ISPRS Journal of Photogrammetry and Remote Sensing 147 (2019) 178–192

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

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

A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds

Chenglu Wen[a], Xiaotian Sun[a], Jonathan Li[a,c,*], Cheng Wang[a,b], Yan Guo[a], Ayman Habib[d]

[a] Fujian Key Laboratory of Sensing and Computing for Smart Cities, 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
[d] Lyles School of Civil Engineering, Purdue University, West Lafayette, IN 47907, USA

ARTICLE INFO

Keywords:
Point cloud
Road marking
Extraction
Classification
Completion
Deep learning

ABSTRACT

Road markings play a critical role in road traffic safety and are one of the most important elements for guiding autonomous vehicles (AVs). High-Definition (HD) maps with accurate road marking information are very useful for many applications ranging from road maintenance, improving navigation, and prediction of upcoming road situations within AVs. This paper presents a deep learning-based framework for road marking extraction, classification and completion from three-dimensional (3D) mobile laser scanning (MLS) point clouds. Compared with existing road marking extraction methods, which are mostly based on intensity thresholds, our method is less sensitive to data quality. We added the step of road marking completion to further optimize the results. At the extraction stage, a modified U-net model was used to segment road marking pixels to overcome the intensity variation, low contrast and other issues. At the classification stage, a hierarchical classification method by integrating multi-scale clustering with Convolutional Neural Networks (CNN) was developed to classify different types of road markings with considerable differences. At the completion stage, a method based on a Generative Adversarial Network (GAN) was developed to complete small-size road markings first, then followed by completing broken lane lines and adding missing markings using a context-based method. In addition, we built a point cloud road marking dataset to train the deep network model and evaluate our method. The dataset contains urban road and highway MLS data and underground parking lot data acquired by our own assembled backpacked laser scanning system. Our experimental results obtained using the point clouds of different scenes demonstrated that our method is very promising for road marking extraction, classification and completion.

1. Introduction

With the rapid development of advanced driver assistance systems (ADAS) and autonomous vehicles (AVs), High Definition (HD) maps with lane level information have attracted the attention of many researchers. Such HD maps could provide highly detailed inventories of all stationary physical assets related to roadways such as road lanes, road edges, shoulders, dividers, traffic signals, signage, paint markings, poles, and all other critical data needed for the safe navigation of roadways and intersections by AVs. As one of the essential components in HD maps, road markings are used for guiding AVs. For example, lane lines indicate the driving area, arrows show the driving direction, and posted speed limits indicate the maximum safe speed under favorable driving conditions.

Many researchers have addressed the extraction and classification of road markings using both video and still images acquired by digital cameras mounted on a vehicle. Wu and Ranganathan (2012) extracted the maximally stable extremal regions (MSERs) from images and used template matching for road marking extraction. Lee et al. (2017) proposed a fully convolutional network for road marking detection and recognition based on images. Pan et al. (2018) developed a new type of convolutional network designed to detect lane lines. However, such image-based methods are sensitive to lighting and weather conditions.

Mobile laser scanning (MLS) systems, insensitive to lighting conditions, can acquire high-precision, 3D dense point clouds with intensity or "reflectance" raster images. Several methods have been developed to extract and classify road markings from MLS point clouds, which can be found in a recently published review article (Ma et al., 2018). The global threshold method has been widely used for road marking extraction based on MLS infrared reflective intensity images (Yang et al., 2012; Yan et al., 2016; Cheng et al., 2017). In practice, the MLS point

* Corresponding author at: Departments of Geography and Environmental Management/Systems Design Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada.
E-mail addresses: clwen@xmu.edu.cn (C. Wen), junli@xmu.edu.cn, junli@uwaterloo.ca (J. Li), ahabib@purdue.edu (A. Habib).

