ISPRS Journal of Photogrammetry and Remote Sensing 156 (2019) 184–201

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Recovery of urban 3D road boundary via multi-source data

Chenglu Wen[a], Changbin You[a], Hai Wu[a], Cheng Wang[a,b,*], Xiaoliang Fan[a,b], Jonathan Li[a,c]

[a] Fujian Key Laboratory of Sensing and Computing for Smart Cities and the School of Information Science and Engineering, Xiamen University, Xiamen 361005, China
[b] Fujian Collaborative Innovation Center for Big Data Applications in Governments, Fuzhou 350003, China
[c] Departments of Geography and Environmental Management/Systems Design Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada

ARTICLE INFO

Keywords:
Point clouds
Mobile laser scanning
3D road boundary
Multi-source data
GPS trajectory data
Remote sensing image

ABSTRACT

The mapping of road boundaries provides critical information about roads for urban road traffic safety. This paper presents a deep learning-based framework for recovering 3D road boundary using multi-source data, which include mobile laser scanning (MLS) point clouds, spatial trajectory data, and remote sensing images. The proposed road recovery method uses extracted 3D road boundaries from MLS point clouds as inputs. First, after automatic erroneous boundary removal, a CNN-based boundary completion model completes road boundaries. Then, to refine the imperfect road boundaries, road centerlines generated from dynamic taxi GPS trajectory data and remote sensing images are used as completion guidance for a generative adversarial nets model to obtain more accurate and complete road boundaries. Finally, after associating a sequence of taxi GPS recorded trajectory points with the correct 3D road boundaries, inherent geometric road characteristics and road dynamic information are extracted from the complete boundaries and taxi GPS trajectory data, respectively. The testing dataset contains two urban road MLS datasets, and the KITTI dataset. The experimental results on point clouds from different sensors demonstrate our proposed method is effective and promising for recovering 3D road boundary and extracting road characteristics.

1. Introduction

As the main components of a map, roads provide the impetus for the development of a city and give fundamental support for the development of a modern metropolis. In addition, relevant road information plays an important role in various applications such as city planning (Hamilton et al., 2005), road transportation systems (Litescu et al., 2016), driver assistance systems (Mogelmose et al., 2012), and disaster management (Raber et al., 2007). Mobile laser scanning (MLS) point clouds, which have millimeter-level accuracy and contain a wealth of 3D detailed information, have been successfully applied to much road-related research, such as road surface analysis, road marking extraction (Cheng et al., 2017), and roadside object extraction (Yu et al., 2016). As the most succinct topology of a road, the road boundary is one of the essential components of a high definition road map. Because of high-density, long range, and insensitivity to lighting conditions, MLS point cloud data are suitable for extracting highly accurate and complete road boundary. Extracting road boundary from MLS data is usually defined as curb detection along with road surface extraction. However, because of occlusions and variations in data density during collection, raw MLS point clouds are usually incomplete. Thus, the road boundary obtained from MLS point clouds is usually incomplete with many gaps.

The straightforward way to solve the problem of acquiring incomplete data is to re-scan and extract the incomplete parts of the road again, at the extra cost of time, workforce, and money. A more practical approach is to recover incomplete road boundaries. The gaps can be directly completed by linear interpolation, but it is prone to curvature loss in some incomplete curves. The main challenges for recovering 3D urban road boundaries are summarized as follows: (1) Incomplete raw point clouds are collected because of occlusion from cars or vegetation (Fig. 1(a)), or the limited sensing range of the sensor. Incomplete 3D road boundaries are obtained in some situations, e.g., no curb or low curb (see Fig. 1(b)), because of incomplete raw data and the error, or limitation, in the extraction method. (2) The uncertainties of gaps in urban road boundaries make it difficult to determine whether or not the gaps should be completed. Shown in Fig. 1(c), in the median of the roadway, is a turning area, which cannot be completed. Shown in Fig. 1(a) and (b) are gaps that require completion. Thus, automatically achieving an accurate, complete road boundary is an important issue that remains difficult.

With different types of data sources, different forms of roads, with different characteristics, are obtained (see Table 1). Roads extracted

* Corresponding author at: Fujian Key Laboratory of Sensing and Computing for Smart Cities and the School of Information Science and Engineering, Xiamen University, Xiamen 361005, China.
E-mail addresses: clwen@xmu.edu.cn (C. Wen), cwang@xmu.edu.cn (C. Wang).

https://doi.org/10.1016/j.isprsjprs.2019.08.010
Received 7 March 2019; Received in revised form 26 June 2019; Accepted 12 August 2019

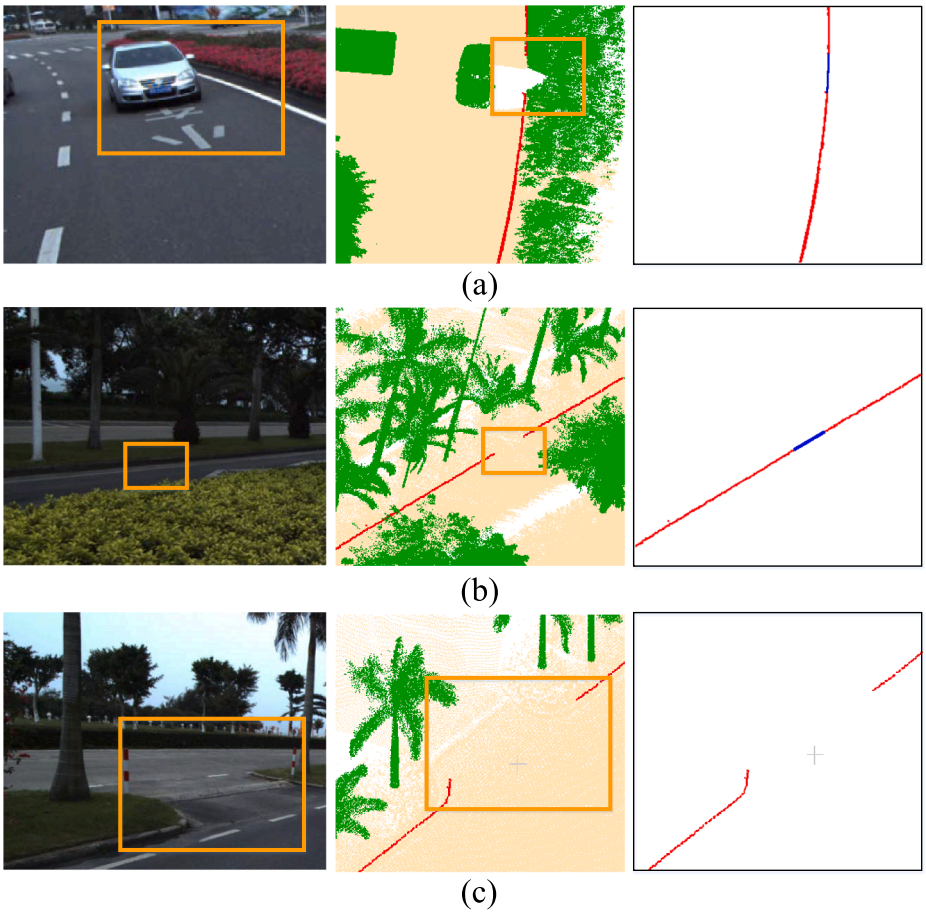

Fig. 1. Examples of incomplete road boundaries extracted from MLS data. From left to right: images, raw road point clouds with boundaries (in red), and target complete boundaries (the blue lines represent completion parts). Yellow boxes represent the gaps in the extracted boundaries. (a) Missing data because of occlusion. (b) Low curb. (c) A turning area in the median of the roadway. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Summary of different forms of roads.

Data source		Form of road	Scene/publications
Single source	Remote sensing image	Surface	Urban/Wegner et al. (2013), Li et al. (2014) Suburban and countryside/Gao et al. (2018)
		Centerline (graph structure)	Urban/Shackelford and Davis (2003), Péteri et al. (2006) Urban and rural/Tupin et al. (1998), Hedman et al. (2010), Zang et al. (2017) Highway/Hu et al. (2007)
		Surface & Centerline	Urban and rural/Cheng et al. (2017)
	GPS	Centerline	Urban/Xie et al. (2014), Ouyang et al. (2014) Urban and freeway/Schroedl et al. (2004)
	ALS	Surface	Urban/Zhu and Mordohai (2009), Li et al. (2015)
		Centerline	Downtown and surrounding residential areas/Zhao and You (2012) Urban/Zhao et al. (2011), Peng and Gao (2011), Hu et al. (2014)
	MLS	Boundary (Edge or curb)	Residential and urban/Xu et al. (2016) Urban/Hervieu and Soheilian (2013), Reza et al. (2017) Urban and industrial region/Zai et al. (2017)
Multi-source	ALS with orthophotos	Surface & Curb	Urban/Vo et al. (2015)
	MLS with EO images	Boundary	Rural/Yeo et al. (2016)

from remote sensing images are usually road surfaces or centerlines that interconnect as net structures. In urban scenes, extracted roads in remote sensing images are usually incomplete and inaccurate because the urban type of road is complex with the ground occluded by influential objects. Additionally, roads in low-resolution images are fuzzy, and roads in high-resolution images have more influencing factors, both of which are inconducive to road extraction. Airborne laser scanning (ALS) data, which provide 3D geographic positions of the ground, have been a good source of data for road surface (Li et al., 2015) or centerline (Hu et al., 2014) extraction in urban scenes. Roads obtained from ALS data usually contain thick lines, and, because of the uneven distribution of the LiDAR points, there are many broken lines.

Spatial trajectory data, especially a large-scale quantity of GPS trajectory data collected from cars, like taxis, traveling on urban roads over extended periods, can cover a whole road (including arterial roads, branch roads, etc.). Furthermore, dynamic traffic information for each road (such as traffic flow, vehicle speeds, road congestion, etc.), which has theoretical scientific significance for road network planning and design, can be extracted from this type of data. Spatial trajectory data have been widely used to extract road centerline (Ouyang et al., 2014).

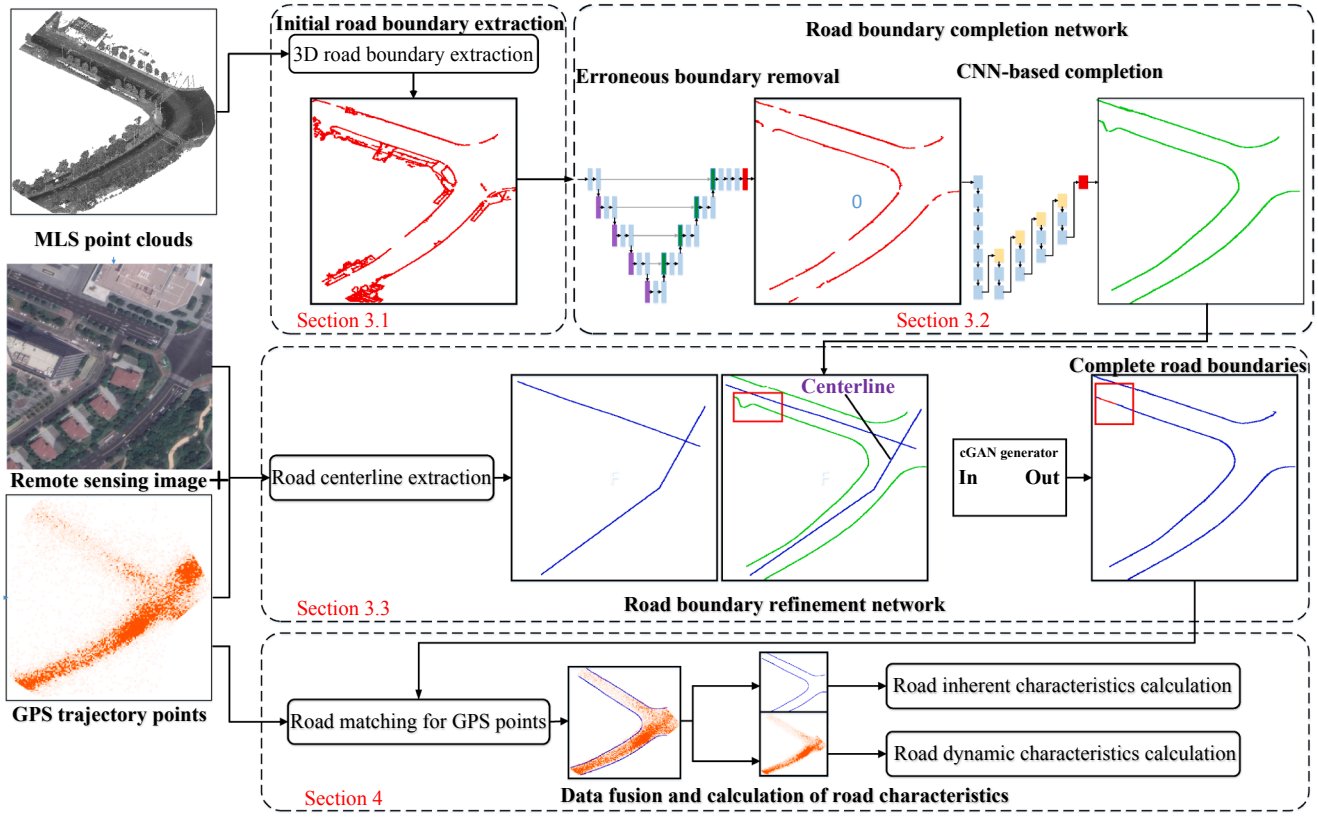

Fig. 2. Workflow of the proposed framework.

