure	0.93	0.94	0.91	0.93

As seen from the extraction results of these four datasets, road markings are basically completely and accurately extracted. However, since the road markings have been painted on the road surfaces for a long time, some parts have been worn by the moving vehicles and pedestrians, thereby resulting in incompleteness of road markings. Therefore, some of the road markings are incomplete in the extraction results in these four datasets. This is shown by the red rectangles #3 in Fig. 16, #1 and #3 in Fig. 17, and #3 in Fig. 18. In addition, caused

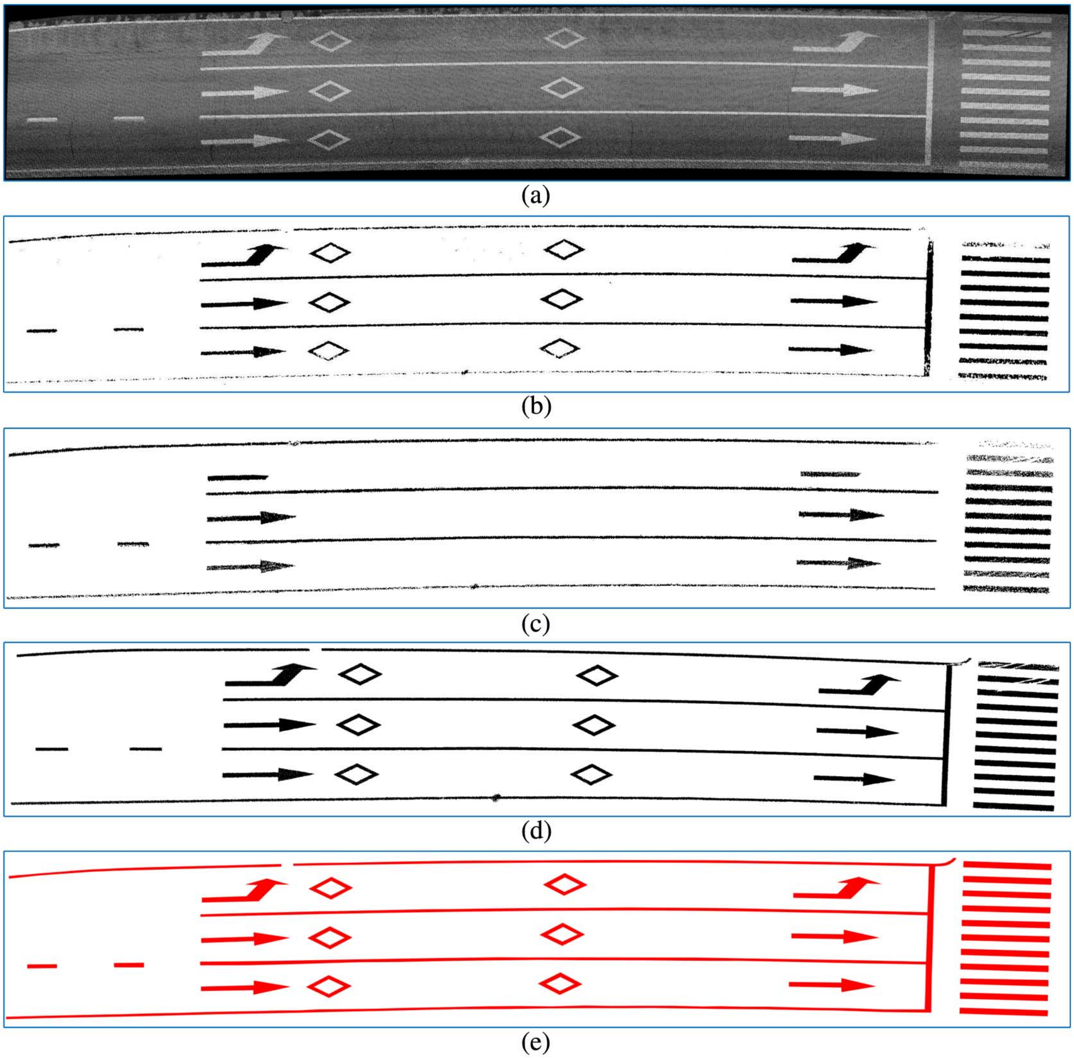


Fig. 20. Extracted road markings from Marking 1 dataset. (a) Road surface point cloud, extracted road markings using (b) Guan's method; (c) Chen's method; (d) our proposed method; and (e) manually labeled reference data.