https://doi.org/10.1016/j.isprsjprs.2018.10.007
Received 11 July 2018; Received in revised form 16 October 2018; Accepted 16 October 2018

density varies in space over the distances between the laser scanners mounted on the vehicle and objects on a roadway. Cheng et al. (2017) improved the robustness of the algorithm by adopting a normalized method to reduce the influence of distance on the intensity value. However, in these methods, the normalization parameters vary from scene to scene. The global threshold-based method does not work well on reflective intensity images with different point distributions. Multi-threshold methods have been developed for road marking extraction by segmenting raw road surface point clouds into many patches and calculating a dynamic threshold in every patch (Guan et al., 2014; Yu et al., 2015). The performance of these methods depends on suitable patch size and requires patch location (including both road marking and the road surface in each patch). Also, a Gaussian Mixture Model (GMM) based method was used to determine whether a point belongs to a road marking (Soilán et al., 2017) by estimating GMM parameters with two single Gaussian distributions. Such a dynamic threshold method has the same disadvantages as the methods presented in Guan et al. (2014) and Yu et al. (2015).

The three main challenges of using MLS point clouds for road marking extraction and classification (see Fig. 1) are summarized as follows:

(1) Variations in reflective intensity and point density

Point clouds are usually collected by MLS systems that are driven across different lanes. The density and intensity of MLS point clouds typically vary in space over distances between the laser scanners mounted on the vehicle and road markings. In general, road markings closer to the trajectory of the vehicle have higher reflective intensities and densities. With increasing distance, the incidence angle of a laser beam becomes larger, while both the reflective intensity and the point density decrease. As shown in Fig. 1(a), the intensity and density of two lanes, which are approximately 3.5 m apart (one lane width), may differ by up to 20% and 50%, respectively. In this case, most of the threshold-based extraction methods based on assumption that the acquired MLS point clouds having consistent and even reflective intensity and density, do not perform well.

(2) Low contrast between road markings and surrounding road surface

Road wear is a very common situation. A worn road marking usually shows a lower reflective intensity value than a normal marking. This typically results in a low-intensity contrast between road markings and its surrounding road surface, see Fig. 1(b), which makes the existing intensity-based methods (whether local or global) ineffective.

(3) Lack of consistency of road markings in point clouds

Lack of consistency in reflective intensity of road markings and low-intensity contrast of the points result in the incomplete detection of road markings. Furthermore, occlusion from the objects in complex road environments also leads to the incomplete detection of road markings. Complete road markings are generally required in some applications such as HD maps for AVs. In addition, incomplete road marking data are often inevitable due to occlusions. To build a complete map, manual inspection and editing are needed to complete the road marking data. As such, we provide an auxiliary semi-automated/automated approach to road marking completion to assist the manual inspection and editing task. Thus, quality enhancement of the road marking point clouds is required.

In this paper, a deep learning-based framework was developed for automated extraction, classification and completion of road markings. A pixel-level U-net segmentation network was developed to extract the road markings. The road surface was segmented first into a series of patches as training data. The Intersection over Union (IoU) loss in the U-net model was used to guide training instead of cross-entropy in