Because of covering a wide range of roads, the completeness level of roads obtained from a massive spatial trajectory data is higher than the completeness level from other data. However, accuracy is worse because of the relatively poor accuracy of the GPS positioning in an urban area. The completeness of a spatial trajectory data provides a potential to build the relationship among different roads and guide the completion of a more detailed road, especially for solving the uncertainties of gaps and determining whether to be completed.

Roads, extracted from different data sources, have different benefits and drawbacks. If an MLS-based road boundary is formed by fusing extraction results from other data sources, a more complete, higher accurate road boundary, rich in detailed information, is likely to be obtained. However, varying resolutions and accuracies of different sources of data cause difficulty when fusing them to guide the recovery of the road boundary.

In this paper, we propose a deep learning-based 3D road boundary recovery framework using multi-source data. The multi-source data include MLS point clouds, spatial trajectory data, and remote sensing images. The workflow of our framework is shown in Fig. 2. More specifically, the 3D road boundaries extracted from MLS point clouds are inputs of the framework. The initial road boundary lines, which usually include some erroneous segments, are first cleaned by a U-Net segmentation model based on images converted from boundary point clouds. Then, a CNN-based boundary completion model, which conserves well the thickness and curvature of a line segment, is used to efficiently complete boundary lines images. Then, considering the imperfect local details of a road boundary that result from CNN-based boundary completion, based on the centerlines generated from spatial trajectory data and remote sensing images, refined road boundaries are obtained using a conditional generative adversarial network (cGAN). Last, to combine dynamic road characteristics (e.g., traffic flow) with inherent geometric road characteristics (such as width, curvature) and achieve dynamic visualization (traffic data) of a road boundary, a road matching method is developed to associate recorded drifted taxi GPS trajectory points with the correct 3D road boundary.

2. Related work

2.1. Studies on road extraction

Previous studies have reported that roads extracted can be in different forms (as shown in Table 1). The most common methods are based on different kinds of high-resolution images, such as satellite (Péteri et al., 2006; Li et al., 2014; Shackelford and Davis, 2003), synthetic aperture radar (SAR) (Tupin et al., 1998; Hedman et al., 2010), and aerial (Wegner et al., 2013; Hu et al., 2007; Poz et al., 2006; Gao et al., 2018). Most of the studies used road features to assist in road extraction. However, the performance of road extraction in an image is affected by the complexity of the scene. Buildings may mask a road and reduce visibility. For monitoring purpose, a large quantity of taxi GPS trajectory data can be acquired, making it possible to extract the road based on the GPS trajectory (Schroedl et al., 2004; Xie et al., 2014; Ouyang et al., 2014). Apart from imagery and GPS trajectory data, ALS and MLS point clouds have been exploited to extract roads (Zhao and You, 2012; Reza et al., 2017). However, because of the voxelization of the point clouds, the results are not so accurate and complete. Moreover, road extraction methods using ALS and MLS data usually suffer from incomplete data because of occlusion.

2.2. Studies on multi-source data-based road extraction

Given the drawback of using single source data, methods that combine ALS point clouds with high-resolution aerial imagery data were proposed to obtain color information (Gao et al., 2013; Hu et al., 2004). Vo et al. (2015) used a quadtree-based region growing method to extract roads from colored point clouds formed by fusing ALS point

clouds and orthophotos. Hu et al. (2004) developed an iterative Hough Transform-based approach to extract urban grid networks using ALS data and aerial images. To create a 2D semantic map used to generate dirt road boundary measurements, Yeo et al. (2016) fused the output from a 3D LiDAR point cloud classifier and an image scene parser. These methods introduce image information to improve the road extraction performance to some extent, but they do not handle missing parts of the road.

2.3. Studies on road completion

To remedy the problem of missing parts in a road caused by method limitations and occlusions, Zhao et al. (2011) proposed a geometrical guided gap filling method, based on ALS point clouds, to infer the extracted missing centerlines. To determine whether the gaps between true centerlines should be connected, the following four geometrical criteria are calculated: collinearity, proximity, curvature, and length. For MLS point clouds, Xu et al. (2016) proposed an energy function to extract candidate points of curbs and used a least cost path model to refine candidate points. Inspired by Kalman filter models, Hervieu and Soheilian (2013) designed a boundary detection model to estimate a boundary in two linear parts with many occlusions. However, the prediction is uncertain. If no 3D curb profile pattern is observed in a bend, the model will fail to predict the shape of the roadside. Previously published studies have many limitations and have not dealt well with highly complex and uncertain urban road boundaries. Thus, it is vitally important to develop a robust and highly accurate method for road boundary completion.

3. Road boundary extraction and recovery

3.1. 3D road boundary extraction from MLS data

In our previous work (Zai et al., 2017), we developed a road boundary extraction method, which includes the following two steps: (1) By selecting smooth points as seeds and using several attributes that assign points to facets centered on these seeds, facet supervoxels are generated, which are then used to extract initial boundaries by using an α-shape method. (2) A weighted, undirected graph to apply energy minimization, using a graph-cuts method, separates initial boundary points into the road and non-road boundary points. The method mentioned above, as well as several other road extraction methods, provides road boundaries with different completeness to test the feasibility of the proposed road boundary completion method (see Section 5.3).

3.2. Road boundary completion network

3.2.1. Erroneous boundary removal

For urban roads, the extracted boundaries are always rough and have erroneous lines with occlusions from roadside vegetation, which may negatively influence the recovery of the road boundaries. Thus, the erroneous boundaries are removed first. Because of the irregularity of the erroneous boundaries, it is difficult to remove them effectively from a 3D point cloud form directly by a rule-based method.

In our method, 3D road boundary point clouds are first projected onto a horizontal plane and gridded as a 2D road boundary image with grid spacing $R_{\epsilon 1}$. Then, the erroneous boundary removal is treated as a binary classification (classify road boundaries into correct and erroneous boundaries). A U-Net model (Ronneberger et al., 2015) automatically classifies and removes the erroneous boundaries. This network model consists of encoder and decoder layers (Fig. 3). Each encoder layer is composed of convolution and max pooling operations. Each decoder layer is composed of deconvolution and convolution operations. The loss used to train the model is binary cross entropy. For training, 200 training samples are created by using (1) some extracted road boundary parts as input, and (2) the corresponding road boundary parts without erroneous boundaries by manually removing as the target. During training, the batch size and epoch are set at 2 and 100, respectively.

3.2.2. CNN-based model for 2D boundary completion

The missing parts in extracted road boundaries may have different structures with different curvature (e.g., straight line and circular curve) causing difficulty in directly completing road boundaries in the form of 3D point clouds. Recently, researchers have shown an increased interest in 3D shape completion based on fully/weakly-supervised learning (Dai et al., 2017; Stutz and Geiger, 2018). Also, deep learning-based image inpainting/translation showed good performance (Zhu et al., 2017; Isola et al., 2017). Sasaki et al. (2017) proposed a data-driven approach to detect automatically and fill the missing regions of line drawings. Their approach has shown good performance on line segment completion, conserving thickness and curvature. Inspired by image inpainting, we developed a CNN-based method for boundary completion.

Based on the results obtained after erroneous boundary removal, a CNN-based completion model (Fig. 3) is then used to complete gaps. This model consists only of convolution layers and upsampling layers. Except for the first layer, using a kernel size of 5×5, all the convolution layers use a kernel size of 3×3. For activation functions, all convolution layers use ReLU, except for the final one, where the Sigmoid function is used instead. When training, batch normalization is used after each convolution layer, except for the last. The sizes of the input and output images are not fixed, and output images are the same size as the input. The loss used to train our model is Mean Squared Error (MSE), which represents the difference between the input S and the target T as follows:

$$loss(T, S) = \frac{1}{|N|} \sum_{p \in N} (T_p - S_p)^2 \tag{1}$$

where N is the set of all pixels in the image, and S_p and T_p are the values at the pixel p in the input and the target images, respectively. In this section, the training data used is that of Sasaki et al. (2017).

Because of the limitation of the training dataset, the model may be unable to handle gaps in images that are too large. Thus, to guarantee that most of the gaps are detected automatically and completed by the model, to find the best result, 3D boundary point clouds after erroneous boundary removal are projected onto images with a grid spacing $R_{\epsilon 2}$. After obtaining an image-based completion model, a 2D boundary completion result is obtained by using the road boundary image as model input. To maintain fine and smooth boundary point clouds, only the completed 2D pixels by the image completion model are considered to transform and add back to 3D road boundary point clouds.

3.2.3. Completion in 3D form

Before transforming the completed pixels back to 3D, it is essential to detect the gaps in 3D. Here, the gap detection problem is defined as matching the endpoint pair of gaps. First, a Euclidean distance clustering method is applied to partition the boundary points into separated line clusters. For each line cluster (Fig. 4(a)), the endpoint (e.g. point p_m in blue) is the point where the neighborhood points are all on the same side and belong to the line cluster. Using this feature, the endpoints of each line cluster are found and added to set $S_e\{p_1, p_2, \cdots, p_n\}$. Then, gap detection is achieved by pairing the points in S_e. To avoid the mismatch problem, where two matching endpoints belong to different roadside boundaries, the taxi GPS trajectory data, which provide the driving route of the vehicle and give the approximate location of the road, are used to separate the inner and outer boundaries. For point $p_s \in S_e$, the matching endpoint p_e, from the neighborhood points of p_s, must satisfy the following conditions (Fig. 4(b)):

$$Tra\overrightarrow{(p_s)p_e} \cdot Tra\overrightarrow{(p_s)p_s} > 0 \tag{2}$$

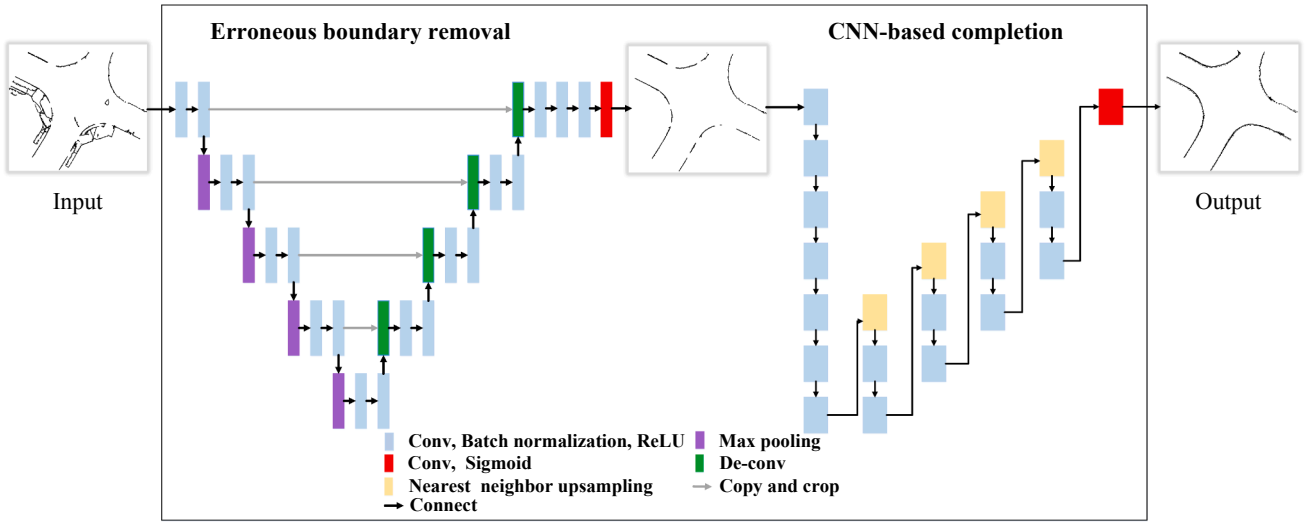

Fig. 3. The structure of the road boundary completion model.

$$D(p_s, p_e) < D(p_r, p_e) \tag{3}$$

where $Tra(p_s)$ represents the point nearest to p_s in the trajectory. $D(p_s, p_e) < D(p_r, p_e)$ means that point p_e is nearer to p_s.

Fig. 4(c) shows the endpoint pairs set S_{ep} of a part of road boundaries. The gap is between the endpoint pair that have the same color. Based on the gap endpoint pair set S_{ep}, the transformation relationship between point clouds and images is reserved and used to transform completed pixels to the newly added 3D points, which represent the missing boundary points. The process used in this paper to project 3D-2D-3D lines, as well as an analysis of precision loss in the projection process, were introduced in detail in our previous work (Wang et al., 2018).

3.3. Road boundary refinement network

Sometimes, because of the uncertainties of urban roads, results that are based on above completion method are imperfect. The problems are summarized as follows: (1) Gaps that should not be completed (Fig. 5(a)). (2) Some large gaps that should be completed (Fig. 5(b)). (3) Irregular completion structures (Fig. 5(c)). Thus, the boundaries from the above CNN-based model must be refined. Whether a gap should be completed can be determined based on the vehicle activities on both sides of the gap. In this paper, we innovatively introduce big taxi GPS trajectory data to generate road centerline, which reflects road structure to some extent, to guide the road boundary refinement with uncertainties.

3.3.1. Road centerline extraction using multi-source data

In our method, to extract road centerlines, we use GPS trajectories from taxis that show a global topology of the road. It is then used as guidance for solving the uncertainties of gaps and refining the imperfect road boundaries. Because a GPS receiver is easily susceptible to random noise and position drift during acquisition, there are many abnormal points in the collected data that affect further extraction of road centerlines. Thus, it is necessary first to remove the abnormal points from the original data.

