# Semi-Automated Generation of Road Transition Lines Using Mobile Laser Scanning Data

Chengming Ye, Jonathan Li, *Senior Member, IEEE*, Han Jiang, He Zhao, Lingfei Ma, and Michael Chapman

***Abstract*— This paper recognizes the research gaps and difficulties in generating transition lines (the paths that pass through a road intersection) in road intersections from mobile laser scanning (MLS) point clouds. The proposed method contains three modules: road surface detection, lane marking extraction, and transition line generation. First, the points covering the road surface are extracted using the voxel-based upward growing and the improved region growing. Then, lane markings are extracted and identified according to the multi-thresholding and the geometric filtering. Finally, transition lines are generated through a combination of the lane node structure generation algorithm and the cubic Catmull–Rom spline algorithm. The experimental results demonstrate that transition lines can be successfully generated for both T- and cross-intersections with promising accuracy. In the validation of lane marking extraction using the manually interpreted lane marking points, the method can achieve average precision, recall, and $F_1$-score of 90.80%, 92.07%, and 91.43%, respectively. The success rate of transition line generation is 96.5%. Furthermore, the buffer-overlay-statistics (BOS) method validates that the proposed method can generate lane centerlines and transition lines within 20-cm-level localization accuracy from the MLS point clouds.**

***Index Terms*— Mobile laser scanning (MLS), road transition line, high-definition road map (HDRM), unmanned aerial vehicle (UAV).**

## I. Introduction

THE development of Autonomous Vehicles (AVs) starts in the 1980s, when Carnegie Mellon University introduced the Navlab vehicles that operated autonomously in a variety of road environments [1]. The range of vehicular automation can be described by the Levels of Vehicle Automation, which is put forward by the U.S. Department of Transportation's National Highway Traffic Safety Administration. The agency separates vehicle automation into five levels: no automation (Level 0), function-specific automation (Level 1), combined function automation (Level 2), limited self-driving automation (Level 3), and full self-driving automation (Level 4) [2]. To help full self-driving automation, extra platforms including motion planning systems, perception systems, mission planners, and behavioral systems, should be set on board [3].

The platforms control the behaviors of an autonomous vehicle based on two kinds of data: real-time perception results from onboard sensors and pre-loaded navigation maps [4]. The onboard sensors, such as optical cameras, Radio Detection and Ranging (RADAR) and Light Detection and Ranging (LiDAR) systems, are essential parts to support the autonomous vehicles driving in road environments. However, urban scenes could be too complex to be completely detected by the sensors. Moreover, some critical road information might not be captured due to the range limits of the sensors. In such cases, road maps, which are used to generate routing trajectories for autonomous driving in well-structured environments, can support real-time motion planning if detailed road information is precomputed [5].

A classical road map for navigation is created from a Geographic Information System (GIS) through information filtering, organizing, and visualizing. The road information stored in a GIS database is mostly gathered by surveying and remote sensing techniques. Aerial photogrammetry, one remote sensing data source, is commonly used for road and road feature extraction. These methods work very well along the main road network in study areas. However, simultaneously, they can be seriously impacted by blocking features along the road, such as high-rise buildings and trees. Moreover, when taking a further step toward road surface information extraction or branch road extraction, the small disconnects in picturing road details caused by blocking objects and missing elevation information makes the reconstruction of roads difficult [6]. In summary, due to the high complexity and diversity of road environments, the generation of HDRMs from aerial images is problematic.

Compared with aerial images, laser scanning, as a continuously promoted remote sensing technique, can provide quick acquisition of high-quality Three-dimensional (3D) information of road scenes when integrated with positioning technology [7]. MLS platforms are useful for providing point clouds for roads within a city. Moreover, in terms of point density, MLS point clouds commonly have more than 10,000 pts/m$^2$, which provides greater than 1 cm in resolution, while it is

Manuscript received October 22, 2017; revised June 15, 2018; accepted February 16, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 41471379 and in part by the National Key Technologies Research and Development Program under Grant 2016YFB0502603. The Associate Editor for this paper was F. Nashashibi. *(Corresponding author: Jonathan Li.)*

C. Ye is with the Key Laboratory of Earth Exploration and Information Technology, Ministry of Education, Chengdu University of Technology, Chengdu 610059, China (e-mail: rsgis@sina.com).

J. Li, H. Jiang, H. Zhao, and L. Ma are with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: junli@uwaterloo.ca; h64jiang@uwaterloo.ca; h224zhao@uwaterloo.ca; l53ma@uwaterloo.ca).

M. Chapman is with the Department of Civil Engineering, Ryerson University, Toronto, ON M5B 2K3, Canada (e-mail: mchapman@ryerson.ca).

Digital Object Identifier 10.1109/TITS.2019.2904735

hard for Airborne Laser Scanning (ALS) point clouds to reach that precision [8]. As a result, road features such as lane markings are only distinguishable on MLS point clouds. Due to its higher flexibility and acquisition rate in large-scaled road scenes, MLS point clouds is more applicable than the other remote sensing techniques for transition line generation in this study.

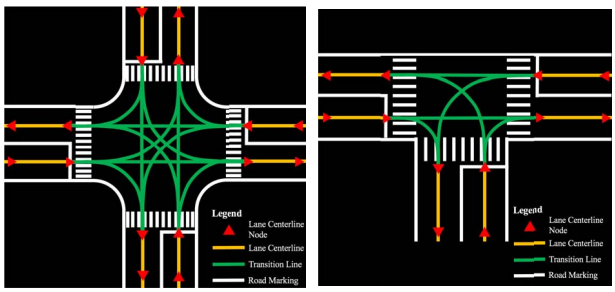
However, there are still some challenges in generating transition lines from MLS point clouds. The complex patterns of road markings in road intersections make the acquisition of road markings challenging. Outliers, such as road side building facades, road pavements, and other points on the road surface, should also be distinguished and removed [9]. Another challenge comes from conducting human knowledge (traffic rules) in lane centerline and transition line generation. In addition to the complex patterns of road environments, challenges also arise from MLS point clouds. Such point clouds contain a mass of dense points with 3D geographic information, reflection information, return information, and so on. Organizing these features would be a key pre-condition in developing efficient algorithms.

The transition line generation method proposed in this study mainly contributes to the construction of HDRMs using MLS point clouds. The detailed contributions are, namely, an improved curb-based region growing algorithm for road surface detection and a semi-automated node structure generation algorithm: (1) The curb-based region growing algorithm detects road surface points by using trajectory points as seeds and then iteratively searching nearby areas. This algorithm is feasible for different types of curbed urban roads, including straight roads, curved roads, and road intersections. Furthermore, it reduces accuracy loss by detecting road surface points directly from point clouds. (2) A semi-automated node structure generation algorithm is presented. The generated lane centerline nodes facilitate the construction of transition lines. By utilizing lane marking point clouds as input data, it does not require Global Navigation Satellite System (GNSS) trajectory covers all lanes, and that improves the efficiency of the data collection in road intersections. This study makes a considerable contribution to the research on generating transition lines for HDRMs, which further contributes to the research of AVs and Vehicle-to-Everything (V2X) technology.

## II. Background and Related Work

### A. Introduction to High-Definition Road Maps

HDRMs describe the road geometry as linked lane segments. They characterize roads, on the one hand, with more recall and, on the other hand, with higher accuracy than standard digital maps to fulfill the requirements of driving assistance systems [10]. Fig. 1 clarifies the definition of transition lines in a cross-intersection (Fig. 1 (a)) and a T-intersection (Fig. 1(b)). Three constraints are involved in the generation of transition lines for an HDRM database. First, each road is decomposed into a sequence of analytic lane centerline curves, which represent the centerlines of road lanes. Second, lane centerlines are directed line segments that have start and end nodes. Third, lane centerlines are connected




(a) (b)

Fig. 1. Definition of transition lines. (a) Cross-intersection. (b) T-intersection.

by transition lines that are represented as curved or straight line segments. The acceptable vehicle positioning errors for lane-level navigation are based on a total error that combines the expected error of the road map database with vehicle positioning error. For three-meter-wide lanes, the lane-level accuracy is specified to be 30 cm in the road map database, with less than 20 cm vehicle positioning error [11].

Transition lines in the HDRM database approximate real vehicle turning paths and describe the geometric details of a road intersection. They present all the possible paths that autonomous vehicles may take to pass the road intersection. Moreover, the connectivity description provides a coordinate description to the inner attributes of the road intersection. Therefore, the vehicle positioning results from onboard sensors can be matched and constrained more continuously and precisely [12]. Transition lines can support smooth switch among a road intersection and provide more benefits to map-based vehicle positioning solutions [13].

### B. Road Surface Detection

Several methods have been developed to detect road surfaces from MLS point clouds, and these algorithms can be categorized into three types based on data structure: scan line, road patch, and voxel.

Scan lines are commonly generated using the scanning angle field or the GNSS time stamp field in MLS point clouds if points are time-wise sorted. With a few points in a single scan line, computation can be conducted efficiently. In [14], road curbs (the boundaries of roads) were detected to extract road surface points from scan lines. Elevation jump and slope of two adjacent points were computed and labeled as road curb points if both were greater than the thresholds. In [15], road curbs were detected from a scan line through constructing a moving window operator. Instead of searching for road boundaries, some methods directly extract road surface points from a scan line. In [16], elevation jump between trajectory points and road surfaces was utilized to extract seed points, then road surface points were detected by moving least squares line fitting. Similarly, in [17], Principal Component Analysis (PCA) was performed to the local neighborhood of seed points for road surface detection. Although dividing a point cloud into scan lines makes complex computation and algorithm applicable, information contained in a scan line is still not enough to detect road curbs in complex road environments [18]. Furthermore,

when MLS point clouds are not sorted by the time stamp field, or they are mosaicked from several datasets, scan line based algorithms need extra time for sorting of point clouds.