by the occlusions of moving vehicles while acquiring mobile LiDAR point cloud data, laser beams cannot reach the shadows of the obstacles, thereby resulting in data incompleteness. This is another cause of the incompleteness of road markings in the extraction results, as shown by the red rectangles #1 in Fig. 16, #2 in Fig. 17, and #1 and #2 in Fig. 18. As a whole, the proposed road marking extraction method performs very well in extracting road markings directly from 3-D mobile LiDAR point clouds.

As seen from the road marking classification results of these four datasets, road markings are basically correctly classified into their corresponding types. However, at some road sections with zebra crossings, the strips of the zebra crossing near to the road curbs are often painted overlapped with the boundary lines, as shown by the red rectangle #1 in Fig. 15. Our voxel-based normalized cut segmentation method can effectively segment the stop line from the boundary line; however, it failed to segment such zebra crossing strips from the boundary lines. Therefore, some of the zebra crossing strips overlapped with the boundary lines were falsely classified as boundary lines. This is shown by the red rectangles #1 and #2 in Fig. 15, and #2 in Fig. 16. On the whole, the proposed road marking classification method can effectively classify both large-size and small-size road markings.

F. Time Complexity Analysis and Quantitative Evaluation

The proposed road marking extraction and classification method was implemented using C++ running on an Intel (R) Core (TM) i5-3470 computer. The computing time in each processing step was recorded for analyzing the time complexity of our proposed method. Table III lists the computing time of road surface segmentation, road marking extraction, and road marking classification for the four datasets. As seen from Table III, road marking classification takes more computing time than road marking extraction. The time complexity of our proposed method is quite low. This is benefited from the road surface segmentation step, which dramatically reduces the quantity of the point cloud data to be processed. Take the YWB dataset with about 69 million points and a length of 1510 m in the road direction as an example. Our proposed method took 36.24 s to segment the road surface point cloud from the raw mobile