Consider a taxi GPS trajectory point set, $T_p\{p_1, p_2, \cdots, p_i, \cdots, p_k\}$, where $p_i(c, t, lgt, lat, sp, dr)$ denotes a recorded taxi GPS trajectory point i, and the recorded values for the number of taxis, time, longitude, latitude, car speed, and direction are represented by c, t, lgt, lat, sp, and dr, respectively. Then, the abnormal point set is represented as:

$$A_p = \{p \in T_p | D(p_i(c, t), p_{i-1}(c, t-1)) > m \vee (p_i(sp = 0) \wedge p_i(dr = 0))\} \tag{4}$$

where $D(p_i(c, t), p_{i-1}(c, t-1)) > m$ means that a point i is added to A_p if the distance from point $i-1$ to point i is larger than the expected distance m, where points i and $i-1$ are collected from the same taxi in adjacent times t and $t-1$. $p_i(sp = 0) \wedge p_i(dr = 0)$ means that a point i is added to $A_p$if the value of the speed and direction are both equal to zero.

When a taxi is driving along a narrow road, the collected GPS trajectory points may show the same location because of traffic congestion. Such stagnated points, which affect the topology of the extracted road, also must be removed. Here, we use the density-based clustering method (Rodriguez and Laio, 2014) to find high-density areas, remove the trajectory points within these areas, and then use Gaussian filtering to ensure the smoothness of the overall trajectory. Based on the directional value of the points, including straight and curved, taxi GPS

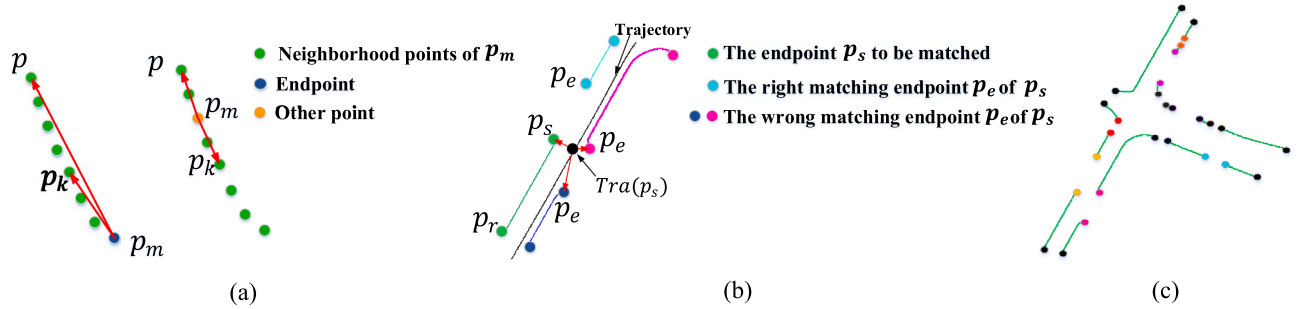

Fig. 4. (a) Endpoint and other points in line cluster. (b) Right and wrong matching endpoints of p_s. (c) Endpoint pairs with different colors (endpoints in black are unmatched points).

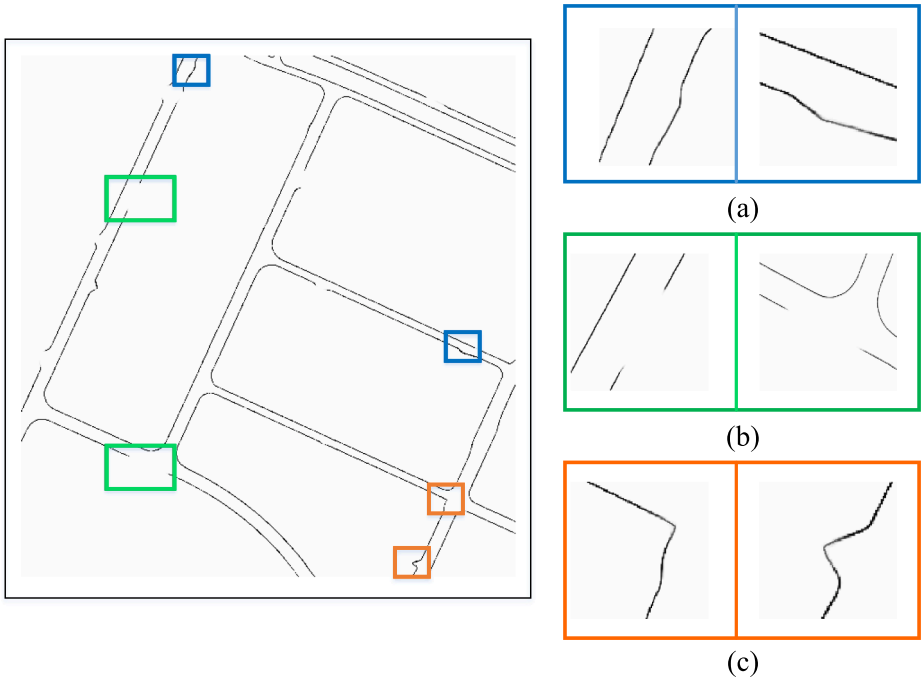

Fig. 5. Examples of imperfect completion results after road boundary completion.

trajectory points are clustered and segmented to be separated road points, for which the centerlines are fitted by a quartic polynomial fitting.

A wide collection range, obtaining information rapidly, and visualizing inaccessible areas are advantages of remote sensing images. Therefore, remote sensing images are also a good source for obtaining global road data. Because some urban roads in remote sensing images are seriously occluded by buildings, the results extracted from remote sensing images are incomplete and of low accuracy; however, the extracted results can still improve the structure of the road centerline obtained by taxi GPS trajectories. Here, we use a deep learning network (Zhou et al., 2018) to extract roads from remote sensing images. According to the global coordinate information of an image, the road pixels extracted from the image are then transformed to a 3D point cloud (height, z, set at 0) and merged with the road centerline. Some road centerlines are added or extended toward the road point clouds; therefore, the resulting final centerline is more complete.

Although it might drift and be inaccurate because of the substandard positioning accuracy of the taxi GPS trajectory data, the generated road centerline can match the accuracy of the road boundary point cloud and provide an important reference for road boundary refinement. Conversely, the road boundary point cloud can correct the deviation of the complementary taxi GPS trajectory points.

3.3.2. *cGAN-based road boundary refinement under uncertainties*

To solve the uncertainties of gaps and refine the road boundaries, we first determine whether the completed gap should be removed based on the road centerline. If the road centerline crosses the gap, revert it to the original structure. Thus, mistaken completion can be solved. However, the problems of gaps without completion and irregular completion structures remain unsolved.

Generative models, for example the GAN model (Goodfellow et al., 2014), have performed well in image restoration and image translation. Here, we introduce the conditional GAN (cGAN) model, which is an extended network of the GAN model, to solve the problems of gaps without completion (considered as an image restoration problem) and irregular completion structures (considered as an image translation problem). Guided by the centerline of a road, the cGAN-based model refines the imperfect road boundaries with rich details, especially for curved boundaries, which are difficult for rule-based methods to restore.

The GAN model contains two adversarial models: generator model, G, and discriminator model, D. Briefly, G produces an output that D cannot distinguish as a "fake" sample. G distinguishes the output of D from the "real" sample as much as possible. The GAN model learns mapping from a random noise vector z to output x. The cGAN model extends the GAN model by learning mapping from a condition y and a random noise vector z to an output x. For cGAN, refinement of a road boundary translates an image, with an imperfect road boundary line and centerline obtained from taxi GPS trajectories, into an image with a ground truth road boundary line. Here, a condition y represents an image with an imperfect road boundary line and centerline. An output x represents an image with a ground truth road boundary line; z is a random noise vector. By changing the condition y, the cGAN model generates different targets without modifying the generative network and easily applies them to other road types.

The cGAN-based model requires only a small amount of training data to achieve refinement of a road boundary. The training samples include: (1) manually cut correct boundaries and their corresponding centerlines, (2) imperfect boundaries (gaps without completion and irregular completion structures) and their corresponding centerlines generated from correct boundaries, and (3) imitated imperfect boundaries and their corresponding centerlines by hand-drawing.

Using training samples of gaps without completion and irregular completion structures, two corresponding image translation models are obtained after cGAN training. Since the cGAN model works on a 2D plane, each gap completion line with two side boundary lines and centerline is projected onto a 2D image. The road boundaries with gaps are broken off, which are different from the boundaries with an irregular complete structure; therefore, the road boundary samples, separated easily into two categories, will apply different cGAN models to refine (see Fig. 6). Because of the possibility of an undesired completed structure, the results of refining a gap are again input to the model for the problem of an irregular completion structure. After refining the imperfect completion gaps in the images, the refined results in the images are transformed back to the final complete 3D road boundaries by converting 2D pixels to 3D points.

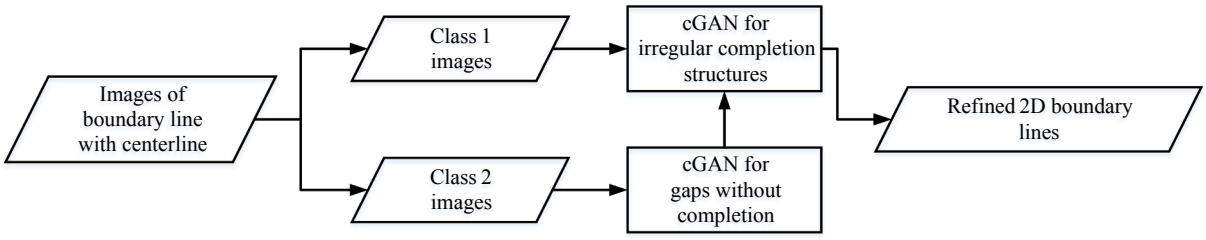

Fig. 6. Flow chart of 2D boundary line refinement.

4. Road matching and road characteristics calculation

It has been proven that geometric road parameters influence traffic accidents (Miaou and Lum, 1993; Karlaftis and Golias, 2002). In addition, for roads, some dynamic characteristics, such as traffic flows analyzed from GPS trajectory data of cars, contribute to reducing traffic accidents and further improving traffic facilities. Thus, an inventory of regular road information to obtain road characteristics is significant to traffic safety. Based on the road boundary and taxi GPS trajectory points, we present a method for matching taxi GPS trajectory points with corresponding road boundaries to achieve inherent and dynamic road characteristics calculation.

4.1. Inherent road characteristics calculation

With complete boundaries, some inherent geometric road characteristics, such as horizontal and vertical alignment parameters (see Fig. 7), which have strong relationships with traffic, are selected to calculate. Some required formulae to calculate inherent road characteristics are shown in Table 2.

(1) **Horizontal alignment parameters:** Horizontal alignment parameters contain road width, the curvature of horizontal curves, length of horizontal curves, and deflection angles. The deflection angle $\Delta\alpha$ is the exterior angle at the intersection of the two tangent lines. The length of the horizontal curve Δs is the arc length between P_{hs} and P_{he}. The curvature of a horizontal curve $\overline{C}$ is the mean curvature between P_{hs} and P_{he}.
(2) **Vertical alignment parameters:** Vertical alignment parameters contain vertical curve length (VCL), gradient change, and K-value. VCL is calculated using the curve integral of $f_v(x, z)$. The change in gradient Δg indicates the algebraic difference of the gradient between vertical starting point P_{vs} and vertical end point P_{ve}.

4.2. Road matching for dynamic characteristics calculation

Because of poor signals, communication anomalies, or GPS positioning errors, many recorded taxi GPS trajectory points cannot be located on the correct road, leading to incorrect dynamic road information. We match taxi GPS recorded trajectory points to the correct road of an underlying 3D road boundary by introducing the map matching method, which matches a serial of recorded location points with edges in an existing street graph or the network of a digital map. The HMM-based method, which finds the hidden sequence that produces this observation sequence for a given observation sequence, is highly accurate for map matching under certain conditions. In our method, we introduce the HMM-based matching method to solve the matching problem between 3D road boundaries and taxi GPS trajectory points. The observation sequence represents a series of taxi GPS trajectory points; the hidden sequence represents the proper road segments to which these points belong.

The key to applying map matching to 3D boundaries is to generate the same form of 2D road line segments from 3D boundaries. We first partition the obtained 3D road boundary into a set of boundary segments having inner and outer boundary lines. Considering the consistency of the boundaries after completion, the inner and outer boundary lines for each boundary segment are separated by distance clustering. Then, every 20 m on a longer boundary line, we select a point and find its nearest point in a shorter boundary line. Based on the number of road lanes, which is calculated by the road width (calculated as the distance between a point in a longer boundary line and the nearest point in a shorter boundary line) and each lane width (about 7.5 m in our selected roads), the number of centerlines is determined. Each centerline endpoint is obtained from the coordinate values of the pair points. With these points, we generate a 2D centerline shapefile for each 3D road boundary segment. The shapefile is a data format used as road segments for map matching. After generating the centerline shapefile from 3D road boundaries, our road matching problem is transformed into a map matching problem.