Road patches can be generated by segmenting MLS point clouds into blocks with the assistance of trajectory data at a specified time or distance interval. Compared with scan lines, road patches not only have high computational efficiency but also contain more geometric information on road surfaces. Many road surface detection methods are based on road patches. In [7] and [19], road patches were transversely sectioned into corresponding profiles, and then the profiles were gridded to form a pseudo scan-line with a given width. Road curbs in each road patch were detected based on slope and elevation jump between two consecutive points. In [20] and [21], the saliency of points within each patch were measured, and then salient points were clustered to compute the characteristics of road scenes and detect road curbs. Although road patch based methods provide promising results in road surface detection, the accuracy is influenced by the trajectory data. In T-intersections, the curved trajectories might cause false segmentation of MLS point clouds.

Voxels are cuboid-shaped space with a given length, width, and height. Point clouds can be gridded using voxels. Some experts have tried to detect road surfaces and road curbs in voxels. In [22], the dynamic Digital Elevation Models (DEMs) were constructed with the average height of points in voxels to diminish the impact of moving vehicles around the scanners and to capture the complete geometric features of the road surfaces. Dynamic DEMs can present complete road curbs and road surfaces; however, they are built from 2D sequential laser range finder data and vehicle state data, which are not available in most MLS point cloud datasets. In [23], multi-category labeling was conducted based on the voxelization of MLS point clouds. Furthermore, in [24], density gradients of adjacent voxels were calculated to extract road curb voxels and removing noise. The method achieves promising results; however, it mainly focused on detecting road curbs. Road surface points should be further extracted with the detected curbs. Moreover, the computational time of applying this method could be lengthy when processing high-density and high-volume MLS point clouds in road intersections.

### C. Road Marking Detection

Extracting road marking points from MLS point clouds contains two typical steps: candidate road marking point extraction and post-refinement. Although road markings are highly reflective objects painted on road surfaces, the reflected laser pulse intensities also depend on the incident angle of the emitted laser beams and the range between measured road markings and laser scanners. Generally, the laser pulse intensity value decreases with the increase of range and incident angle [7]. The extraction methods can be classified into two categories: geo-referenced feature (GRF) image-based methods and point-based methods.

The generation of GRF images was mainly based on the Inverse Distance Weighting (IDW) interpolation [19]. To solve the uneven distribution and fluctuation of intensities, the multi-threshold segmentation method was conducted for road marking extraction using GRF images [19]. Moreover, in [25] and [26], estimated transverse range was used to divide GRF images into blocks and then determined an intensity threshold for each block based on its range from laser scanners. The accuracy of GRF image based methods can be over 90%; however, road markings were projected to a 2D plane, which inevitably led to some accuracy loss. The refinement of candidate road marking points in GRF images based methods is directly applied to the extracted pixels. False positive and false negative errors can be efficiently eliminated by the median filtering [27] or the multi-scale tensor voting [28]. To fill the incomplete road markings, labeled GRF images were converted to binary images in [25]. A linear shaped structuring operator was then used to enlarge the road marking areas and refine the extraction results.

Unlike GRF image based methods, point based road marking extraction methods directly analyze 3D road surface point clouds and detect candidate road marking points. In [16], road surface points were segmented into scan lines according to the scanning angle and detected road marking edge points through a dynamic window median filter. In [29], the multi-thresholding strategy was adopted to road surface point clouds to solve the uneven distribution of intensities and to extract road marking points. This method offers a promising approach in road marking extraction and it can achieve high accuracy. The refinement of candidate road marking points extracted by point based methods is also critical since small elements on the road surface might have a high reflective level and cause false extraction noise. False positive and false negative errors can be removed by dividing the candidate road marking points into scan lines [16]. In [29], the spatial density filter was developed to eliminate false positive and false negative errors.

### D. Lane Centerline and Transition Line Generation

In recent studies, lane centerline generation was widely conducted by analyzing GNSS trajectories. Lane centerlines were extracted using an analytical equation that best fits the trajectories, with the constant curve as circles and null curve as straight lines [30]. The method was promising if road pattern was simple. In [31], the node structure was generated to represent planar roadways for macro-scale navigation. However, GNSS trajectory based lane centerline generation methods required the survey to each lane and driving as close as possible to its center, which is not applicable for large urban areas.

Road intersections are defined as the places where two or more road sections across [12]. In road environments, there is no marked transition path from a lane to another in road intersections; however, it does not indicate that vehicles can drive freely in these areas. Therefore, generating a “virtual” line that connects one lane to another while maintaining the continuity between these two lanes is a great solution for HDRM construction. To generate transition lines, it was assumed that vehicles traveled at the road intersection with a constant steering angle in [12] and [13]. A lane centerline was firstly simplified as a line segment with a pair of ‘entry’

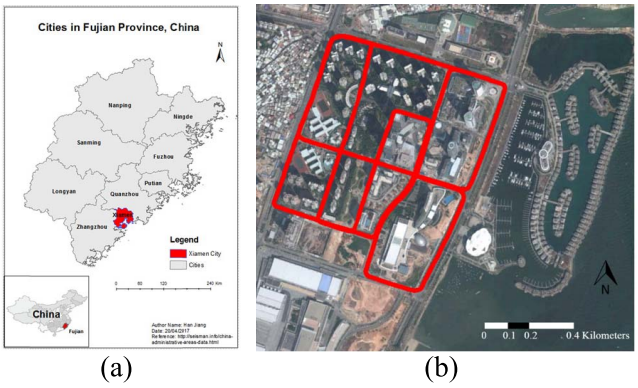



Fig. 2. Study area in Xiamen, Fujian, China. (a) Location of the study area. (b) Trajectory (red) overlaid on Google imagery.

and 'exit' node, and then direction symbols were calculated by the vectors of the start lane and the end lane. Finally, an arc section was calculated to represent the transition line. This method provided a solution to link 'exit' nodes and 'entry' nodes; however, it cannot preserve the continuity at joint points. A promising method for transition line generation was presented in [32]. They employed the cubic Catmull-Rom spline with five control points to generate line segments from an 'exit' node to an 'entry' node and to keep the continuity at joint points. The Central point of a road intersection, the 'exit' lane, and the 'entry' lane were utilized to determine the five control points. The success rate reached 90% in the experiment.

## III. Methodology for Transition Line Generation

### A. Study Area and Datasets

The study area is in Xiamen City, southeastern Fujian, China (longitude 24° 28'47.41"N, 118° 05' 21.91"E) (see Fig. 2. (a)). It is a tropical city that has a monsoonal humid subtropical climate. This kind of climate is characterized by mild and dry winter and hot and humid summer. The vegetation in Xiamen City is year-long green. Since there is less rainfall during the winter season and it is more suitable for data collection, the point clouds were collected on December 20, 2013. The surveyed area, called International Conference and Exhibition Center Block, is an urban area where traffic is free-flowing (see Fig. 2 (b)). As a result, most road markings and road curbs in this area are complete, and cracks are rare on road surfaces. Moreover, roads with two lanes (one lane in each direction) or four lanes (two lanes in each direction) are intersected in this area. T-intersections and cross-intersections are formed. In addition, the intersections are characterized by numerous roadside trees, vehicles, light poles, and traffic poles. Diverse types of road intersections with the various numbers of lanes makes this area ideal for testing the proposed method.

A total of 2.961 billion points with the size of 19.7 GB were collected and stored in 17 LAS files. The MLS point clouds were collected by a RIEGL VMX-450 MLS system. The gray-scale values of the MLS intensity were transformed to 0 to 255. About 0.78 billion trajectory points were generated and provided as ancillary data. Four test datasets,

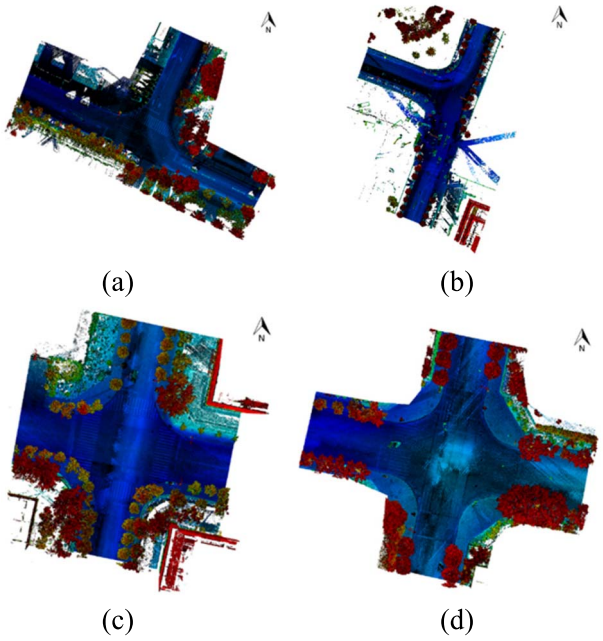



Fig. 3. Four test datasets. (a) Test Dataset 1. (b) Test Dataset 2. (c) Test Dataset 3. (d) Test Dataset 4.

including two T-intersections (Test Datasets 1 and 2) and two cross-intersections (Test Datasets 3 and 4) are selected from the original point clouds (see Fig. 3.). Test Dataset 1 is a T-intersection where two four-lane roads and one two-lane road intersected, and Test Dataset 2 is a T-intersection where three four-lane roads intersected. Test Datasets 3 and 4 are cross- intersections where four four-lane roads intersected. Since they cover all road intersection types and road marking types in the study area, these four datasets are selected. All the four road intersections are characterized by detectable road curbs, standard road markings, and complex roadside objects.