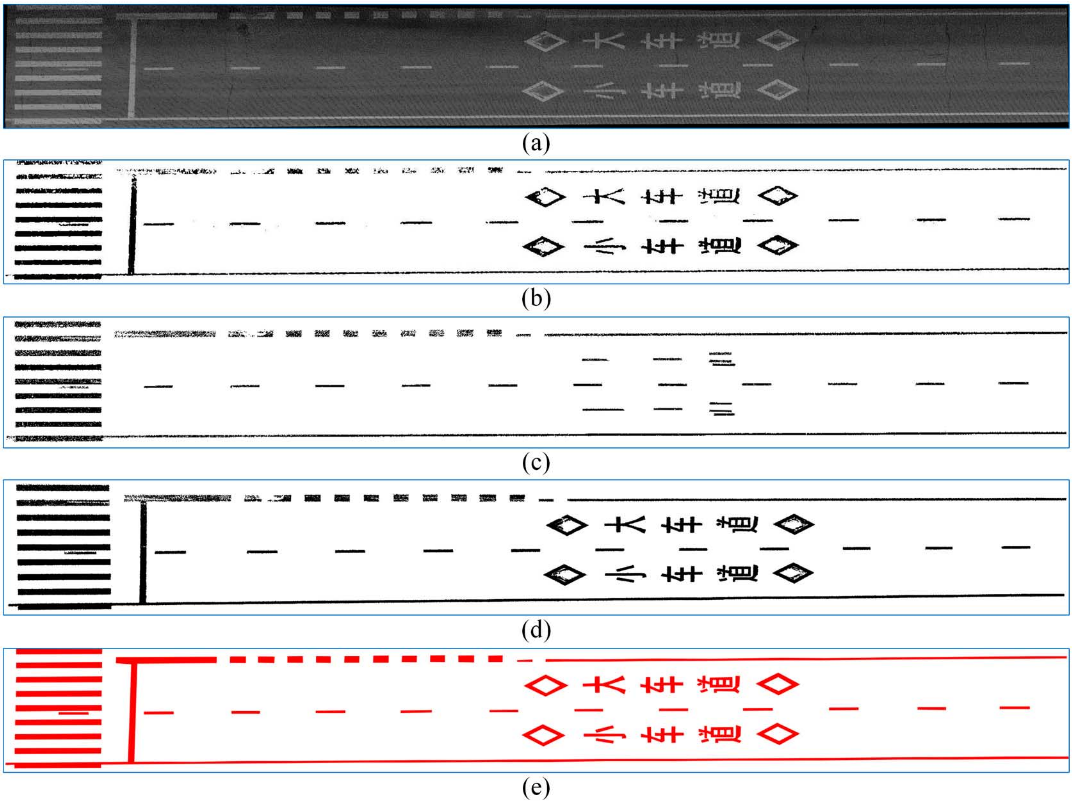


Fig. 21. Extracted road markings from Marking 2 dataset. (a) Road surface point cloud, extracted road markings using (b) Guan's method; (c) Chen's method; (d) our proposed method; and (e) manually labeling method.

LiDAR point cloud, 48.88 s to extract road markings from the segmented road surface point cloud, and 53.06 s to classify the extracted road markings into specific types. The total computing time for the YWB dataset is only 138.18 s. Therefore, our road marking extraction and classification method is capable of rapidly processing large volumes of mobile LiDAR point clouds.

A set of road surface point clouds was selected to quantitatively evaluate the performance of our road marking extraction method. After road markings were extracted, we rasterized both the road surface point clouds and the extraction results into 2-D geo-referenced feature images using the rasterization method in [4]. The rasterized road surface point clouds and the extraction results are shown in the first and second rows in Fig. 19, respectively. The manually labeled reference data based on the geo-referenced feature images in the first row are shown in the last row. To quantitatively evaluate the road marking extraction results, we used the following three measures: completeness, correctness, and F-measure [4]. These three measures are defined as follows:

$$\text{Completeness} = \frac{\text{TP}}{TP + FN} \tag{21}$$

$$\text{Correctness} = \frac{TP}{TP + FP} \tag{22}$$

$$F - \text{measure} = \frac{2 \times \text{Completeness} \times \text{Correctness}}{\text{Completeness} + \text{Correctness}} \tag{23}$$

where TP, FN, and FP are the numbers of true positives, false negatives, and false positives, respectively. As shown by the quantitative evaluations in Table IV, our proposed road marking extraction method achieves an average completeness, correctness, and F-measure of 0.93, 0.92, and 0.93, respectively, on the selected samples. Therefore, our road marking extraction method obtains high performance and extracts road markings with little commission and omission errors.

G. Comparative Study

A comparative study was conducted to compare our proposed road marking extraction method with two recently published methods: Guan's method [4] and Chen's method [8]. Guan's method uses 2-D geo-referenced feature imagery generated from 3-D mobile LiDAR point clouds to extract road markings, whereas Chen's and our proposed methods extract road markings directly from 3-D point clouds. The road surface point clouds selected for the comparative study and the road marking extraction results using different methods are shown in Figs. 20 and 21, respectively. Chen's method focuses only on the lane markings along the direction of the trajectory, whereas Guan's and our proposed methods can extract any types of road markings. However, our proposed method can extract road markings more completely and accurately than Guan's method. To further quantitatively evaluate the road marking extraction results obtained using different methods, the three measures including completeness, correctness, and F-measure were computed based on the manually labeled reference data [see Figs. 20(e) and 21(e)]. The quantitative evaluation results are listed in Table V. Comparatively, our proposed method outperforms the

TABLE V
QUANTITATIVE EVALUATION RESULTS OF DIFFERENT METHODS

Samples	Marking 1			Marking 2		
Method	Guan [4]	Chen [8]	Proposed	Guan [4]	Chen [8]	Proposed
Completeness	0.86	0.75	0.93	0.89	0.71	0.93
Correctness	0.90	0.91	0.92	0.91	0.92	0.91
F-measure	0.88	0.82	0.93	0.90	0.80	0.92

other two methods in completely and accurately extracting road markings from mobile LiDAR point clouds.