Given a taxi GPS trajectory point gp_t, for road matching, there may be more than one road segment candidate within a certain distance. The candidate points, which are the projection points of gp_t on these candidate segments, are regarded as vertices in Markov chains. For each road segment s_i, there is a measurement probability that the point

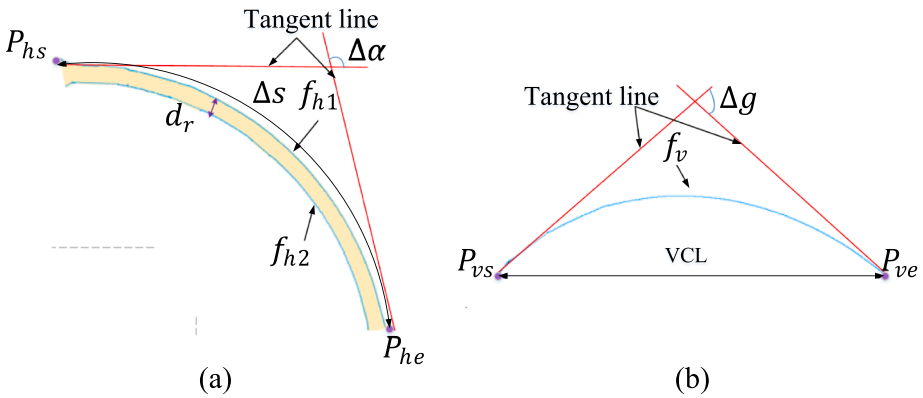

Fig. 7. (a) Measurement of horizontal alignment parameters. (b) Measurement of vertical alignment parameters.

Table 2
Formulae for inherent road characteristics calculation.

Inherent road characteristics	Formulae	Parameter name
Horizontal alignment parameters	$f_{h1}(x, y) = 0$	Horizontal projection outer boundary
	$f_{h2}(x, y) = 0$	Horizontal projection inner boundary
	$\overline{C} = \left\lvert \frac{\Delta\alpha}{\Delta s} \right\rvert$	Curvature of horizontal curve
Vertical alignment parameters	$f_v(x, z) = 0$	Vertical curve (Vertical projection boundary curve of a hilly road)
	$VCL = \int f_v(x, z)ds$	Vertical curve length
	$K = \frac{VCL}{\Delta g}$	Flatness of vertical curve

would likely be observed if it is on the road segment. The weights between adjacent GPS candidate points, which are represented by transition probability, are regarded as edges in a Markov chain. We introduce measurement and transition probabilities (Newson and Krumm, 2009) to infer the correct road segment of each taxi GPS trajectory point.

Measurement probabilities, modeled as zero-mean Gaussian, take the following form:

$$p(gp_t|s_i) = \frac{1}{\sqrt{2\pi}\sigma_{pt}} e^{-\frac{(\|gp_t - x_{t,i}\|_{gc})^2}{2\sigma_{pt}^2}} \tag{5}$$

where $x_{t,i}$ represents the candidate point of the measured point gp_t on road segment i; gc indicates great circle distance (the shortest surface path between gp_t, and $x_{t,i}$); σ_{pt} is the standard deviation of the GPS measurements (set at 20 m in the paper).

Transition probabilities, which assume that the transitions, whose driving distance is about the same as the great circle distance between the measured points gp_t andgp_{t+1}, are modeled according to the following formula:

$$dis_t = |\|gp_t - gp_{t+1}\|_{greatcircle} - \|x_{t,i} - x_{t+1,j}\|_{route}| \tag{6}$$

where $\|x_{t,i} - x_{t+1,j}\|_{route}$ is the driving distance between the pair of candidate points $x_{t,i}$ and $x_{t+1,j}$. According to Formula (7), the smaller the difference in distance between the great circle and the route, the more likely the results will be matched correctly.

With the measurement and transition probabilities mentioned above, the Viterbi algorithm (Viterbi, 1967) is used to compute the best path through the HMM lattice and infer the correct road segment for each taxi GPS trajectory point. The dynamic characteristics of the correct taxi GPS trajectory points on each road are obtained, e.g., traffic flow and traffic density are calculated as follows:

$$Tf = \text{num}(\text{p} \in (Q_p - A_p) \wedge \text{p(t)} \in timeRange \wedge \text{p(lgt, lat)} \in roadArea \tag{7}$$

$$Td = \frac{Tf}{Length(road)} \tag{8}$$

where Tf (traffic flow) refers to the total number of cars on a specified road area within a certain period, and Td (traffic density) refers to the number of cars in the unit length (usually one km) of a road within a certain period. In addition, from the direction the taxis are traveling, the stream direction of the road is also ascertained.

5. Results and discussion

5.1. Dataset

The data in this study include MLS point clouds, spatial trajectory data, and remote sensing images. The MLS point clouds used were obtained, by a RIEGL VMX-450 MLS system, from the International Conference and Exhibition Center (ICEC) area and a part of the Coastal Ring Road (CRR), Xiamen, China. The ICEC contains a complicated road environment and structure; the CRR has a lot of vegetation and busy traffic, which leads to incomplete boundaries extracted, making both urban datasets appropriate for road boundary recovery evaluation. From January to May 2018, spatial trajectory data (GPS trajectory data of taxis), was collected along five main roads in the ICEC. For the ICEC, remote sensing images, with a pixel resolution of 0.5 m, were obtained from Google Map.

5.2. Parameter sensitivity analysis

In the proposed method, the configurations of the grid spacing $R_{\epsilon 1}$ and $R_{\epsilon 2}$ of a projection image are important factors for road boundary recovery performance. To find an optimal grid spacing $R_{\epsilon 1}$, we tested the performance of the configuration of the parameter on the erroneous boundary removal results. Also, to find an optimal grid spacing $R_{\epsilon 2}$, we tested the performance of the configuration of the parameter on the boundary completion results. We tested the following ten grid spacing: 0.2–1.1 m, at intervals of 0.1 m. The erroneous boundary removal and boundary completion results were both obtained using completeness-grid spacing, correctness-grid spacing, and quality-grid spacing (see Fig. 8). Three metrics used for evaluation are defined as follows:

$$\text{Completeness:} \quad cp = \frac{TP}{L_{gt}} \tag{9}$$

$$\text{Correctness:} \quad cr = \frac{TP}{L_{fc}} \tag{10}$$

$$\text{Quality:} \quad q = \frac{TP}{L_{fc} + FN} = \frac{TP}{TP + FP + FN} \tag{11}$$

whereTP is the length of the correct final boundaries, FP is the length of the boundaries that do not exist in the ground truth boundaries, and FN is the length of the ground truth boundaries that do not exist in the final complete boundaries.L_{gt} is the total length of the ground truth road boundaries, and L_{fc} is the total length of the final complete road boundaries.

As shown in the first row of Fig. 8, different grid spacing $R_{\epsilon 1}$ achieved different erroneous boundary removal performance. The first row of Fig. 8(a) shows that the three metrics are all better than the others when grid spacing $R_{\epsilon 1}$ is equal to 0.2 m. Though the time cost is highest, it is still short and acceptable. Thus, we set $R_{\epsilon 1}$ at 0.2 m. The second row of Fig. 8(a) shows the completeness of the boundaries increases slightly at first and then decreases with an increase in grid spacing $R_{\epsilon 2}$. This is because the road boundary gap is smaller if the grid spacing is larger. The completion model performs better on smaller gaps. However, if the grid spacing is set too large, the acquired low-quality boundary line image causes the gap detection and completion to fail. As grid spacing increases, correctness decreases. This is because larger grid spacing means the lines in an image are thicker and rougher, resulting in loss of boundary detail. For some gaps in curves with small curvature, it is easy to directly draw straight lines for these gaps. In addition, in this case, wrong line connections occur between some inner and outer boundaries that are in close proximity. As shown in the

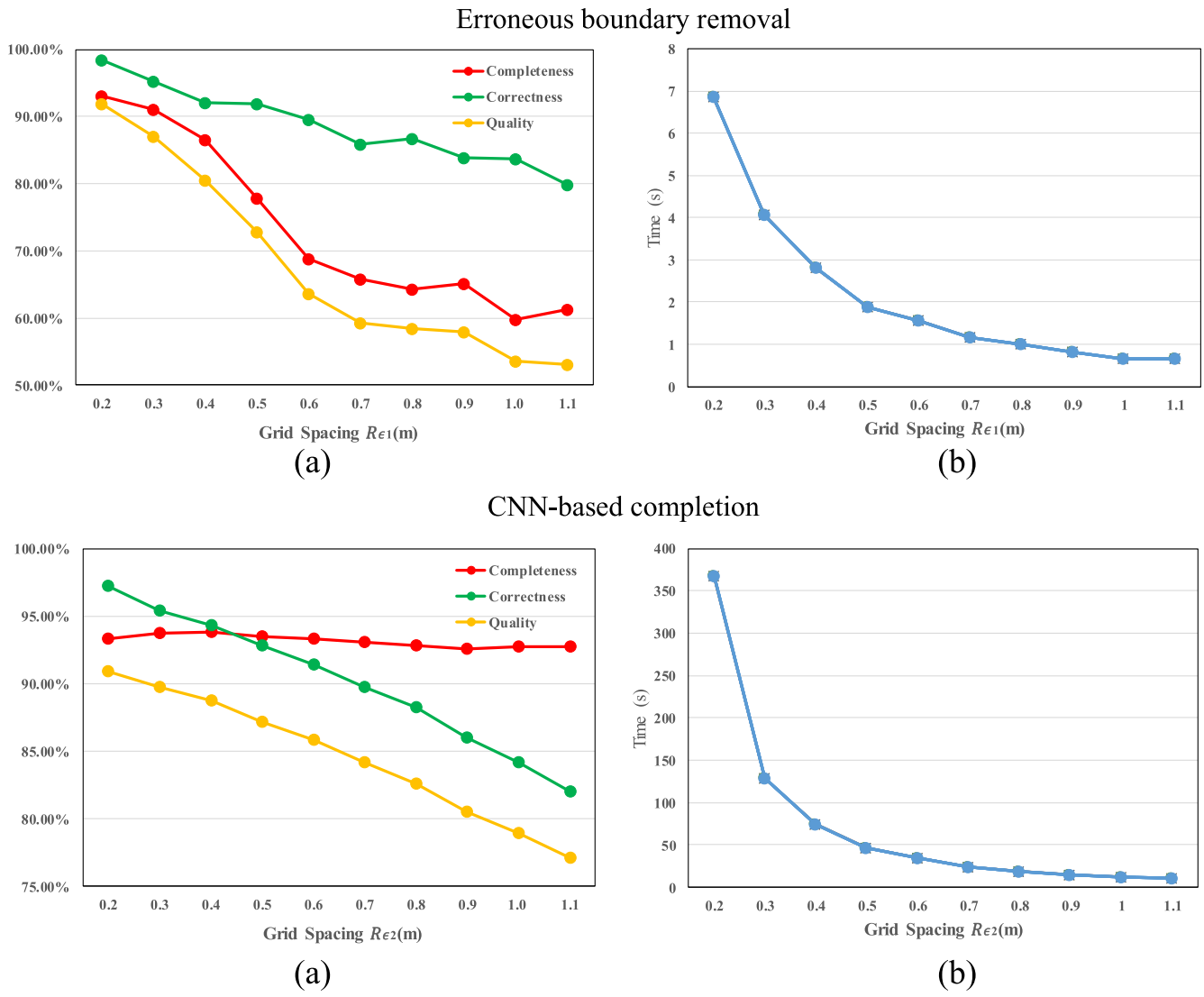

Fig. 8. Illustrations of the performance of erroneous boundary removal and CNN-based completion under different grid spacing: (a) Completeness, correctness, and quality results. (b) Time cost.

second row of Fig. 8(a), when grid spacing $R_{\epsilon 2}$ is equal to 0.2 m and 0.3 m, completion quality is almost the same. However, the time cost for model completion is much less when $R_{\epsilon 2}$ is set at 0.3 m, rather than at 0.2 m. Thus, considering the trade-off between boundary quality and time efficiency, we set $R_{\epsilon 2}$ at 0.3 m. In practice, to achieve optimal performance, we first project the extracted road boundary point clouds onto the image with grid spacing 0.2 m and perform erroneous boundary removal. Then, the correct road boundary point clouds are restored from the 3D points stored in each image segmented pixel and re-projected onto the image with grid spacing 0.3 m for CNN-based completion.

5.3. Road boundary completion results

5.3.1. Quantitative evaluation

To quantitatively evaluate our method, we first manually extracted ground truth road boundary points on two test datasets. For occluded road boundaries, according to the actual road situation, points were manually added between two endpoints of the occluded road boundaries. To verify the robustness of our method for 3D boundary completion with different incompleteness, road boundaries with different degrees of completeness were derived using several road boundary extraction methods, including projection-based (Serna and Marcotegui, 2013), characteristic-based (Fang et al., 2015) and supervolex-based (Zai et al., 2017). The extraction results are shown in Fig. 9. We compared the complete road boundaries with the ground truth boundaries and adopted the previously mentioned three metrics: completeness, correctness, and quality.