### B. Overview of Workflow

The proposed methodology contains three modules: road surface detection, lane marking extraction, and transition line generation. Fig. 4 presents the workflow of the proposed methodology. In Module I, the voxel-based upward-growing algorithm [7] is employed for ground point detection from input point clouds. The region growing algorithm is improved to enhance curb-based road surface detection. In Module II, the multi-thresholding algorithm [29] is used for road marking extraction. Geometric feature filtering [33] is implemented to extract lane markings. In Module III, a node structure generation algorithm is proposed to generate lane geometries and lane centerlines from lane markings. The cubic Catmull-Rom spline [34] is employed for transition line generation.

### C. Module I: Road Surface Detection

The voxel-based upward-growing algorithm [7] is employed to detect ground points. Since MLS point clouds have large data volumes and high point density, the removal of non-ground (such as traffic signs, traffic lights, light poles, and tree crowns) can improve the computational efficiency and

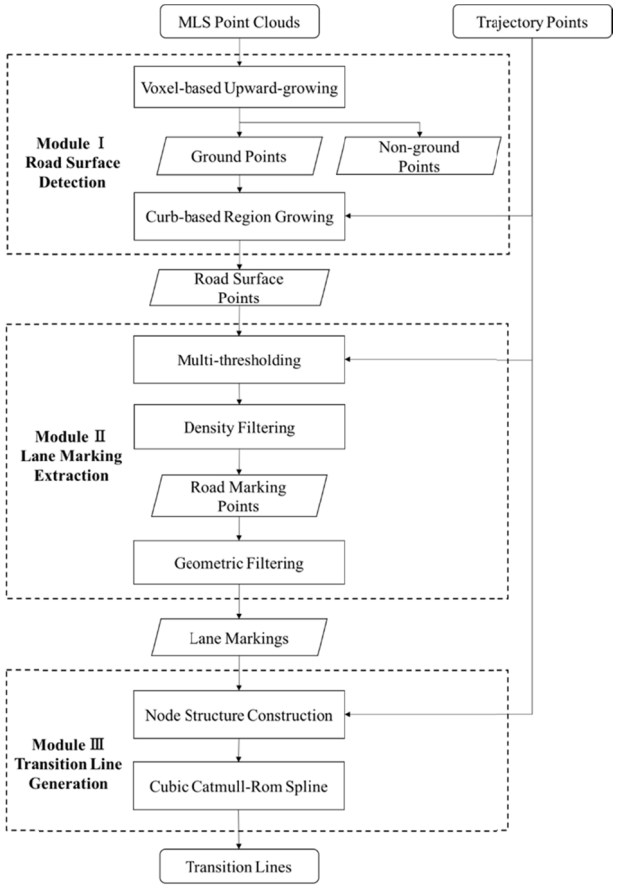



Fig. 4. Workflow of the proposed method.

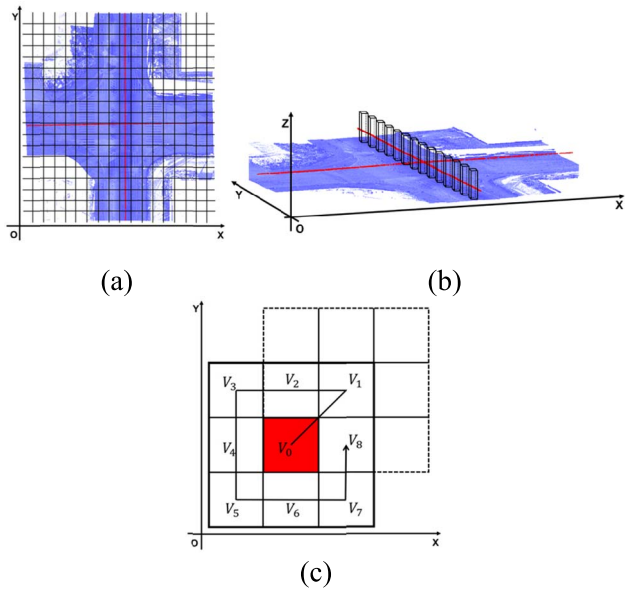



Fig. 5. Principle of the region growing. (a) Voxelization of ground points. (b) Seed voxels in road surface detection. (c) Breadth-first searching.

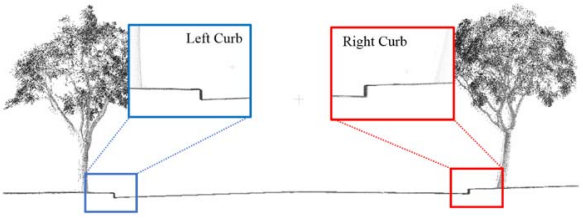



Fig. 6. Close-up illustration of road curbs.

accuracy of the following algorithms. Point clouds are firstly partitioned into point blocks to alleviate the ground

undulation in each block and to decrease the processing time. Points in each block are analyzed and processed separately in the following steps. Then, points in a block is voxelized with cube-shaped voxels. For each voxel, the upward-growing process grows along with its adjacent voxels recursively to form a voxel segment. Finally, a local ground undulation threshold and a global ground undulation threshold are used to label the non-ground voxels (voxels that contain non-ground points) in the voxel segment. By filtering out the non-ground voxels, ground points are detected.

The region growing algorithm is improved to enhance road surface detection, mainly in two ways. Trajectory points are important indicators of the road surface in seed selection because they mostly exist above road surfaces. Nevertheless, trajectory points cannot be directly used as seed points, since they all have a higher elevation than nearby road surface points. Thus, the processing unit should contain trajectory points and ground points simultaneously. To achieve this, the shape of voxels is modified. As shown in Figs. 5 (a) and (b), the ground points (blue) and the trajectory points (red) are gridded into cuboid-shaped voxels, which have a height equal to the point cloud space. Constructing these voxels generates the link between road surface points and the trajectory points. Those voxels that contain at least one trajectory point and one road surface point are selected as the original seed voxels.

The Breath-first Searching (BFS) algorithm [35] is adopted in the voxel growth step. As shown in Fig. 5. (c), a seed voxel $V_0$ has eight adjacent voxels, named $V_1, V_2, \ldots, V_8$. The BFS algorithm exhaustedly searches $V_1$ to $V_8$ in sequence. If any of these adjacent voxels does not contain a road curb segment, it is pushed into a queue. Once $V_1$ to $V_8$ are all searched and labeled, the BFS algorithm picks up the next voxel in the queue as a new seed voxel and repeats the same searching pattern.

Fig. 6 shows the close-up views of the road curbs and demonstrates that the road curbs are vertical or mostly vertical to road surfaces and have obvious elevation jumps. Therefore, road curbs can be detected through elevation jump and slope thresholds. Road curb voxels (voxels that contain road curb points) are estimated based on the following two observations.

*1) Elevation Jump:* If part of a road curb is covered by a voxel, the elevation jump in the voxel should meet the following observation [28]:

$$T_{curb\_min} \leq E_c \leq T_{curb_max}, \tag{1}$$

where $E_c$ denotes the elevation jump in the voxel, and $T_{curb\_min}$ and $T_{curb\_max}$ is the threshold of the elevation jump. According to [28], street design and construction manuals in many countries document that the height of road curbs

generally ranges from 10 cm to 25 cm, and $T_{curb\_min}$ and $T_{curb\_max}$ is predefined as 8 cm and 30 cm, respectively.

*2) Slope:* The slope observation is defined as following [28]:

$$\arctan\left[\frac{(z_{i+1}-z_i)}{\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}}\right] \geq T_{slope}, \quad (2)$$

where $(x_i, y_i, z_i)$ and $(x_{i+1}, y_{i+1}, z_{i+1})$ are the coordinates of two adjacent points in a voxel, and $T_{slope}$ is the slope threshold. According to [28], the $T_{slope}$ is predefined as $2\pi/3$.

The growth stops when the following conditions occur: first, there is no seed voxel in the queue; second, the distance from a searched voxel to the original seed voxel is greater than the pre-defined threshold. The first condition indicates that the searching is completed. As an additional stop condition, a distance is set to restrain the searching radius of the BFS algorithm. Because of the ground's undulation, if the growth from an original seed with higher elevation is not limited, road curbs located in lower positions might not be detected.

### *D. Module II: Lane Marking Extraction*

Lane marking refers to road markings that indicate lane edges. Typical lane markings include single lines, double lines, and dashed lines. Since lane markings are critical information for lane centerline generation, they are extracted from the road surface points using four adapted algorithms. Generally, road markings are painted by highly reflective material on the road surfaces. Road marking points have higher intensities than surrounding road surface points. Thus, the multi-thresholding [29] and the Otsu's thresholding [36] are employed for candidate road marking point extraction. The candidate road marking points might contain false extraction noise. The noise is caused by small objects that have similar intensities as road markings on road surfaces. To eliminate the noise, the spatial density filtering is conducted [29]. Furthermore, the extracted road marking points are isolated and have no topological relationships. In other words, they have no semantic meanings. To extract lane marking points from road marking points, clustering should be applied. By assuming that nearby points belong to the same road marking, the Conditional Euclidean clustering is employed [7].