V. CONCLUSION

Extracting road markings directly from 3-D mobile LiDAR point clouds is a very challenging task. Most of the existing methods simplified such a task by converting 3-D point clouds into 2-D geo-referenced feature images. However, such a conversion may lead to incompleteness and incorrectness in feature extraction. In this paper, we have proposed a rapid and effective method for extracting and classifying road markings directly from 3-D mobile LiDAR point clouds. The proposed method has been applied to four mobile LiDAR point cloud datasets acquired by a RIEGL VMX-450 system. Time complexity analysis showed that the proposed method can handle large volumes of mobile LiDAR point clouds in a short time. Through quantitative evaluation, the proposed method achieved an average completeness, correctness, and F-measure of 0.93, 0.92, and 0.93, respectively. Moreover, comparative studies demonstrated that the proposed method outperforms the other two methods and extracts road markings more accurately and completely with less commission and omission errors. In addition, instead of processing the 2-D geo-referenced feature imagery, the proposed method extracts road markings directly from 3-D mobile LiDAR point clouds. Therefore, the geospatial information of the road markings is preserved and can be used in other applications. However, the limitation of the proposed method exists in the classification of dashed centerlines and dashed boundary lines when handling curved roads. As an alternative solution, the side-by-side and end-to-end arrangement modes can be relaxed to have an angle deviation to improve the classification performance.

ACKNOWLEDGMENT

The authors would like to acknowledge the anonymous reviewers for their valuable comments.

REFERENCES

[1] R. Gopalan, T. Hong, M. Shneier, and R. Chellappa, "A learning approach towards detection and tracking of lane markings," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1088–1098, Sep. 2012.

[2] O. Tournaire and N. Paparoditis, "A geometric stochastic approach based on marked point processes for road mark detection from high resolution aerial images," *ISPRS J. Photogramm. Remote Sens.*, vol. 64, no. 6, pp. 621–631, Nov. 2009.

[3] B. Yang, L. Fang, Q. Li, and J. Li, "Automated extraction of road markings from mobile LiDAR point clouds," *Photogramm. Eng. Remote Sens.*, vol. 78, no. 4, pp. 331–338, Apr. 2012.

[4] H. Guan *et al.*, "Using mobile laser scanning data for automated extraction of road markings," *ISPRS J. Photogramm. Remote Sens.*, vol. 87, pp. 93–107, Jan. 2014.

[5] M. Thuy and F. P. León, "Lane detection and tracking based on LiDAR data," *Metrol. Meas. Syst.*, vol. 17, no. 3, pp. 311–321, Dec. 2010.

[6] A. Mancini, E. Frontoni, and P. Zingaretti, "Automatic road object extraction from mobile mapping systems," in *Proc. IEEE/ASME Int. Conf. Mechatron. Embedded Syst. Appl.*, Suzhou, China, Jul. 2012, pp. 281–286.

[7] P. Lindner, E. Richter, G. Wanielik, K. Takagi, and A. Isogai, "Multichannel LiDAR processing for lane detection and estimation," in *Proc. Int. IEEE Conf. Intell. Transp. Syst.*, St. Louis, MO, USA, Oct. 2009, pp. 1–6.

[8] X. Chen *et al.*, "Next generation map making: Geo-referenced ground-level LiDAR point clouds for automatic retro-reflective road feature extraction," in *Proc. ACM SIGSPATIAL Int. Conf. Adv. Geogr. Inf. Syst.*, Seattle, WA, USA, Nov. 2009, pp. 488–491.

[9] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man Cybern.*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[10] 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.

[11] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[12] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[13] R. Salakhutdinov and G. Hinton, "An efficient learning procedure for deep Boltzmann machines," *Neural Comput.*, vol. 24, no. 8, pp. 1967–2006, Aug. 2012.