Part of the road boundary completion results in the ICEC and CRR are shown in Figs. 10 and 11, respectively. Enlarged for visualization are several parts of typical road scenes: road intersections in Fig. 10(a) and (b); road curves in Figs. 10(c), (d) and 11(b)–(d); and straight roads in Figs. 10(e), (f), and 11(a). Table 3 shows the meta completion results of the evaluation metrics based on three initial road boundaries with different degrees of completeness extracted by three methods. Table 4 gives corresponding quantitative evaluation results. For the proposed completion method achieved using the two datasets (ICEC and CRR), based on boundaries from the method of Zai et al. (2017), the completeness, correctness, and quality achieved on ICEC and CRR are as follows: completeness: 91.34% and 92.14%, respectively; correctness: 89.87% and 95.91%, respectively; quality: 82.81% and 88.65%, respectively. Table 4 indicates that the erroneous boundary removal step positively contributed to the boundary completion results. It also indicates that the quality of the initial boundaries has a strong effect on the completion results. The lower completeness of the initial boundaries indicates a more serious loss in the structure of the boundaries,

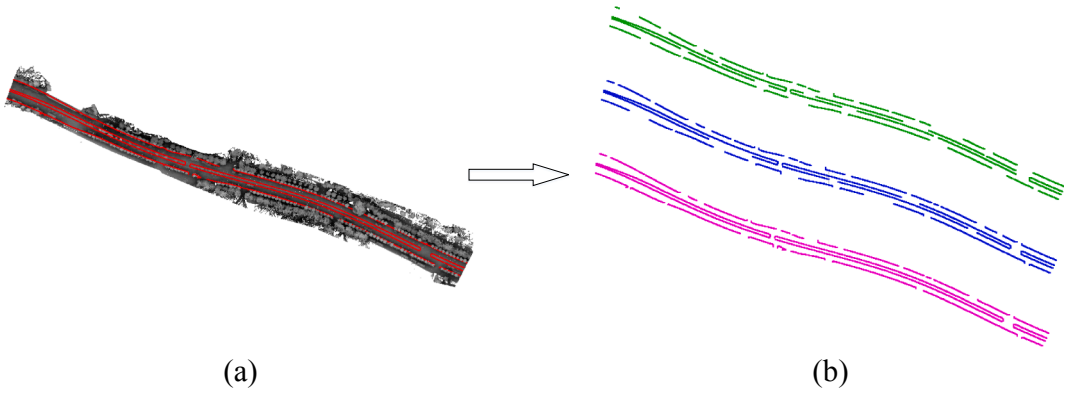

Fig. 9. The extraction results of a part of the road boundary in CRR dataset. (a) Raw point clouds with ground truth boundaries (in red). (b) Road boundaries obtained using the methods of Serna and Marcotegui (2013) (green), Fang et al. (2015) (blue), and Zai et al. (2017) (pink), respectively. From top to bottom, the completeness is 58.33%, 70.23%, and 91.80%, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

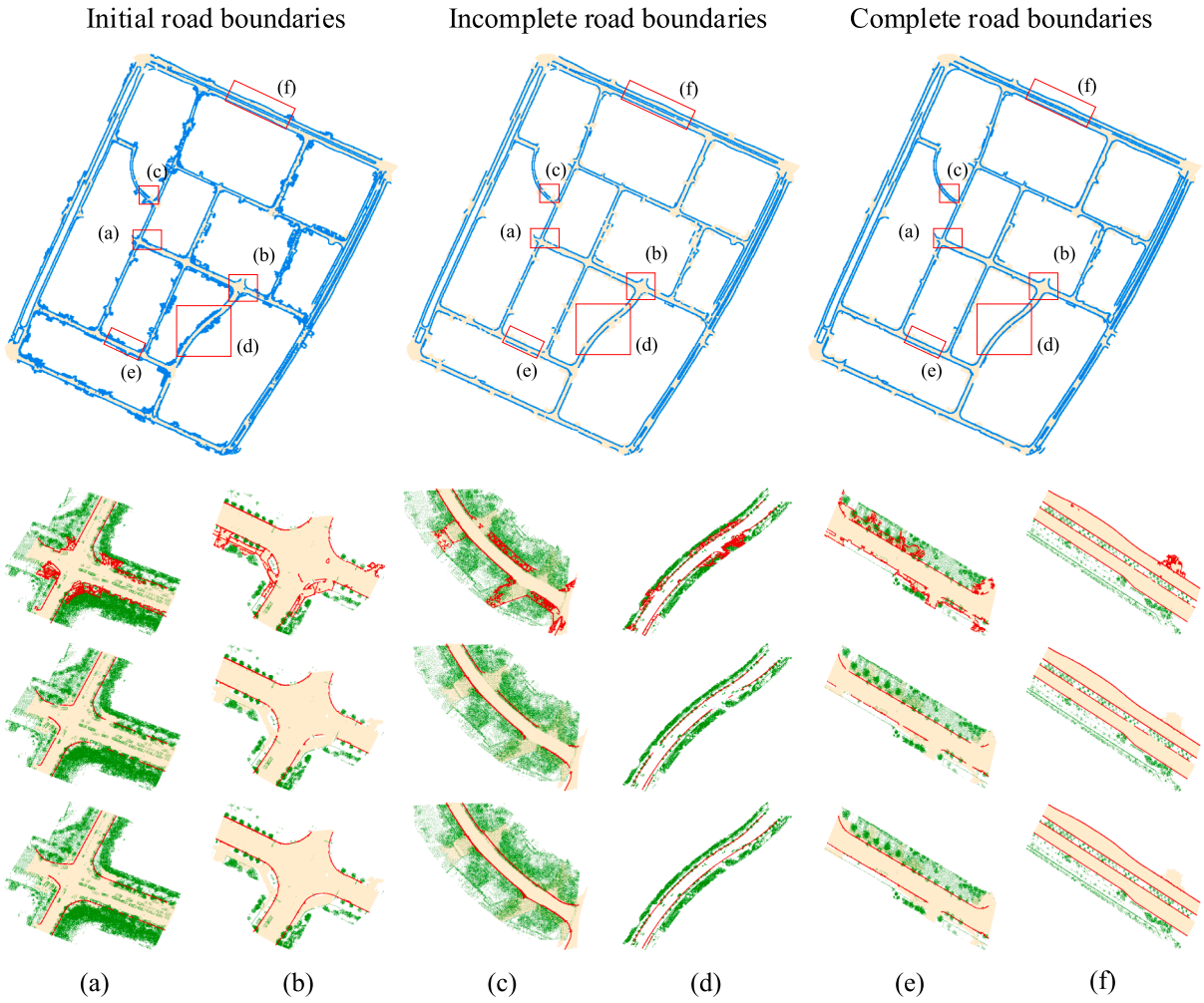

Fig. 10. Road boundary completion examples in the ICEC. (a), (b) Road intersection. (c), (d) Road curves. (e), (f) Straight road. The second row of Figs. (a–f) is the extracted road boundary parts by the method of Zai et al. (2017), the third row is road boundary parts after erroneous boundary removal, and the last row is the completion results.

resulting in an increase of uncertain factors for boundary completion. Thus, when the initial boundary completeness is lower, the final length of the boundaries that do not exist in the ground truth boundaries (*FP*) and the length of the ground truth boundaries that do not exist in the final complete boundaries (*FN*) are longer. However, even based on road boundaries with a degree of completeness lower than 60%, our completion method still obtains satisfactory *TP* and substantially improve completeness. Thus, our proposed method performs well in 3D road boundary completion and is robust to different degrees of road boundary completeness.

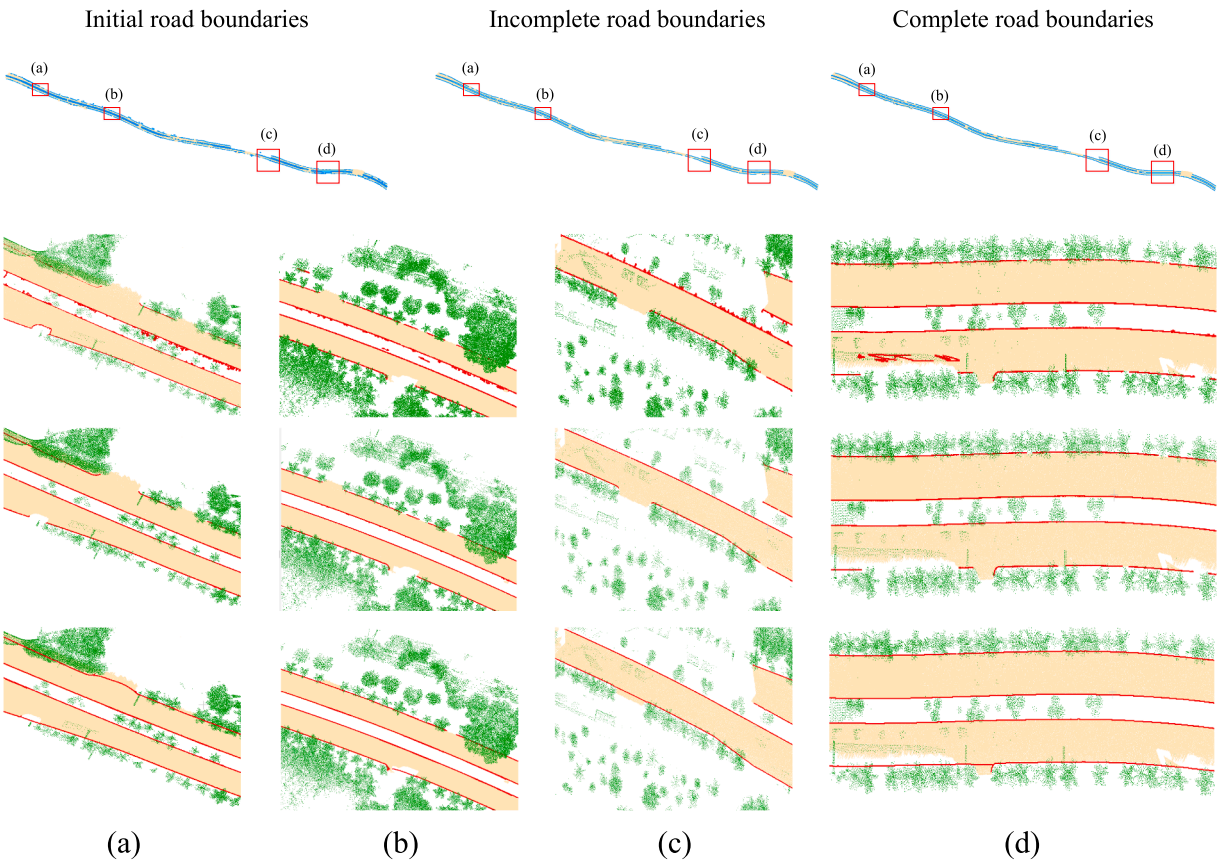

Fig. 11. Road boundary completion examples in the CRR. (a) Straight road. (b), (c), (d) Road curves. The second row of Figs. (a–f) is the extracted road boundary parts by the method of Zai et al. (2017), the third row is road boundary parts after erroneous boundary removal, and the last row is the completion results.

In addition, we used different allowable error distances to obtain different evaluation results with ICEC and CRR datasets. Error distance is defined as the average distance between added points and their nearest points on ground truth boundaries. The error distance used in the results shown in Tables 3 and 4 is 0.1 m. As shown in Table 5, with an error distance increase in the two datasets, the three metrics increase only slightly, indicating the newly added points coincide well with the ground truth boundaries.

5.3.2. Test on lower quality dataset

To further test the feasibility of our method in point clouds with different quality, we tested the proposed method using the KITTI dataset (Geiger et al., 2012). Using a series of single frames and ground truth poses from the KITTI-odometry benchmark, which was collected by a Velodyne laser scanner, we reconstructed point clouds for a road section. A part of the road point clouds that include curbs was used for testing. The accuracy of the acquired KITTI point clouds is about 5–10 cm, which is much smaller than the millimeter-level accuracy of the MLS point clouds. Considering the lower density of the KITTI point clouds, to correctly estimate the tangent plane, at least 1000 neighboring points for each point were used, causing a heavy cost in time for this step. To reduce the time cost, the original point clouds were first down-sampled by a VoxelGrid filter (Rusu and Cousins, 2011) with a size of 0.1 m. Then, according to a practical test, the parameters in the experiment were modified as follows: the number of nearest points (K) from 15 to 40; angle threshold to determine if two vectors are parallel (θ) from 22.5° to 8.0°; outlier clustering threshold (N) from 50 to 15. The other parameters remained the same as given in the paper of Zai et al. (2017). Fig. 12 shows the initial road extraction, completion, and ground truth boundaries for this KITTI test dataset. As shown in Table 4, the completeness, correctness, and quality in the KITTI dataset are 89.90%, 83.91%, and 76.68%, respectively. Remarkably, completeness increases from 52.65% to 89.90%. Thus, our boundary completion method also achieves satisfying results for the KITTI dataset.

5.3.3. Computational efficiency

Boundary extraction and completion in 3D form in our proposed framework, coded with C++, were run on a personal computer configured with an Intel(R) Core (TM) i7-6700 CPU 3.4 GHz and a RAM of 16 GB. Erroneous boundary removal and 2D boundary completion were run on a Linux PC with an Intel Core (TM) i5-4460 CPU and two NVIDIA Titan Z GPUs with 12 GB memory. The initial road boundary extraction was discussed in detail by Zai et al. (2017). Therefore, we focused only on the time to compute erroneous boundary removal and boundary completion. As shown in Table 6, 2D boundary completion based on image inpainting model, using the CPU, requires most of the total processing time. Here, the computational cost depends only on the size of the input image. The sizes of the three boundary images (ICEC, CRR, and KITTI) are 4288 × 4080, 1984 × 6480, and 6112 × 3920, respectively. Thus, the computational cost of gap completion in KITTI is maximum. However, under the acceleration of the GPU, this step can be 60 times faster. Total time costs in datasets ICEC, CRR, and KITTI are about 192.52 s, 107.87 s, and 188.93 s, respectively.