When road marking points are clustered, geometric features of point segments can be calculated and utilized for lane marking extraction. Generally, there are five types of road marking in the test datasets: solid line, dashed line, zebra crossing line, Chinese character, and turning arrow. The road marking painting standards, which are documented in [37]. A minimum bounding rectangle is generated to extract the shape features of each road marking cluster. According to the length $L$ and width $W$ of the minimum bounding rectangles, lane markings are extracted using the geometric parameters listed in Table I.

### *E. Module III: Transition Line Generation*

In this module, lane node structure, which represents lane geometries in road intersections, is firstly constructed by a combination of lane marking node structure generation and 'exit' and 'entrance' node pairing. Moreover, the cubic Catmull-Rom spline is employed to generate transition lines for paired nodes according to [32] and [34].

The lane node structure construction algorithm utilizes lane marking points and trajectory points as input data. The trajectory points are resampled at an interval $I_T$ to generate a point set $T_j (j = 0, 1, \ldots, L)$. As shown in Fig. 7 (a), $T$ is used to partition the lane marking point cloud into blocks $(Block_i (i = 0, 1, \ldots, N))$. The length of blocks is restricted by a threshold $T_B$. For each block, a Cartesian coordinate system is confirmed, with an X axis towards the front of the vehicle, a Y axis towards the left of the vehicle, and a Z axis towards the top of the vehicle. The coordinate origin is set at the midpoint of the line segment that connects $T_j$ and $T_{j+1}$.

Fig. 7 (b) presents the processing of a block $Block_i$. Firstly, lane marking points (black) in $Block_i$ are projected onto the YOZ plane of the coordinate. Then, by exhaustedly searching the projected lane marking points, central points $MP_k$ $(1 \leq k \leq 4)$ (blue) of lane markings are computed. If the width of a lane marking segment is greater than a threshold $T_M$, it is labeled as a stop line (the white line where vehicles stop for traffic). Furthermore, candidate lane centerline nodes (red) are generated by calculating the midpoint of $MP_k$ and $MP_{k+1}$. If the range between $MP_k$ and $MP_{k+1}$ is less than a threshold $R_T$, the candidate lane centerline node calculated from them is removed. In addition, by assuming that lanes are straight near road intersections, the least square fitting is employed to generate lane centerlines from generated nodes. If a lane centerline has an intersection with a stop line, it is labeled as 'exit' lane, and its vertex is labeled as 'exit' node. Otherwise, the lane is labeled as 'entry' lane, and its vertex is labeled as 'entry' node.

TABLE I
PARAMETERS FOR LANE MARKING EXTRACTION

| Road marking type | Range of $L$ and $W$ |
|---|---|
| Dashed lines | $150\ cm \leq L \leq 250\ cm$ & $5\ cm \leq W \leq 20\ cm$ |
| Single lane lines | $400\ cm \leq L \leq 2500\ cm$ & $5\ cm \leq W \leq 20\ cm$ |
| Lane lines connected by a stop line | $600\ cm \leq L \leq 2500\ cm$ & $W \geq 500\ cm$ |
| Outliers | Otherwise |

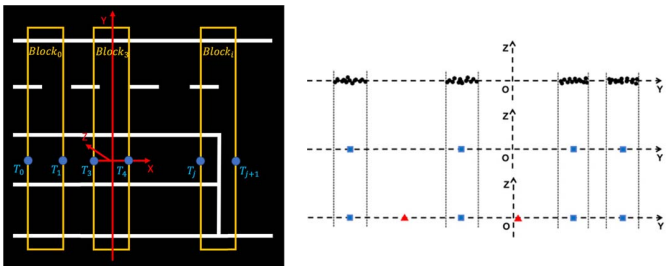



Fig. 7. Lane node structure construction. (a) Lane marking partitioning. (b) Node structure generation.

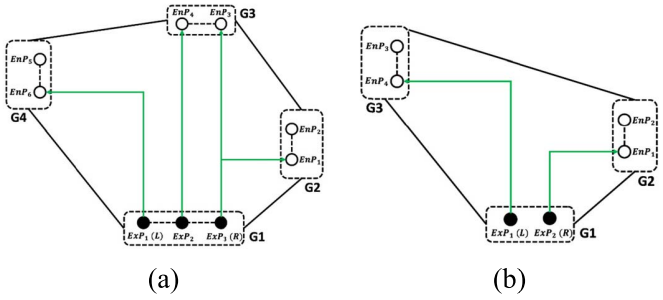



Fig. 8. Pairing 'entry' and 'exit' nodes. (a) Pairing 'entry' and 'exit' nodes in a cross-intersection. (b) Pairing 'entry' and 'exit' nodes in a T-intersection.

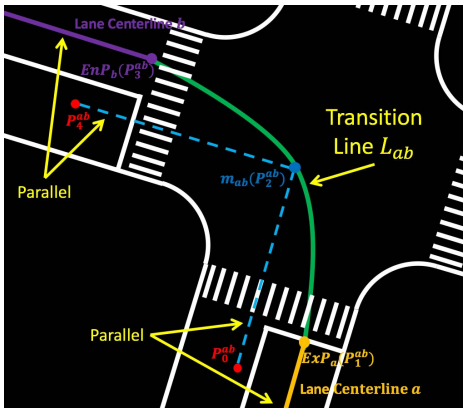



Fig. 9. Calculating a transition line.

When the node structure of a road intersection is generated, all proper 'entry' nodes for an 'exit' node are then determined. As shown in Fig. 8. (a), 'entry' nodes and 'exit' nodes on four branch roads in a cross-intersection are grouped as $G_1, G_2G_3$, and $G_4$. The convex hull (black lines) of the four groups is firstly generated by the Gift Wrapping algorithm to build topological relationships for them [38], [39]. Then, the algorithm searches the 'exit' nodes ($ExP_i(1 \leq i \leq 3)$) in $G_1$, and all the 'entry' nodes ($EnP_i(1 \leq i \leq 6)$) in $G_2$, $G_3$ and $G_4$. Corresponding 'entry' nodes $EnP_3$ and $EnP_4$ for the 'exit' node $ExP_3$ and $ExP_2$, respectively, are in the non-adjacent node group $G_2$. Additional connections are generated from the rightmost 'exit' node $ExP_3$ to the 'entry' node $EnP_1$ and from the leftmost 'exit' node $ExP_1$ to the 'entry' node $EnP_6$. The algorithm processes the 'exit' nodes in $G_2$, $G_3$ and $G_4$ in the same pattern.

In T-intersections, two branch roads that have the same direction follow the same pairing rules as branch roads in cross-intersections. However, pairing rules for the other branch road should be additionally discussed, because vehicles cannot go straight ahead on this branch road. As shown in Fig. 8 (b), 'exit' nodes of the branch road are labeled as solid points $ExP_j(1 \leq j \leq 2)$, and 'entry' nodes are labeled as points $EnP_j(1 \leq j \leq 4)$ s. Links are constructed

from the rightmost 'exit' $ExP_2$ to 'entry' node $EnP_1$ and from the leftmost 'exit' $ExP_1$ to 'entry' vertex $EnP_4$. Through the Gift Wrapping algorithm, proper 'entry' lanes for a 'exit' lane can be determined at T- and cross-intersections.

TABLE II
ACCURACY OF ROAD MARKING EXTRACTION WHEN APPLYING DIFFERENT SEARCHING RANGES

| Searching Range (m) | 5 | 10 | 15 | 20 | 25 | 30 | 35 |
|---|---|---|---|---|---|---|---|
| $F_1$-score (%) | 86.20 | 90.83 | 93.37 | 93.99 | 94.46 | 94.34 | 94.23 |
| Recall (%) | 82.15 | 89.64 | 94.83 | 95.88 | 96.60 | 96.55 | 96.59 |
| Precision (%) | 90.68 | 92.05 | 91.96 | 92.17 | 92.42 | 92.23 | 91.98 |

To generate a safe path from an 'exit' node to an 'entry' node and to keep the continuity at the nodes, the cubic Catmull-Rom spline is employed [34]. Five control points $P_k^{ab}(0 \leq k \leq 4)$ on a transition line $L_{ab}$ (green curve line), which connects lane centerline $a$ (yellow line) and $b$ (purple line), are determined according to [32] (see Fig. 9). The 'exit' node $ExP_a$ (yellow dot) of $a$ is set as $P_1^{ab}$, and the 'entry' node $EnP_b$ (purple dot) of $b$ is set as $P_3^{ab}$. An intermediate point $m_{ab}$ (blue dot) is set as $P_2^{ab}$. $P_0^{ab}$ is set to a location that makes line $P_o^{ab}P_2^{ab}$ (blue dashed line) parallel to $a$ to ensure the continuity at $P_1^{ab}$, because the tangent vector at a control point on a cubic Catmull-Rom spline is determined by the adjacent control points on two sides of it. The final control point $P_4^{ab}$ is set with the same pattern according to $b$. When five control points $P_k^{ab}(0 \leq k \leq 4)$ are determined, for $\leq \ddot{U}t \leq \ddot{U}1$, the cubic Catmull-Rom spline curves can be defined.

## IV. RESULTS AND DISCUSSION

### A. Results of Transition Line Generation

Ground points were detected through the voxel-based upward-growing algorithm. Figs. 10 (a) to (d) presents the ground point detection results. Road surface points were detected by the improved region growing algorithm. According to [27],$T_{curb\_min} = 8\ cm$, $T_{curb\_max} = 30\ cm$, and $T_{slope} = 2\pi/3$ were set. To determine the optimal parameters for $W_R$ and $T_B$, and to evaluate the sensitivity of each of them, two experiments were carried out. The purpose of detecting road surface points is to improve the accuracy of the road marking extraction. Thus, road markings extracted by different parameters were evaluated to determine the optimal parameters. Test Dataset 1 was used for the experiments, since it has different and exhaustive features.