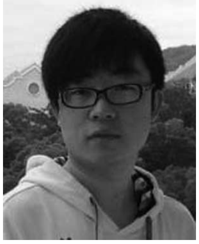
[14] R. Salakhutdinov, J. B. Tenenbaum, and A. Torralba, "Learning with hierarchical-deep models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1958–1971, Aug. 2013.

[15] T. Tieleman, "Training restricted Boltzmann machines using approximations to the likelihood gradient," in *Proc. 25th Int. Conf. Mach. Learn.*, Helsinki, Finland, Jul. 2008, pp. 1064–1071.

[16] H. Larochelle and Y. Bengio, "Classification using discriminative restricted Boltzmann machines," in *Proc. 25th Int. Conf. Mach. Learn.*, Helsinki, Finland, Jul. 2008, pp. 536–543.

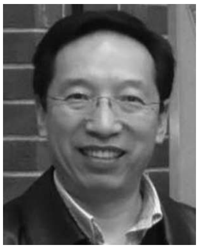
[17] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," in *Proc. 12th Int. Conf. Artif. Intell. Statist.*, Clearwater, FL, USA, Apr. 2009, vol. 12, pp. 448–455.

[18] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram," *Comput. Vision Graph. Image Process.*, vol. 29, no. 3, pp. 273–285, Mar. 1985.



Yongtao Yu received the B.S. degree in computer science and technology from Xiamen University, Xiamen, China, in 2010. He is currently pursuing the Ph.D. degree in computer science and technology at the Department of Computer Science, Xiamen University.

His research interests include computer vision, machine learning, mobile laser scanning, and information extraction from 3-D point clouds.



Jonathan Li (M'00–SM'11) received the Ph.D. degree in geomatics engineering from the University of Cape Town, Cape Town, South Africa, in 2000.

He is currently with the Key Laboratory of Underwater Acoustic Communication and Marine Information Technology (MOE) and the School of Information Science and Engineering, Xiamen University, Xiamen, China. He is also heading the Laboratory for Geo-spatial Technology and Remote Sensing, Faculty of Environment, University of Waterloo, Waterloo, ON, Canada, where he is a Professor and an Elected Member of the University Senate, Nashville, TN, USA. He has coauthored more than 200 publications, over 60 of which were published in refereed journals. His research interests include information extraction from earth observation images and 3-D surface reconstruction from mobile laser scanning point clouds.

Dr. Li is the Chair of the Inter-Commission Working Group I/Va on Mobile Scanning and Imaging Systems of the International Society for Photogrammetry and Remote Sensing (2012–2016), the Vice Chair of the Commission on Hydrography of the International Federation of Surveyors (2011–2014), and the Vice Chair of the Commission on Mapping from Remote Sensor Imagery of the International Cartographic Association (2011–2015).

Haiyan Guan received the Ph.D. degree in geomatics from the University of Waterloo, Waterloo, ON, Canada, in 2014.

She is currently a Senior Research Fellow with the Department of Geography and Environmental Management, University of Waterloo. She has coauthored more than 30 research papers published in refereed journals, books, and proceedings. Her research interests include airborne, terrestrial, and mobile laser scanning data processing algorithms and 3-D spatial modeling and reconstruction of critical infrastructure and landscape.

Fukai Jia received the B.S. degree in computer science and technology from Ocean University of China, Qingdao, China, in 2012. He is currently pursuing the M.S. degree in computer science and technology at the Department of Computer Science, Xiamen University, Xiamen, China.

His research interests include mobile laser scanning point cloud processing, and 3-D object detection and recognition from mobile laser scanning point clouds.

Cheng Wang (M'12) received the Ph.D. degree in information communication engineering from the National University of Defense Technology, Changsha, China, in 2002.

He is a Professor and Vice Dean of the School of Information Science and Engineering, Xiamen University, Xiamen, China. He has coauthored more than 80 papers. His research interests include remote sensing image processing, mobile laser scanning data analysis, and multisensor fusion.

Dr. Wang is the Co-Chair of the ISPRS WG I/3, council member of the Chinese Society of Image and Graphics (CSIG), and member of SPIE.