5.4. Road boundary refinement results

To refine a road boundary after completion, the road centerline, first extracted from the taxi GPS trajectories, was improved by remote sensing images. An example of road centerline extraction is shown in Fig. 13. The following items: (1) Taxi GPS trajectory points clustered to each separated road lane, (2) fitted road centerlines, (3) remote sensing image of the roads, (4) extracted roads from images (in black), and (5)

Table 3
Road boundary completion results with different degree of incompleteness in three datasets (ICEC/CRR/KITTI).

Dataset	L_{gt}(m)	Boundary extraction method	Extraction results				Extraction + erroneous boundary removal				Extraction + erroneous boundary removal + CNN-based completion			
			L_{fc}(m)	*TP*(m)	*FP*(m)	*FN*(m)	L_{fc}(m)	*TP*(m)	*FP*(m)	*FN*(m)	L_{fc}(m)	*TP*(m)	*FP*(m)	*FN*(m)
ICEC	20003.96	Projection-based (Serna and Marcotegui, 2013)	13060.61	11390.16	1670.45	8613.80	11330.55	10978.17	352.38	9025.79	19797.46	17635.49	2161.97	2368.47
		Characteristic-based (Fang et al., 2015)	18282.83	15015.69	3267.14	4988.27	15285.63	14522.87	762.76	5481.09	20130.66	17797.52	2333.14	2206.44
		Supervolex-based (Zai et al., 2017)	22038.56	17333.84	4704.72	2670.12	17539.70	16566.25	973.45	3437.71	20330.61	18271.12	2059.49	1732.84
CRR	6941.69	Projection-based (Serna and Marcotegui, 2013)	3828.78	3622.41	206.37	3319.28	3611.53	3572.89	38.64	3368.8	6568.59	5890.71	677.88	1050.98
		Characteristic-based (Fang et al., 2015)	5113.31	4935.87	177.44	2005.82	4844.75	4800.18	44.57	2141.51	6777.14	6175.32	601.82	766.37
		Supervolex-based (Zai et al., 2017)	6612.43	6287.23	325.20	654.46	6045.89	5946.95	98.94	994.74	6668.83	6395.89	272.94	545.80
KITTI	4140.70	Supervolex-based (Zai et al., 2017)	2530.68	2180.14	1960.56	350.54	2122.69	2015.85	106.84	514.83	4436.47	3722.44	714.03	418.26

Table 4
Quantitative evaluation results of road boundary completion in three datasets (ICEC/CRR/KITTI).

Dataset	Method for initial boundary extraction	Extraction results			Extraction + erroneous boundary removal			Extraction + erroneous boundary removal + CNN-based completion		
		Completeness (%)	Correctness (%)	Quality (%)	Completeness (%)	Correctness (%)	Quality (%)	Completeness (%)	Correctness (%)	Quality (%)
ICEC	Projection-based (Serna and Marcotegui, 2013)	56.94	87.21	52.55	54.88	96.89	53.93	88.16	89.08	79.56
	Characteristic-based (Fang et al., 2015)	75.06	82.13	64.53	72.60	95.01	69.93	88.97	88.41	79.68
	Supervolex-based (Zai et al., 2017)	86.65	78.65	70.15	82.81	94.45	78.97	91.34	89.87	82.81
CRR	Projection-based (Serna and Marcotegui, 2013)	52.18	94.61	50.68	51.47	98.93	51.19	84.86	89.68	77.31
	Characteristic-based (Fang et al., 2015)	71.10	96.53	69.33	69.15	99.08	68.71	88.96	91.12	81.86
	Supervolex-based (Zai et al., 2017)	90.57	95.08	86.52	85.67	98.36	84.47	92.14	95.91	88.65
KITTI	Supervolex-based (Zai et al., 2017)	52.65	86.15	48.54	48.68	94.97	76.43	89.90	83.91	76.68

Table 5
Quantitative evaluation results of different error distances in ICEC and CRR dataset.

Dataset	Error distance (m)	Completeness (%)	Correctness (%)	Quality (%)
ICEC	0.1	91.34	89.87	82.81
	0.2	91.89	90.11	83.47
	0.3	92.26	90.29	83.93
	0.4	92.40	90.58	84.30
	0.5	92.52	90.79	84.58
CRR	0.1	92.14	95.91	88.65
	0.2	92.33	96.09	88.99
	0.3	92.75	96.32	89.58
	0.4	92.90	96.47	89.85
	0.5	93.11	96.68	90.22

road centerlines by combining (2) and (4) are given in Fig. 13(a), (b), (c), (d), and (e), respectively.

Then, according to the two imperfect completion results summarized above, we built 2000 training samples based on road boundaries and road centerlines. Each imperfect completion category has 500 correct samples and 500 imperfect samples. Some examples of training samples are shown in Fig. 14. For the expectation of the network training input, we combined target sample images with input sample images as side-by-side images. During the training process, the batch size and epoch of the cGAN model were set at 2 and 300, respectively.

To test the performance of the road boundary refinement, 500 negative samples (250 samples for each category) were used for test. We compared the generated refinement lines with the target lines and calculated the distance between each point of the refinement lines and the nearest point in the target lines. Average error distance was used to evaluate the performance of the proposed cGAN-based boundary line refinement method. Time cost and average error distance for each imperfect completion category with 250 testing samples are given in Table 7. Time costs for the refinement of gaps without completion and irregular completion structures are about 7.23 s and 3.61 s, respectively. The refinement modules for gaps without completion and irregular completion structures achieved results with an average error distance of 0.48 m and 0.56 m, respectively. Remarkably, the average error distance of the refinement modules, with irregular completion structures, decreases from the original 3.44 m to 0.56 m. The results

Table 6
Time cost of each processing step at the road boundary completion stage.

Dataset	Steps			Total time cost (s)
	Erroneous boundary removal (s)	2D boundary completion (s)	Completion in 3D form (s)	
ICEC	29.85	129.11	33.56	192.52
CRR	5.18	94.91	7.78	107.87
KITTI	2.72	181.19	5.02	188.93

show that the refinement for straight lines is very accurate. The main errors occur in the curves, which are difficult to obtain the ground truth. A small difference in the curvature between the refinement and target curves causes a large error distance. In addition, error distances were divided into many intervals and the proportion of each interval is shown in Fig. 15. In general, the average error distance of the testing results from the two modules with a centerline is less than 0.6 m. The greatest number of testing samples has an error distance within 0.2 m, which indicates the effectiveness of the proposed road boundary refinement.

To further demonstrate the effect of the road centerline on road boundary refinement, we also trained a network using training samples without a road centerline. Fig. 16 shows some comparative testing results for each category of imperfect completion. Without centerline guidance, for some curves with gaps, it is difficult to determine the way of connection, resulting in random line connections (pink lines in Fig. 16(a)). With centerline guidance, the refinement model modifies curves with gaps or with irregular structures based on the rough curvature of the centerline. The average error distance listed in Table 7 indicates that the refinement model with a centerline achieves a smaller error distance and provides better refinement, especially for gaps without completion. The average error distance of the refinement module without a centerline for gaps without completion has the greatest error distance (2.87 m), and the largest proportion of error distances greater than 1.0 m, which are much worse than the error distances with a centerline. Results indicate that the road centerlines obtained from taxi GPS trajectories effectively assist road boundary refinement.

To assess each step of road boundary recovery, including erroneous

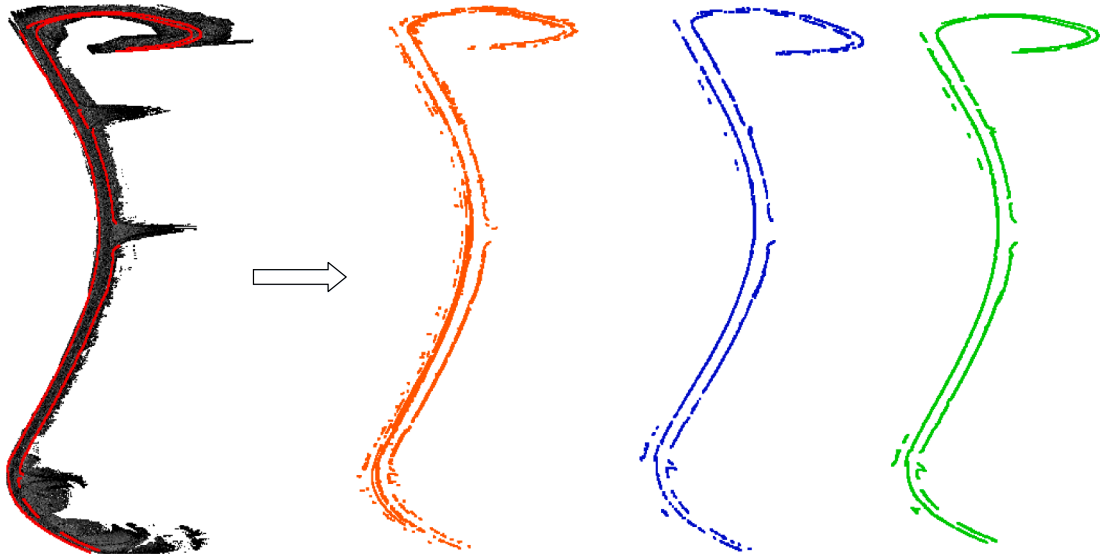

Fig. 12. From left to right, raw KITTI point clouds with ground truth (in red), extracted road boundaries by the method of Zai et al. (2017), road boundaries after erroneous boundary removal, and completed road boundaries, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

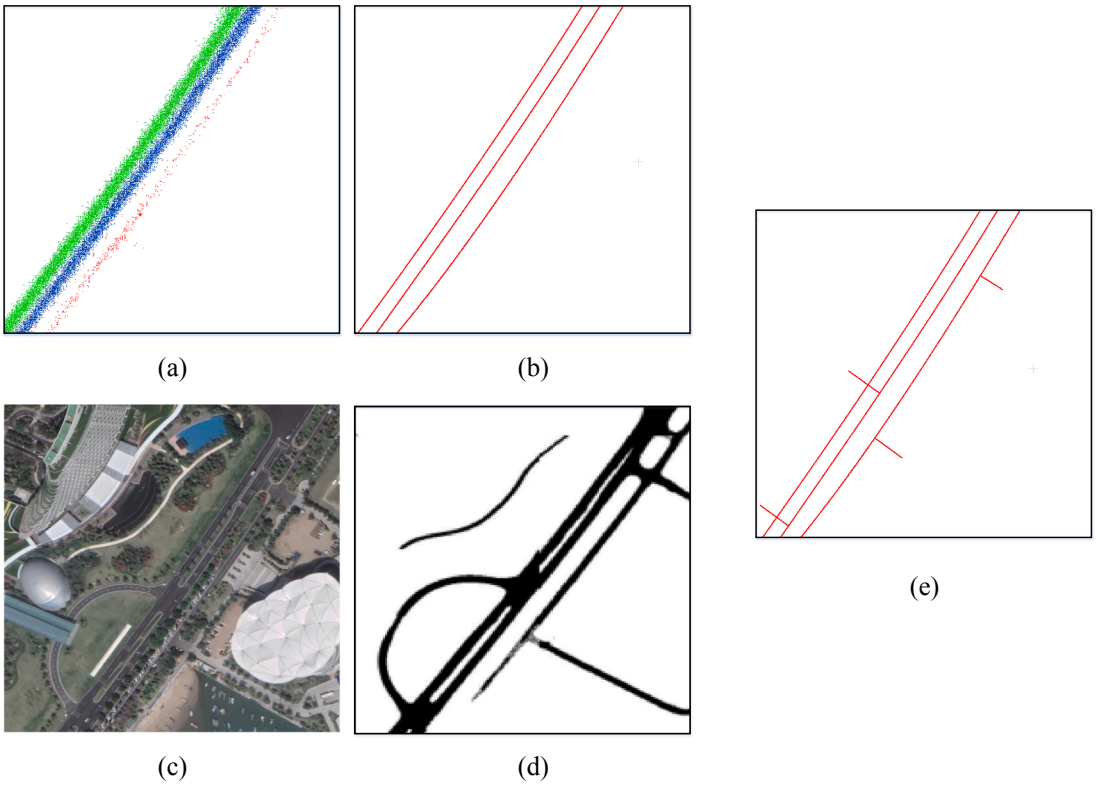

Fig. 13. Road centerline extraction. (a) Taxi GPS trajectory points. (b) Road centerlines obtained from taxi GPS trajectory points. (c) Remote sensing image. (d) Roads extracted from image (c). (e) Final road centerlines merged from (b) and (d).