Table II presents the $F_1$-score, recall, and precision of road marking extraction results using different searching ranges, with recall and precision recorded as well. A fixed voxel size $W_R = 10$ cm was used to explore the influences of searching ranges. Then, the road surface detection was carried out with different searching ranges, which were from 5 m to 35 m at an interval of 5 m. Generally, the $F_1$-score exponentially improved as the searching range increased from 5 m (86.20%) to 25 m (94.46%), and then slightly dropped from 25 m (94.46%) to 35 m (94.23%). The improvement of the $F_1$-score from 5 m to 25 m was caused by the increase of recall. Since the BFS searching started from the trajectory, the road surface was not completely detected when the searching range was

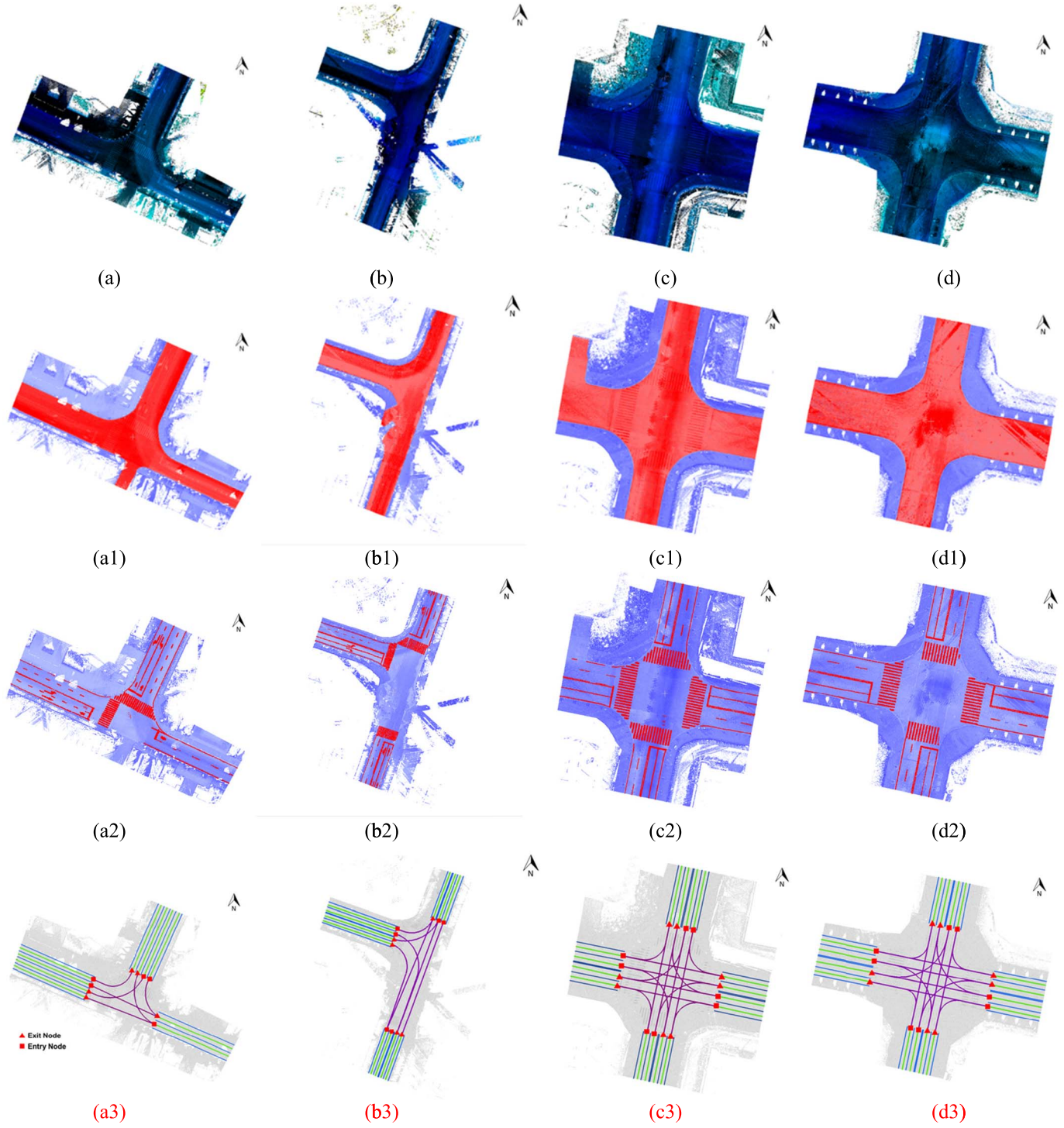



Fig. 10. Transition line generation results from four test datasets. (a) to (d) are detected ground points. (a1) to (d1) are detected road surface points. (a2) to (d2) are extracted road marking points. (a3) to (d3) are generated transition lines.

small, and road markings close to the boundaries of the road surface were not extracted. However, when the searching range was overlarge, outliers on the pavements might be involved in road surface detection results. Accordingly, searching range $T_B$ was set as 25 m for the BFS searching in road surface extraction.

Table III presents the $F_1$-score of road marking extraction results using different voxel sizes, with recall and precision recorded simultaneously. The searching range $T_B = 25$ m was fixed to explore the influences of the voxel size, which was from 6 cm to 16 cm at an interval of 2 cm. Generally, the $F_1$-score slightly increased when the voxel size increased from 6 cm (93.18%) to 10 cm (94.46%), and then significantly decreased when the voxel size was larger than 10 cm. When the voxel size was overlarge, road markings close to the boundary of the road surface were involved in the same voxel

TABLE III
ACCURACY OF ROAD MARKING EXTRACTION WHEN APPLYING DIFFERENT VOXEL SIZES

| Voxel Size (cm) | 6 | 8 | 10 | 12 | 14 | 16 |
|---|---|---|---|---|---|---|
| F1-score (%) | 93.18 | 93.98 | 94.46 | 93.29 | 91.59 | 89.66 |
| Recall (%) | 96.89 | 97.01 | 96.60 | 93.52 | 90.33 | 86.62 |
| Precision (%) | 89.74 | 91.13 | 92.42 | 93.06 | 92.89 | 92.93 |

as the road curb and removed, which caused a significant loss in recall. However, more points were covered with larger voxels, which made the calculation of elevation jump and slope more accurate. Hence, $W_R = 10$ cm was optimal for the road surface detection in this study.

With the defined parameters, the detected road surface points in Test Datasets 1 to 4 are presented in Fig. 10 (a1) to (d1). The overall performance of the improved algorithm was promising in the four test datasets. One false detection occurred in Test Datasets 1 and 2 (Fig. 10. (a1) and (b1)), as part of the road pavement was labeled as road surface. The road surface detection method was sensitive to the condition of road curbs; thus, the defects of road curbs might cause a large area of false detection. Three features of the algorithm can be concluded from the results. First, all road markings can be detected on the extracted road surfaces. Second, the boundaries of road surfaces are not smoothed due to the cubic shape of voxels. Third, small holes might exist due to noise on road surfaces.

Road marking points were extracted from road surface points, using the multi-thresholding algorithm. Road marking extraction results are shown in Fig. 10. (a2) to (d2). Generally, road markings are painted using a material that has high reflectance to laser pulses, and road marking points have higher intensities than the surrounding road surface points. However, due to the decay of road markings and obstacles on roads, some road marking points might have low intensities, and some road markings could be broken. In such cases, errors might occur in road marking extraction. Fig. 11 presents a critical error that occurred in the results for road marking extraction in Test Dataset 2. Road features are indicated by variations in color intensity. The road marking close to the road boundary was covered by a material that had low reflectance to laser pulses, or it might be completely decayed. Since its intensity was lower than that of the road surface, the road marking was not extracted from the road surface. Such defects of the road marking could lead to a failure in lane structure generation, thus causing errors in transition line generation.

Lane node structures were generated by the proposed method. The resampling interval $I_T$ was determined by the corresponding rate of the navigation system used in the MLS system and the driving speed. In this study, the corresponding rate of the GNSS was 0.01s, and the maximum driving speed was 40 km/h. Thus, the interval of trajectory points was about 8.3 cm. Note that the over-segmentation of the lane markings might aggravate the influence of noise; the resampling interval

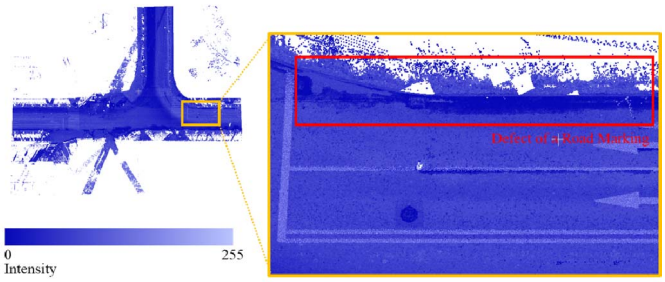



Fig. 11. Defect of the road marking in Test Dataset 2.

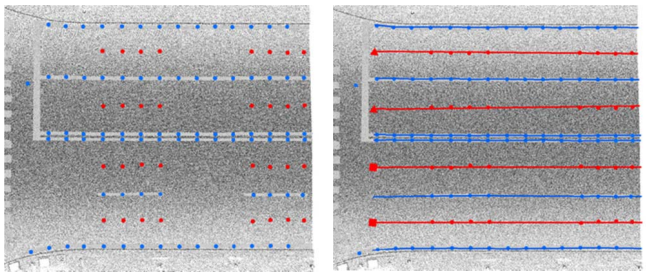


Fig. 12. Lane node structure generation results in a road segment.