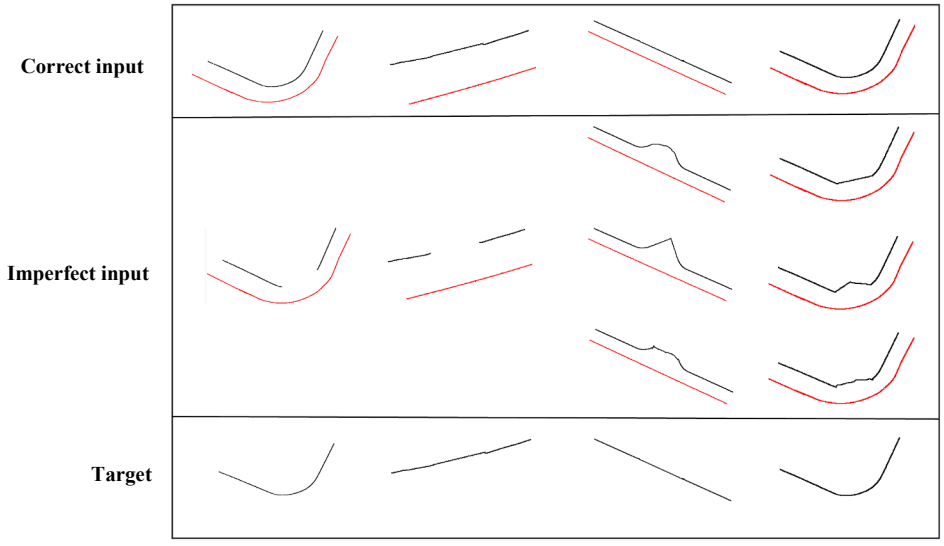

Fig. 14. Training sample examples of gaps without completion (first two columns) and irregular completion structures (last two columns). Black lines are road boundaries and red lines are road centerlines. From top to bottom are correct input samples, imperfect input samples, and target samples. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

boundary removal, CNN-based boundary completion and cGAN-based boundary refinement (with/without a centerline), a section of the ICEC road, which has corresponding taxi GPS trajectory data and remote sensing images to obtain road centerlines, was selected for testing. Table 8 indicates that the erroneous boundary removal step improves correctness with only a small decrease in completeness, and CNN-based completion significantly improves completeness and quality, but decreases correctness. The two steps were taken jointly to achieve better completion quality. Also, boundary refinement without a centerline slightly affects the results, but boundary refinement with a centerline

Table 7
Testing sample size, time cost, and average error distance.

Problem category	Time cost (s)	Original error distance (m)	Average error distance (with centerline) (m)	Average error distance (without centerline) (m)
Gaps without completion	3.62 + 3.61	–	0.48	2.87
Irregular completion structures	3.61	3.44	0.56	1.11

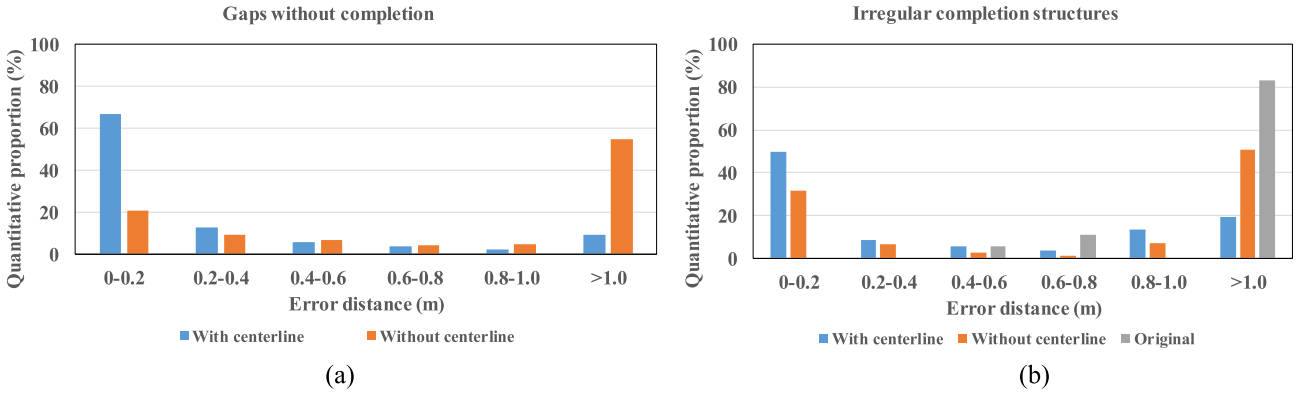

Fig. 15. Error distance according to the proportion of testing samples.

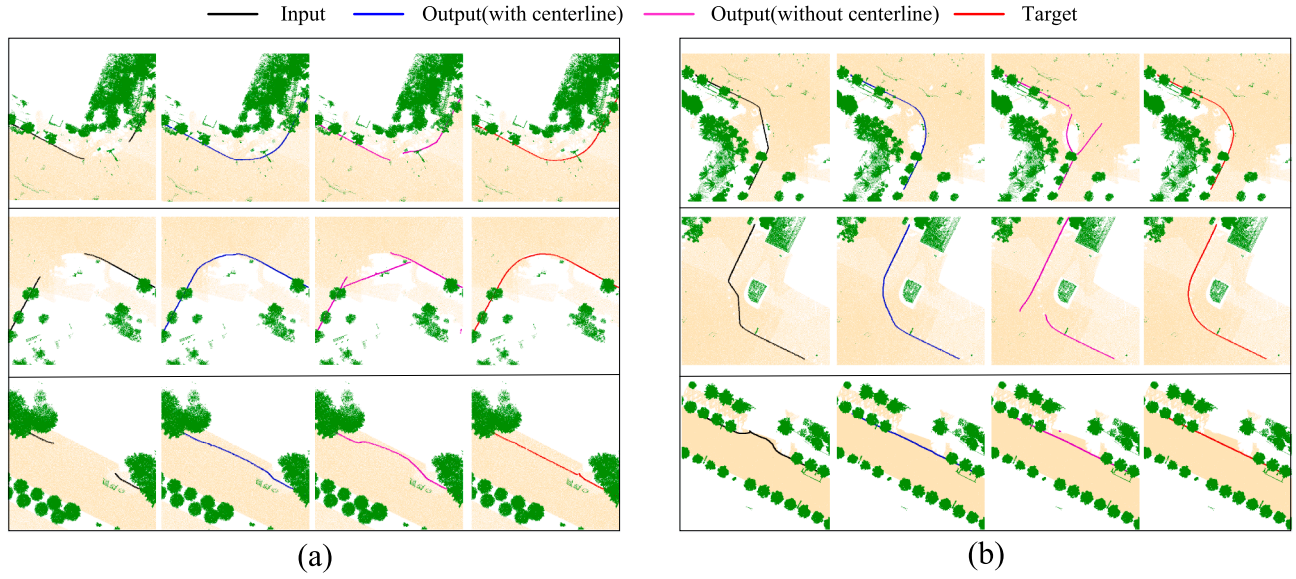

Fig. 16. Examples of testing results. (a) Gaps without completion. (b) Irregular completion structures.

Table 8
Quantitative evaluation results of each boundary recovery step on a section of ICEC dataset.

Steps	Completeness (%)	Correctness (%)	Quality (%)
Extraction by the method of Zai et al. (2017)	80.90	91.49	75.24
Extraction + erroneous boundary removal	76.20	97.93	74.99
Extraction + erroneous boundary removal + CNN-based completion	91.39	86.01	79.56
Extraction + erroneous boundary removal + CNN-based completion + cGAN-based refinement (without centerline)	92.46	86.59	80.88
Extraction + erroneous boundary removal + CNN-based completion + cGAN-based refinement (with centerline)	93.11	88.76	83.29

Table 9
Horizontal alignment parameter results for sample roads.

Sample number (Curve)	$\Delta\alpha$(degrees-minutes-seconds)			Δs(m)			$\overline{C}$			Sample number (Road)	d_r(m)		
	Ground Truth	Result	Error	Ground Truth	Result	Error	Ground Truth	Result	Error		Ground Truth	Result	Error
1	58°24′46″	59°04′32″	39′46″	166.15	166.01	−0.13	0.0061	0.0062	0.0001	1 (Curve)	7.13	7.05	−0.08
2	65°30′38″	66°18′39″	48′1″	214.01	213.90	−0.12	0.0053	0.0054	0.0001	2 (Curve)	25.81	25.85	0.04
3	75°45′06″	75°34′05″	−11′1″	26.69	26.80	0.12	0.0495	0.0492	0.0003	3 (Straight)	8.04	8.01	−0.03
4	83°59′24″	83°57′25″	−1′59″	28.77	28.70	−0.07	0.0510	0.0511	0.0001	4 (Straight)	15.04	14.98	−0.06
5	87°30′49″	86°44′24″	46′24″	31.03	30.92	−0.11	0.0492	0.0490	0.0003	5 (Straight)	14.98	14.91	−0.07
Accuracy	± 48′01″			± 0.13			± 0.0003				± 0.08		

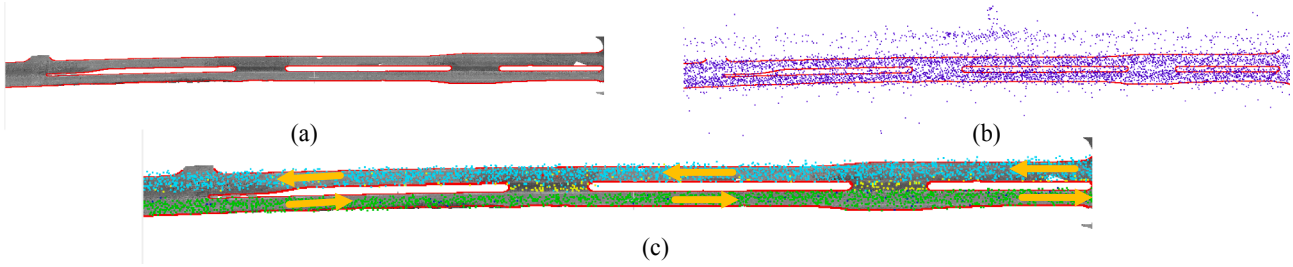

Fig. 17. (a) Road surface point clouds with boundary in red. (b) Scattered taxi GPS trajectory points around road boundary. (c) Taxi GPS trajectory points within road boundary after road matching. Yellow arrows represent the direction of the traffic flow. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

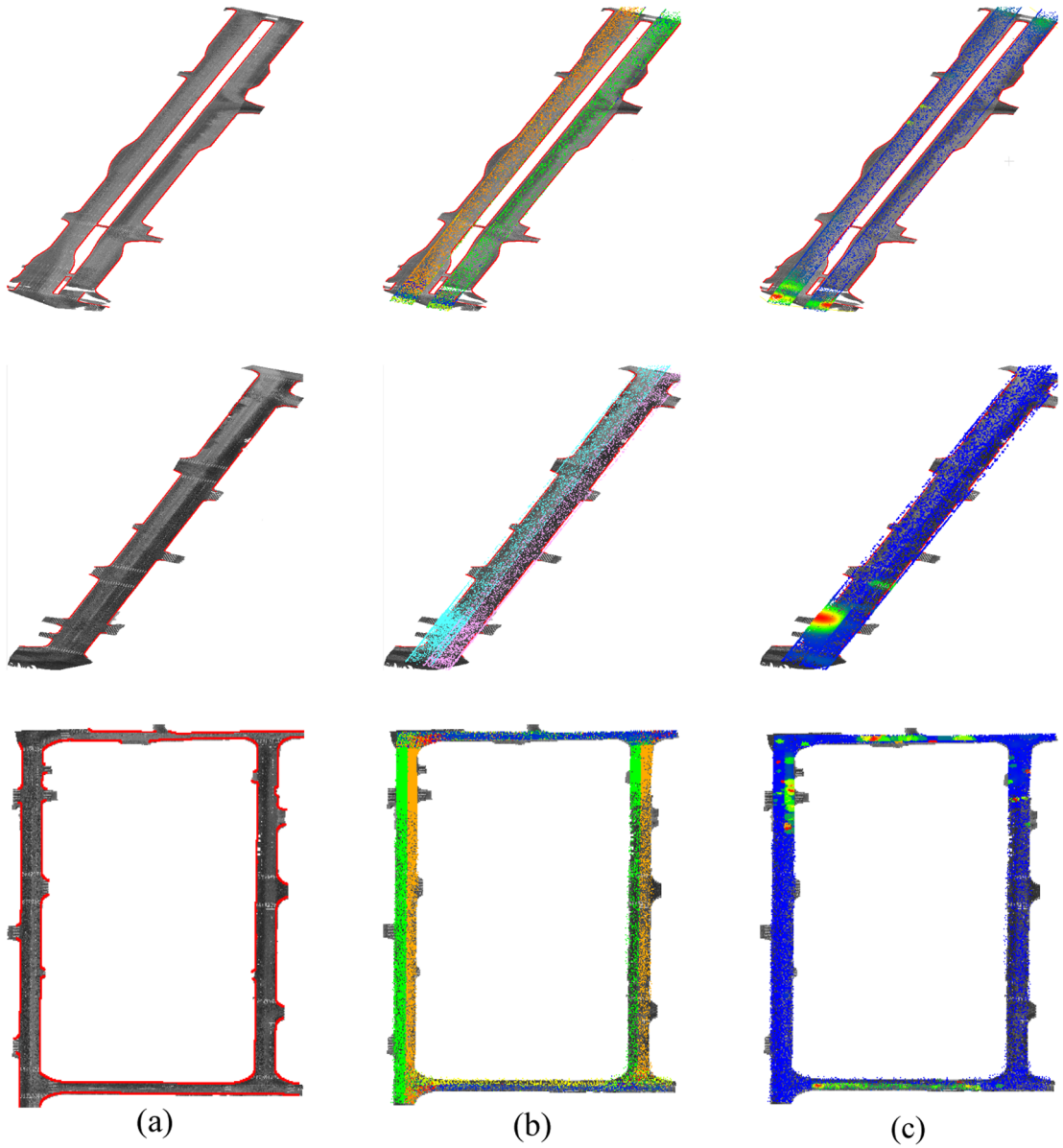

Fig. 18. Three road scenes with matched GPS trajectory points. The first row is a two-way separated road, the second row is a two-way road without isolation belt, and the last row is ring road with seven lanes. (a) Road surface point clouds with boundary in red. (b) GPS trajectory point flow of each lane (in different colors). (c) Traffic density result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

shows increased performance on all three metrics. With these steps combined, both final completeness and quality achieve an increase of more than 8% at the cost of a decrease in correctness of nearly 3%.