$I_T = 6$ was defined for the test datasets. The width of a point block was about 50 cm. Additionally, the following three thresholds were used in this algorithm: $T_M$, threshold to detect stop lines; $T_{Rmin} and T_{Rmax}$, thresholds for node filtering. The thresholds were defined according to the minimum lane width in the study area, which was 3 m. When points in a data block were projected onto the transformed coordinate system, if the length of a clustered road marking was greater than $T_M = 2m$, the marking was labeled as a stop line. To eliminate false lane centerline nodes, $T_{Rmin}$ was set as 2 m and $T_{Rmax}$ was set as 4 m. Fig. 12 presents lane node structures in a road segment, with lane marking nodes in blue and lane centerline nodes in red.

Transition lines were generated by the cubic Catmull-Rom spline algorithm. As shown in Figs. 10 (a3) to (d3), test datasets are overlapped by the generated lane centerlines (green), lane markings (blue), and transition lines (purple). 'Entry' nodes of lane centerlines are square dots, and 'exit' nodes are triangle dots. In the four test datasets, most corresponding 'entry' and 'exit' nodes were correctly connected by transition lines, except that two transition lines connected wrong pairs of 'exit' and 'entry' nodes in Test Dataset 2 (see Fig. 10 (b3)). Due to the heavy defects of the leftmost lane marking on the top branch of the T intersection in Test Dataset 2, only three of four lanes on that branch were extracted, and the leftmost lane was missing. As a result, the second left lane on the top branch became the leftmost lane, and then, it was wrongly connected to the leftmost lane on the bottom branch. Simultaneously, the second left lane on the bottom branch had no 'entry' lane. Thus, the success rate of transition line generation was about 96.5%, again suggesting that road marking defects had negative effects on transition line generation.

TABLE IV
COMPUTING TIME OF THE PROPOSED METHOD

| Dataset | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Size | 565MB | 521MB | 905MB | 623MB |
| No. of MLS Points | 19,243,445 | 25,537,193 | 17,444,202 | 16,094,181 |
| No. of Trajectory Points | 13748 | 25982 | 12946 | 16973 |
| Time for Ground Point Detection (s) | 13.1s | 21.5s | 12.8s | 12.3s |
| Time for Road Surface Point Detection (s) | 29.3s | 35.3s | 50.4s | 53.9s |
| Time for Road Marking Extraction and Refinement (s) | 309.2s | 347.1s | 456.9s | 424.2s |
| Time for Lane Marking Extraction (s) | 105.5s | 126.5s | 185.7s | 193.2s |
| Time for Transition Line Generation (s) | 15.5s | 17.1s | 30.6s | 30.1s |
| Total Time (s) | 472.6s | 547.5s | 736.4s | 713.7s |

### B. Overall Performance Assessment

HDRM generation is a post-processing process. To speed it up, automation of the process is critical. Therefore, the computing time of the proposed method in four test datasets is shown in Table IV. The consuming time was calculated in each procedure and in total. The computer used in this study had an AMD FX-8350 4.2 MHz CPU, and 16 GB RAM. The proposed method performed very well considering the large-volume point cloud data. The voxel-based upward-growing and the improved region growing algorithms efficiently removed outliers from the test datasets. Road marking extraction and refinement procedure contributed most of the processing time because the multi-thresholding algorithm exhaustively processed every single point of the input data. By taking advantage of the Conditional Euclidean clustering algorithm, the lane marking extraction procedure did not consume much time. Furthermore, the lane marking node structure generation algorithm and the cubic Catmull-Rom spline algorithm were both efficient. Therefore, the overall efficiency of the proposed method could be further improved by modifying the multi-thresholding algorithm and the density filtering algorithm.

### C. Accuracy Assessment of Road Surface

The performance of the road surface extraction method is assessed by thematic accuracy, which refers to the differences between the labeled attributes of remotely sensed features and the true attributes of reference features [40]. The thematic accuracy is measured by three criteria: recall, precision, and $F_1$-score [41], [42]. The recall indicates the integrity of the extracted road surfaces. The precision denotes the percentage of valid road surfaces. The global score is evaluated by $F_1$-score. The indicators are expressed as follows:

$$Recall = TP/(TP + FN) \tag{3}$$

$$Precision = TP/(TP + FP), \text{ and} \tag{4}$$

TABLE V
ACCURACY ASSESSMENT OF ROAD SURFACES USING MANUALLY INTERPRETED POINT CLOUDS

| Test Dataset | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Precision (%) | 91.53 | 91.57 | 88.02 | 93.86 |
| Recall (%) | 96.28 | 92.73 | 95.06 | 97.59 |
| $F_1$-score (%) | 93.84 | 92.15 | 91.40 | 95.69 |

TABLE VI
ACCURACY ASSESSMENT OF ROAD MARKINGS USING MANUALLY INTERPRETED POINT CLOUDS

| Test Dataset | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Precision (%) | 92.42 | 88.67 | 91.71 | 90.38 |
| Recall (%) | 96.60 | 83.54 | 96.28 | 91.84 |
| $F_1$-score (%) | 94.46 | 86.23 | 93.94 | 91.10 |

$$F1 - score = 2 \times precision \times recall / (recall + precision), \tag{5}$$

where TP represents true positive, $FP$ indicates false positive, and $FN$ indicates false negative. In this part, $TP$ is the number of road surface points which are correctly classified, whereas $FP$ is the number of outliers that are falsely classified as road surface points. $FN$ refers to the number of road surface points which are misclassified as outliers.

By using manually interpreted road surface points from original MLS datasets as reference points, the quantitative assessments were conducted based on the recall, precision and F1-score. As illustrated in Table V, the proposed road surface extraction algorithm is capable of obtaining 95.42% in recall, 91.25% in precision and 93.27% in F1-score, respectively. The value of precision is less than that of recall for each sample, which demonstrates that certain outliers are misclassified as road surface points. Additionally, the number of manually labeled reference points is less than that of extracted points due to the defects of road curbs. Thus, the overall performance of the proposed road surface extraction algorithm is underestimated in the final results.

### D. Accuracy Assessment of Road Markings

The quantitative assessment of road marking extraction was also conducted based on the recall, precision and F1-score. In this part, $TP$ is the number of road marking points which are correctly classified, whereas $FP$ is the number of outliers that are falsely classified as road marking points. $FN$ refers to the number of road marking points which are misclassified as outliers.

The results of road marking accuracy assessment using manually interpreted point clouds are listed in Table VI. The average precision of the test datasets was 90.80%. Generally, the loss in precision has two causes. On the one hand, false clustering might occur when zebra crossing lines and lane lines are connected. On the other hand, the density filter cannot remove all noise from candidate road marking points. The average recall of the four test datasets reached 92.07%.

TABLE VII
COMPARISON WITH OTHER ROAD MARKING EXTRACTION METHODS

| Dataset | 1 | | | | 3 | | | |
|---|---|---|---|---|---|---|---|---|
| Method | Chen et al. [44] | Zhang [43] | Yu et al. [29] | Proposed Method | Chen et al. [44] | Zhang [43] | Yu et al. [29] | Proposed Method |
| Recall (%) | 68.90 | 91.38 | 80.71 | 92.42 | 77.45 | 90.57 | 91.66 | 91.71 |
| Precision (%) | 92.11 | 82.95 | 83.25 | 96.60 | 90.76 | 87.53 | 94.39 | 96.28 |
| $F_1$-score (%) | 78.83 | 86.96 | 81.96 | 94.46 | 83.58 | 89.02 | 93.00 | 93.94 |

The loss in recall might be caused by the removal of small road marking segments when conducting the geometric feature filtering. The $F_1$-score of the four test datasets were all over 86%, which implied that the road marking extraction method can extract most of the road marking points in T- and cross-intersections. The proposed method performed well in Test Datasets 1, 3, and 4, whereas Test Datasets 2 contributed more to the loss of accuracy due to the broken road markings. Thus, the method should be improved in the future, to extract complete road markings in complex road intersection scenes.

### *E. Comparative Study of Road Marking Extraction*

Since road marking information was essential for transition line generation, a comparative study was conducted to compare the road marking extraction method used in this study with three other methods: Yu *et al.* [29], Zhang [43], and Chen *et al.* [44]. The methods were tested in Test Dataset 1 and Test Dataset 3. The performance of the methods was evaluated with precision, recall, and $F_1$-score (see Table VII).

Chen's method focused only on the lane markings along the direction of the vehicle trajectory. Thus, road markings, including turning arrows, stop lines, and Chinese characters, were not extracted, and that caused a signification loss of recall. Zhang's method detected most of the road markings in the two test datasets. However, road pavements that were smooth and had large areas were not eliminated by the high-pass filtering. As a result, road pavement points that have high intensities were misclassified as road markings. In addition, Zhang transformed MLS point clouds to geo-referenced images and detected road marking pixels on the images. Labeling road marking points through the images also caused a loss of precision. Yu's method had low $F_1$-score in Test Dataset 1, since its road surface detection method and road marking extraction method are both based on the straight trajectory. When the method was applied in Test Dataset 1, road pavements that were close to the central part of the intersection were misclassified as road surface, and the road surface was not segmented correctly. That caused a loss of $F_1$-score in the result for Test Dataset 1.