5.5. Road characteristics analysis

5.5.1. Inherent road characteristic evaluation

To quantitatively evaluate the accuracy of the proposed method for

inspecting inherent road characteristics, a Leica total station TS 15i-1 and a Leica RTK GS15, which can provide millimeter level measurements, were used to collect on-site measurement data as the ground truth. Given the lack of intuitive vertical road curves and most of our road data are almost horizontal, we collected only horizontal alignment parameters to evaluate. For illustration, five roads, including inner and outer boundary, were selected for calculating road width. Detailed results are shown in Table 9. Calculating $\Delta\alpha$, Δs, $\overline{C}$, and d_r on these road parts, our proposed method achieved an accuracy of ± 48′01″, ± 0.15 m, ± 0.0003, and ± 0.08 m, respectively.

5.5.2. Dynamic road characteristic evaluation

To explore the dynamic road information extraction, a typical road with two lanes was selected. Here, we only visualized road matching result of the taxi GPS trajectory points and the direction of the traffic flow. The distribution of taxi GPS trajectory points on the road with two lanes separated by an isolation belt, located at the lower right of ICEC (see Fig. 10), are shown for before and after road matching in Fig. 17(b) and (c), respectively. Offset taxi GPS trajectory points, inside the isolation belt and outside the road boundary, were matched within the road boundary. Different colors represent different direction values. The average direction values of points in green and light blue are about 31 and 214, respectively. According to these values, the direction of the taxi flow, represented by the light-yellow arrows in Fig. 17(c), can also be determined. The results indicate that our proposed method performs well in matching taxi GPS recorded trajectory points to the correct road lane of the underlying road boundary.

In addition, shown in Fig. 18 are the following other three road scenes with matched taxi GPS trajectory points: a two-way separated road located atop the ICEC (see Fig. 10), a two-way road without an isolation belt, and a ring road with seven traffic lanes located at the lower left of the ICEC (see Fig. 10). Fig. 18(b) shows a good division of traffic flow in the two-way lanes (pink and blue). The traffic density, based on the matched taxi GPS trajectory points, is shown in Fig. 18(c). The red points in Fig. 18(c) represent the highest traffic density areas, which are usually on road intersections, turnings or the openings of narrow trails. This indicates the feasibility and great potential of combining dynamic road information with static road structures to achieve dynamic visualization of a road boundary.

6. Conclusions and future work

In this paper, we have presented a novel framework for 3D road boundary recovery using MLS point clouds, spatial trajectory data, and remote sensing images. The proposed framework successfully recovers the road boundaries by integrating an erroneous boundary removal, a CNN-based completion model, and a cGAN-based boundary refinement model guided by road centerlines obtained from taxi GPS trajectory points and remote sensing images. Using recovered road boundary point clouds and taxi GPS recorded trajectory points associated with the proper road, inherent and dynamic road information can be extracted, accordingly. The ability to recover the 3D road boundary that are in poor conditions (e.g., incomplete raw data and uncertainties of gaps) is an impressive achievement, as it has been tested on point clouds datasets with different qualities. For two MLS datasets (ICEC and CRR) and a Velodyne dataset (KITTI), the completeness, correctness, and quality achieved on ICEC, CRR, and KITTI are as follows: completeness: 91.34%, 92.14%, and 89.90%, respectively; correctness: 89.87%, 95.91%, and 83.91%, respectively; quality: 82.81%, 88.65%, and 76.68%, respectively. The trained models are made available at https://github.com/youcb/Road_boundary_recovery. Future work will focus on designing end-to-end algorithms for 3D road boundary recovery and exploring the integration and visualization of a road network using multi-source data.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grants No. 61771413) and the Fundamental Research Funds for the Central Universities (Grant No. 20720170047).

References

Cheng, G., Wang, Y., Xu, S., Wang, H., Xiang, S., Pan, C., 2017a. Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural network. IEEE Trans. Geosci. Remote Sens. 55 (6), 3322–3337.
Cheng, M., Zhang, H., Wang, C., Li, J., 2017b. Extraction and classification of road markings using mobile laser scanning point clouds. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 10 (3), 1182–1196.
Dai, A., Qi, C.R., Nießner, M., 2017. Shape completion using 3D-encoder-predictor CNNs and shape synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6545–6554.
Fang, L., Yang, B., Chen, C., Fu, H., 2015. Extraction 3D road boundaries from mobile laser scanning point clouds. In: Proceedings of the IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services, pp. 162–165.
Gao, L., Shi, W., Lv, Z., Zhang, H., 2013. Road network extraction based on airborne LiDAR and high resolution remote sensing imagery. Rem. Sens. Technol. Appl. 28 (4), 562–568.
Gao, X., Sun, X., Yan, M., Sun, H., Fu, K., Zhang, Y., Ge, Z., 2018. Road extraction from remote sensing images by multiple feature pyramid network. In: Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, pp. 6907–6910.
Geiger, A., Lenzm, P., Urtasun, R., 2012. Are we ready for autonomous driving? The KITTI vision benchmark suite. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3354–3361.
Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al., 2014. Generative adversarial networks. Adv. Neural Inform. Process. Syst. 2672–2680.
Hamilton, A., Wang, H., Tanyer, A.M., Arayici, Y., Zhang, X., Song, Y., 2005. Urban information model for city planning. Electron. J. Inform. Technol. Constr. 10, 55–67.
Hedman, K., Stilla, U., Lisini, G., Gamba, P., 2010. Road network extraction in VHR SAR images of urban and suburban areas by means of class-aided feature-level fusion. IEEE Trans. Geosci. Remote Sens. 48 (3), 1294–1296.
Hu, J., Razdan, A., Femiani, J.C., Cui, M., Wonka, P., 2007. Road network extraction and intersection detection from aerial images by tracking road footprints. IEEE Trans. Geosci. Remote Sens. 45 (12), 4144–4157.
Hu, X., Li, Y., Shan, J., Zhang, J., Zhang, Y., 2014. Road centerline extraction in complex urban scenes from lidar data based on multiple features. IEEE Trans. Geosci. Remote Sens. 52 (11), 7448–7456.
Hu, X., Tao, C.V., Hu, Y., 2004. Automatic road extraction from dense urban area by integrated processing of high resolution imagery and LIDAR data. In: Proceedings of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, pp. 320–324.
Hervieu, A., Soheilian, B., 2013. Road side detection and reconstruction using LiDAR sensor. In: Proceedings of the IEEE Intelligent Vehicles Symposium, pp. 1247–1252.
Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A., 2017. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5967–5976.
Karlaftis, M.G., Golias, I., 2002. Effects of road geometry and traffic volumes on rural roadway accident rates. Accid. Anal. Prevent. 34 (3), 357–365.
Li, C., Zhang, D., Tao, J., Wang, F., 2014. Information extraction and its application of urban road networks based on google images. Rem. Sens. Technol. Appl. 27 (1), 100–105.
Li, Y., Yong, B., Wu, H., An, R., Xu, H., 2015. Road detection from airborne lidar point clouds adaptive for variability of intensity data. Int. J. Light Electron Opt. 126 (23), 4292–4298.
Litescu, S.C., Viswanathan, V., Aydt, H., Knoll, A., 2016. The effect of information uncertainty in road transportation systems. J. Comput. Sci. 16, 170–176.
Miaou, S.P., Lum, H., 1993. Modeling vehicle accidents and highway geometric design relationships. Accid. Anal. Prevent. 25 (6), 689–709.
Mogelmose, A., Trivedi, M.M., Moeslund, T.B., 2012. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: perspectives and survey. IEEE Trans. Intell. Transp. Syst. 13 (4), 1484–1497.
Newson, P., Krumm, J., 2009. Hidden markov map matching through noise and sparseness. In: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems, pp. 336–343.
Ouyang, H., Liu, J., Liu, Y., Liao, Z., Chen, She, 2014. An extraction method of road network based on walking gps trajectories. Comput. Modern. 2, 124–128.
Peng, J., Gao, G., 2011. A method for main road extraction from airborne LiDAR data in urban area. In: Proceedings of 2011 International Conference on Electronics, Communications and Control, pp. 2425–2428.
Péteri, R., Ranchin, T., Scientist, C., 2006. Automatic road network extraction using collaborative linear and surface models. Proceedings of the International Conference on MAPPS/ASPRS.
Poz, A.P.D., Zanin, R.B., Vale, G.M.D., 2006. Automated extraction of road network from medium-and high-resolution images. Pattern Recogn. Image Anal. 16 (2), 239–248.
Raber, G.T., Jensen, J.R., Hodgson, M.E., Tullis, J.A., Davis, B.A., Berglund, J., 2007. Impact of lidar nominal post-spacing on DEM accuracy and flood zone delineation. Photogramm. Eng. Remote Sens. 73 (7), 793–804.

Reza, F.R., Keiichi, U., Guo, K., Kohichi, O., 2017. Road edge detection on 3D point cloud data using encoder-decoder convolutional network. In: Proceedings of the Conference on 2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing, pp. 95–100.
Rodriguez, A., Laio, A., 2014. Clustering by fast search and find of density peaks. Science 344 (6191), 1492–1496.
Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: convolutional networks for biomedical image segmentation. In: Proceedings of International Conference on Medical Image Computing and Computer-assisted Intervention, pp. 234–241.
Rusu, R.B., Cousins, S., 2011. 3D is here: point cloud library (PCL). Proceedings of the IEEE International Conference on Robotics and Automation.
Sasaki, K., Iizuka, S., Simo-Serra, E., Ishikawa, H., 2017. Joint gap detection and inpainting of line drawings. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5768–5776.
Schroedl, S., Wagstaff, K., Rogers, S., Langley, P., Wilson, C., 2004. Mining gps traces for map refinement. Data Min. Knowl. Disc. 9 (1), 59–87.
Serna, A., Marcotegui, B., 2013. Urban accessibility diagnosis from mobile laser scanning data. ISPRS J. Photogramm. Remote Sens. 84, 23–32.
Shackelford, A.K., Davis, C.H., 2003. Fully automated road network extraction from high-resolution satellite multispectral imagery. In: Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, pp. 461–463.
Stutz, D., Geiger, A., 2018. Learning 3D Shape completion from laser scan data with weak supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1955–1964.
Tupin, F., Maitre, H., Mangin, J.F., Nicolas, M., Pechersky, E., 1998. Detection of linear features in SAR images: application to road network extraction. IEEE Trans. Geosci. Rem. Sens. 36 (2), 434–453.
Viterbi, A.J., 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. Inf. Theory 13 (2), 260–269.
Vo, A.V., Truonghong, L., Laefer, D.F., 2015. Aerial laser scanning and imagery data fusion for road detection in city scale. In: Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, pp. 4177–4180.
Wang, C., Hou, S., Wen, C., Gong, Z., Li, Q., Sun, X., Li, J., 2018. Semantic line framework-based indoor building modeling using backpacked laser scanning point cloud. ISPRS J. Photogramm. Remote Sens. 143, 150–166.
Wegner, J.D., Montoya-Zegarra, J.A., Schindler, K., 2013. A higher-order CRF model for road network extraction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1698–1705.
Xie, X., Philips, W., Veelaert, P., Aghajan, H., 2014. Road network inference from GPS traces using DTW algorithm. In: Proceedings of the IEEE Conference on Intelligent Transportation Systems, pp. 906–911.
Xu, S., Wang, R., Zheng, H., 2016. Road curb extraction from mobile lidar point clouds. IEEE Trans. Geosci. Remote Sens. 55 (2), 996–1009.
Yeo, Y., Xiao, X., Zhang, X., 2016. Rural scene parsing and road boundary estimation by fusion of lidar pointcloud and EO images. In: Proceedings of the International Conference on Information Fusion, pp. 1760–1767.
Yu, Y., Li, J., Wen, C., Guan, H., Luo, H., Wang, C., 2016. Bag-of-visual-phrases and hierarchical deep models for traffic sign detection and recognition in mobile laser scanning data. ISPRS J. Photogramm. Remote Sens. 113, 106–123.
Zai, D., Li, J., Guo, Y., Cheng, M., Lin, Y., Luo, H., Wang, C., 2017. 3-D road boundary extraction from mobile laser scanning data via supervoxels and graph cuts. IEEE Trans. Intell. Transp. Syst. 19 (3), 802–813.
Zang, Y., Wang, C., Yu, Y., Luo, L., Yang, K., Li, J., 2017. Joint enhancing filtering for road network extraction. IEEE Trans. Geosci. Remote Sens. 55 (3), 1511–1525.
Zhao, J., You, S., Huang, J., 2011. Rapid extraction and updating of road network from airborne lidar data. Proceedings of the IEEE Applied Imagery Pattern Recognition Workshop.
Zhao, J., You, S., 2012. Road network extraction from airborne LiDAR data using scene context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 9–16.
Zhou, L., Zhang, C., Wu, M., 2018. D-LinkNet: linkNet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 192–1924.
Zhu, Q., Mordohai, P., 2009. A minimum cover approach for extracting the road network from airborne LIDAR data. In: Proceedings of the IEEE 12th International Conference on Computer Vision Workshops, pp. 1582–1589.
Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E., 2017. Toward multimodal image-to-image translation. In: Proceedings of the Conference on Neural Information Processing Systems, pp. 466–477.