The proposed method successfully extracted most of the road marking points in Test Dataset 1 and 3.Unlike Chen's method, it had the ability to extract all types of road markings on road surfaces. Unlike Zhang's geo-referenced feature image based method, it directly extracted road marking points from MLS point clouds and did not cause accuracy losses due to data transformation. Moreover, it performed better in road surface detection; thus, noises caused by road pavement points were rare. Compared with Yu's method, the proposed method made an over 10% improvement in $F_1$-score when trajectories were curved in Test Dataset 1. However, some errors still occurred in the results. The errors in the extracted road markings were mainly caused by road marking decay and constant road marking extraction thresholds. Although road marking extraction thresholds were separately determined in each data block, intensity deduction still happened in the data blocks. As a result, road marking points that had low intensities due to the road marking decay might not be extracted. Simultaneously, road surface points that had high intensities could be falsely extracted as road marking points. In addition, noises that were close to the road markings were not eliminated by the density filtering, thus reducing precision.

In conclusion, the comparative study of the existing road marking extraction methods indicates that the proposed method can achieve a better performance than Chen's Zhang's, and Yu's methods in road intersection environments. Moreover, the proposed method does not require a certain distribution of trajectory points, nor transformation of MLS point clouds.

### *F. Accuracy Assessment of Transition Lines*

The generated lane centerlines and transition lines are assessed by the Buffer-overlay-statistics (BOS) method [45]. The method firstly generates buffers in various sizes around test lines and reference lines and then compares them through overlaying and statistics. To present a quantitative accuracy assessment of lines features, buffering and overlaying is conducted iteratively. For a number ($n$) of buffer sizes $B_i, (1 \leq i \leq n)$, conduct the following four steps. Firstly, perform buffering on the line features in the test dataset $X$ and the reference dataset $R$ with $B_i$, and call the generated buffer polygons $XB_i$ and $RB_i$, respectively. Secondly, overlay $X$ with $RB_i$ to generate a mixed dataset $XRB_i$. Overlay $R$ with $XB_i$ to generate a mixed dataset $RXB_i$. Thirdly, calculate the sum length of the line segments from $X$ outside $RB_i$ in $XRB_i$ and the total length of line features in $X$. The recall $C_i$ of $X$ is computed by [45]:

$$C_i = \frac{Length(RinsideXB_i)}{Length(R)}. \tag{6}$$

Finally, calculate the sum length of the line segments from $R$ inside $XB_i$ in $RXB_i$ and the total length of line features in $R$. The miscoding $M_i$ of $X$ is computed by [45]:

$$M_i = \frac{Length(XoutsideRB_i)}{Length(X)}. \tag{7}$$

GRF images that were generated from raw point clouds were utilized as reference data. The gray values of pixels on the GRF images were interpolated from the intensities of raw point clouds using the IDW interpolation method [19]. The resolution of the GRF images was set to be 2 cm. A detailed description of GRF image generation can be found in [19].

TABLE VIII
RECALL AND MISCODING OF TEST DATASETS 3 AND 4

| | Test Dataset 3 | | Test Dataset 4 | |
|---|---|---|---|---|
| Buffer Zoning (cm) | 15 | 20 | 15 | 20 |
| Recall (%) | 91.83 | 100 | 90.98 | 100 |
| Miscoding (%) | 7.01 | 0 | 7.82 | 0 |

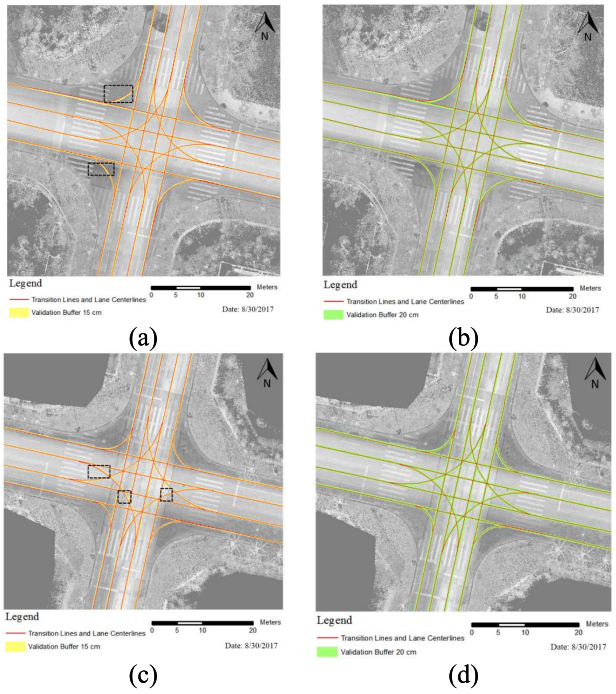



Fig. 13. Generated transition lines and lane centerlines within reference buffers. (a) Transition line validation using 15 cm buffer in Test Dataset 3. (b) Transition line validation using 20 cm buffer in Test Dataset 3. (c) Transition line validation using 15 cm buffer in Test Dataset 4. (d) Transition line validation using 20 cm buffer in Test Dataset 4.

Reference transition lines and lane centerlines were manually interpreted on the GRF images using ArcGIS v10.2.2. Centered by the manually labeled transition lines and lane centerlines, reference buffer zones with different widths, which indicate the range between the buffer zone boundaries and the manually interpreted transition lines and lane centerlines, were created and overlapped with the generated transition lines and lane centerlines.

The calculated values in both recall and miscoding of the generated transition lines and lane centerlines from Test Datasets 3 and 4 are listed in Table VIII. Reference buffers with the width of 15 cm and 20 cm were created to evaluate the performance of the proposed method. When conducting 15 cm BOS to the lines, the recall of Test Datasets 3 and 4 achieved 91.83% and 90.98%, respectively, and the miscoding was 7.01% and 7.82%, respectively. When conducting 20 cm BOS to the lines, the proposed method was capable of achieving 100% recall and 0% miscoding in Test Datasets 3 and 4. With the increased width of reference buffers, the miscoding values decreased, which demonstrated that most of the generated transition lines and lane centerlines were located within the precision allowable reference buffers. Therefore, the proposed method can provide a 20 cm-level localization accuracy to ensure the safety of autonomous vehicles and the precision of HDRMs.

In addition, Fig. 13 presents the generated transition lines and lane centerlines in Test Datasets 3 and 4 within 15 cm and 20 cm width of reference buffers. The black rectangles in Fig. 13 (a) and (c) indicate the miscoding parts in Test Datasets 3 and 4 when the width of reference buffers was set as 15 cm. Furthermore, it is identified that generated transition lines and lane centerlines are completely located within the reference buffers with 20 cm in width.

## V. CONCLUSION

In this paper, we have presented a semi-automated method for transition line generation in road intersections from MLS point clouds. Four test datasets have been used in this study to evaluate the performance of the proposed method. In the validation of lane marking extraction using manually interpreted lane marking points, the method achieved average precision, recall, and $F_1$-score of 90.80%, 92.07%, and 91.43%, respectively. The success rate of transition line generation was 96.5%. Moreover, the BOS method validated that the proposed method can generate lane centerlines and transition lines within a 20-cm range of reference path. In addition, the comparative study demonstrated that the proposed algorithm outperformed the other three existing methods in extracting road marking from MLS point clouds in road intersections.

This paper concludes that the proposed stepwise methodology can generate transition lines and lane centerlines in T-intersections and cross-intersections from MLS point clouds to provide highly accurate navigation and localization services for autonomous vehicles. It also provides a reliable solution to overcome the challenges in the generation of HDRMs for worldwide automotive manufacturers and mapping companies such as HERE, Google, TomTom, and Baidu.

## REFERENCES

[1] C. Thorpe, M. Herbert, T. Kanade, and S. Shafter, "Toward autonomous driving: The CMU Navlab. II. Architecture and systems," *IEEE Expert*, vol. 6, no. 4, pp. 44–52, Aug. 1991.

[2] *Preliminary Statement of Policy Concerning Automated Vehicles*, Nat. Highway Traffic Saf. Admin., Washington, DC, USA, May 2013, pp. 1–14.

[3] D. J. Fagnant and K. Kockelman, "Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations," *Transp. Res. A, Policy Pract.*, vol. 77, pp. 167–181, Jul. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0965856415000804

[4] M. Schreiber, C. Knöppel, and U. Franke, "LaneLoc: Lane marking based localization using highly accurate maps," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, vol. 36, no. 1, pp. 449–454.

[5] M. Buehler, K. Iagnemma, and S. Singh, "The DARPA urban challenge," *Auto. Vehicles City Traffic*. Berlin, Germany: Springer, 2009, pp. 125–163.

[6] J. D. Wegner, J. A. Montoya-Zegarra, and K. Schindler, "A higher-order CRF model for road network extraction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, vol. 9, no. 4, pp. 1698–1705.

[7] 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.

[8] L. Wang, "Semi-automated generation of high-accuracy digital terrain models along roads using mobile laser scanning data," M.S. thesis, Dept. Geogr. Environ. Manage., Univ. Waterloo, Waterloo, ON, Canada, 2016.

[9] 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.

[10] D. Betaille and R. Toledo-Moreo, "Creating enhanced maps for lane-level vehicle navigation," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 4, pp. 786–798, Dec. 2010.

[11] J. Du *et al.*, "CAMP enhanced digital mapping project—Final report," U.S. Dept. Transp., Washington, DC, USA, 2004. [Online]. Available: http://www.nhtsa.gov/DOT/NHTSA/NRD/Multimedia/PDFs/Crash%20Avoidance/2004/FinalRept_111904.pdf

[12] J. Liu, B. Cai, Y. Wang, and J. Wang, "Generating enhanced intersection maps for lane level vehicle positioning based applications," *Procedia—Social Behav. Sci.*, vol. 96, pp. 2395–2403, Nov. 2013.

[13] T. Zhang, D. Yang, T. Li, K. Li, and X. Lian, "An improved virtual intersection model for vehicle navigation at intersections," *Transp. Res. C, Emerg. Technol.*, vol. 19, no. 3, pp. 413–423, Jun. 2011.

[14] C. P. McElhinney, P. Kumar, C. Cahalane, and T. McCarthy, "Initial results from European road safety inspection (EURSI) mobile mapping project," in *Proc. ISPRS Commun. V Tech. Symp.*, 2010, pp. 440–445.

[15] B. Yang, L. Fang, and J. Li, "Semi-automated extraction and delineation of 3D roads of street scene from mobile laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 79, pp. 80–93, May 2013.

[16] L. Yan, H. Liu, J. Tan, Z. Li, H. Xie, and C. Chen, "Scan line based road marking extraction from mobile LiDAR point clouds," *Sensors*, vol. 16, no. 6, p. 903, Jun. 2016.

[17] B. Riveiro, H. González-Jorge, J. Martínez-Sánchez, L. Díaz-Vilariño, and P. Arias, "Automatic detection of zebra crossings from mobile LiDAR data," *Opt. Laser Technol.*, vol. 70, pp. 63–70, Jul. 2015.

[18] C. Cabo, A. Kukko, S. García-Cortés, H. Kaartinen, J. Hyyppä, and C. Ordoñez, "An algorithm for automatic road asphalt edge delineation from mobile laser scanner data using the line clouds concept," *Remote Sens.*, vol. 8, no. 9, p. 740, Sep. 2016.

[19] H. Guan, J. Li, Y. Yu, C. Wang, M. Chapman, and B. Yang, "Using mobile laser scanning data for automated extraction of road markings," *ISPRS J. Photogramm. Remote Sens.*, vol. 87, pp. 93–107, Jan. 2014.

[20] H. Wang *et al.*, "Road boundaries detection based on local normal saliency from mobile laser scanning data," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 10, pp. 2085–2089, Oct. 2015.

[21] M. Soilán, B. Riveiro, J. Martínez-Sánchez, and P. Arias, "Segmentation and classification of road markings using MLS data," *ISPRS J. Photogramm. Remote Sens.*, vol. 123, pp. 94–103, Jan. 2017.

[22] Z. Liu, J. Wang, and D. Liu, "A new curb detection method for unmanned ground vehicles using 2D sequential laser data," *Sensors*, vol. 13, no. 1, pp. 1102–1120, Jan. 2013.

[23] H. Luo *et al.*, "Patch-based semantic labeling of road scene using colorized mobile LiDAR point clouds," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 5, pp. 1286–1297, May 2015.

[24] S. Xu, R. Wang, and H. Zheng, "Road curb extraction from mobile Lidar point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 996–1009, Feb. 2017.

[25] P. Kumar, C. P. McElhinney, P. Lewis, and T. McCarthy, "Automated road markings extraction from mobile laser scanning data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 32, pp. 125–137, Oct. 2014.

[26] 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. 16, no. 5, pp. 2457–2466, Oct. 2015.

[27] M. Cheng, H. Zhang, C. Wang, and J. Li, "Extraction and classification of road markings using mobile laser scanning point clouds," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 3, pp. 1182–1196, Mar. 2017.

[28] 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.

[29] Y. Yu, J. Li, H. Guan, F. Jia, and C. Wang, "Learning hierarchical features for automated extraction of road markings from 3-D mobile LiDAR point clouds," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 2, pp. 709–726, Feb. 2015.

[30] R. Toledo-Moreo, D. Betaille, and F. Peyret, "Lane-level integrity provision for navigation and map matching with GNSS, dead reckoning, and enhanced maps," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 100–112, Mar. 2010.

[31] J. Du and M. J. Barth, "Next-generation automated vehicle location systems: Positioning at the lane level," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 48–57, Mar. 2008.

[32] C. Guo, K. Kidono, J. Meguro, Y. Kojima, M. Ogawa, and T. Naito, "A low-cost solution for automatic lane-level map generation using conventional in-car sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2355–2366, Aug. 2016.

[33] A. Holgado-Barco, B. Riveiro, D. González-Aguilera, and P. Arias, "Automatic inventory of road cross-sections from mobile laser scanning system," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 1, pp. 3–17, Jan. 2017.

[34] E. Catmull and R. Rom, "A class of local interpolating splines," *Comput. Aided Geometric Des.*, vol. 74, pp. 317–326, 1974.

[35] E. F. Moore, "The shortest path through a maze," in *Proc. Int. Symp. Theory Switching*, vol. 30, 1959, pp. 285–292.

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

[37] *Ministry of Housing and Urban-Rural Development of the People's Republic of China & General Administration of Quality Supervision, Inspection and Quarantine of the People's Republic of China, Code for Layout of Urban Road Traffic Signs and Markings*, Standard GB 51038, 2015.

[38] R. A. Jarvis, "On the identification of the convex hull of a finite set of points in the plane," *Inf. Process. Lett.*, vol. 2, no. 1, pp. 18–21, Mar. 1973.

[39] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "33.3: Finding the convex hull," in *Introduction to Algorithms*, 2nd ed. Cambridge, MA, USA: MIT Press, 2001, pp. 955–956.

[40] R. G. Congalton and K. Green, *Assessing the Accuracy of Remotely Sensed Data: Principles and Practices*. Boca Raton, FL, USA: CRC Press, 2008, pp. 55–59.

[41] C. J. van Rijsbergen, *Information Retrieval*, 2nd ed. London, U.K.: Butterworth, 1979, pp. 112–140.

[42] M. Sokolova, N. Japkowicz, and S. Szpakowicz, "Beyond accuracy, F-score and ROC: A family of discriminant measures for performance evaluation," in *Proc. Australas. Joint Conf. Artif. Intell.*, vol. 4304, Dec. 2006, pp. 1015–1021.

[43] H. Zhang, "Rapid inspection of pavement markings using mobile laser scanning point clouds," M.S. thesis, Dept. Geogr. Environ. Manage., Univ. Waterloo, Waterloo, ON, Canada, 2016.

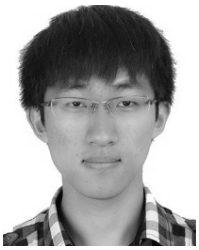
[44] X. Chen, M. Stroila, and R. Wang, "Next generation map making: Geo-referenced ground-level LIDAR point clouds for automatic retro-reflective road feature extraction," in *Proc. 17th ACM SIGSPATIAL, Int. Conf. Adv. Geograph. Inf. Syst.*, Seattle, WA, USA, Nov. 2009, pp. 4–6.

[45] H. Tveite, "An accuracy assessment method for geographical line data sets based on buffering," *Int. J. Geograph. Inf. Sci.*, vol. 13, no. 1, pp. 27–47, Jan. 1999.


**Chengming Ye** received the Ph.D. degree in remote sensing from the Chengdu University of Technology, China. He was a Visiting Scholar with the Mobile Sensing and Geodata Science Group, University of Waterloo, for 12 months. He is currently an Associate Professor with the Key Laboratory of Earth Exploration and Information Technology, Ministry of Education, Chengdu University of Technology. His research interest includes remote sensing and its applications.

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

He is currently a Professor and the Head of the Mobile Sensing and Geodata Science Group, Department of Geography and Environmental Management, University of Waterloo, Canada. He has co-authored more than 400 publications, more than 150 of which were published in refereed journals, including IEEE-TGRS, IEEE-TITS, JSTARS, ISPRS-JPRS, and RSE. His research interests include information extraction from mobile light detection and ranging (LiDAR) point clouds and from earth observation images. He is the Chair of the ISPRS WG I/2 on LiDAR, Air- and Space-borne Optical Sensing from 2016 to 2020 and the ICA Commission on Sensor-Driven Mapping from 2015 to 2019, and the Associate Editor of the IEEE-TITS and JSTARS.



**Han Jiang** received the B.Eng. degree in GIScience from the China University of Geosciences, Beijing, China, in 2015, and the M.Sc. degree in geomatics from the Mobile Sensing and Geodata Science Group, University of Waterloo, Canada, in 2017. His M.Sc. dissertation was about the development of mobile light detection and ranging (LiDAR) point-based algorithms and software tools for creating high-definition road maps to support autonomous vehicles.

**He Zhao** received the B.Eng. degree in electrical engineering from Qingdao University, China, in 2015, and the M.Sc. degree in geomatics from the Mobile Sensing and Geodata Science Group, University of Waterloo, Canada, in 2017. His master dissertation was about the development of mobile laser scanning (MLS) point-based algorithms and software tools for creating high-definition road maps to support autonomous vehicles.

**Lingfei Ma** received the B.Eng. degree in GIScience from the China University of Geosciences, Beijing, China, in 2015, and the BES degree in geomatics and the M.Sc. degree in geomatics, specializing in remote sensing, from the University of Waterloo, Canada, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the Mobile Sensing and Geodata Science Group. His research interests include mobile laser scanning, point cloud processing, feature extraction from point clouds, and high-definition road mapping for autonomous vehicles.

**Michael Chapman** received the Ph.D. degree in photogrammetry from Laval University, Quebec City, QC, Canada.

He is currently a Professor of geomatics with the Department of Civil Engineering, Ryerson University, Toronto, ON, Canada. He has co-authored over 160 technical papers. His research interests include algorithms and processing methods for airborne sensors using GPS/INS geometric processing of digital imagery in industrial environments, terrestrial imaging systems for transportation infrastructure mapping, and algorithms, and processing strategies for metrology applications.