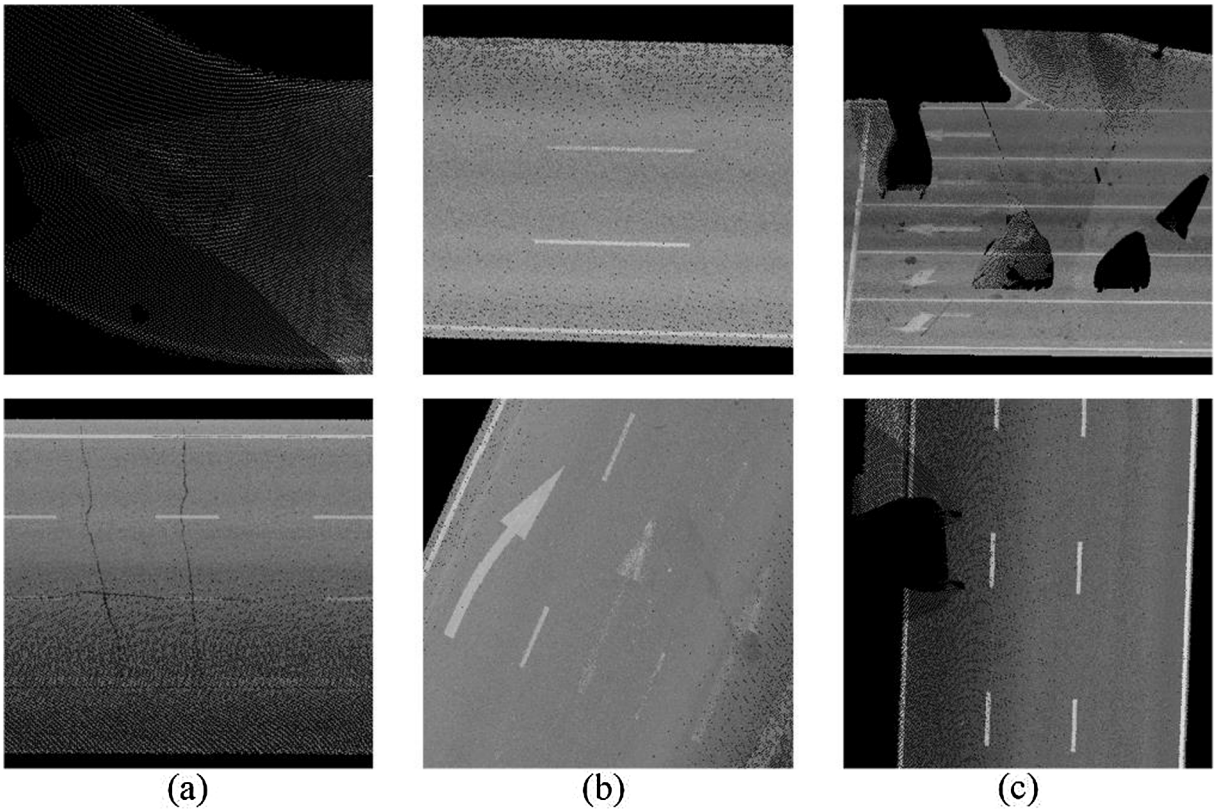


Fig. 1. Examples of road markings: (a) intensity images with varied point densities, (b) road surface with low-intensity contrast between road markings and surrounding road surface, and (c) road surface showing incomplete road markings acquired.

order to obtain more complete road markings. The binary classification was applied using a segmentation network to achieve road marking extraction. A method based on clustering and convolutional neural networks (CNN) was developed for road marking classification. At first, the large-size road markings such as lane lines and pedestrian crossings were classified by multi-scale clustering. Then, a CNN classifier was trained to classify the small-size road markings. A conditional generative adversarial network (cGAN) was used for the small size road marking completion. A context-based method was developed to complete solid and dashed lane lines. The proposed framework produces better road markings for offline HD map generation, further providing a fundamental map dataset for road marking matching during the online perception of an AV.

The main contributions of this paper can be summarized as follows:

First, a modified U-net segmentation network was developed to extract road markings. The effect of varying intensity was considerable reduced by learning the patches at different locations. Because each patch presents a different intensity contrast, the model also performs well on data with a low-intensity contrast ratio.

Second, a combined clustering and CNN model was used to classify road markings. The hierarchical framework of the combined clustering and CNN model made classification of road markings with different sizes more effective.

Third, a cGAN-based model along with context information was used to improve the quality of point clouds to reduce in completeness of road markings. The cGAN model needs only a small amount of training data to achieve small size road marking completion. Our context-based method further improves completeness of lane lines by taking into account their continuity and regularity.

Last, a road marking dataset was built based on the proposed framework, which will be released to encourage further studies. The dataset contains three types of scene data: highways, urban roads, and underground parking lots with both raw point clouds and labelled road marking ground truths.

2. Related work

2.1. Extraction and classification of road markings in point clouds

Because road markings usually show higher intensities than road surfaces, threshold-based methods have been commonly used for road marking extraction. The multi-segment threshold strategy (Yu et al., 2015) first divides point clouds into several blocks. Then, each block is divided into multi-segment structures with a width value. Finally, to extract road markings, the multi-segment structures are segmented separately. Spatial Density Filtering (SDF) distinguishes road marking points from noise by calculating the spatial density at every point. Guan et al. (2015) proposed Weighted Neighboring Difference Histogram (WNDH) and Multiscale Tensor Voting (MSTV) methods to segment and extract road markings from noise corrupted Geo-Referenced Feature (GRF) images. To segment road markings, WNDH first calculates the intensity histogram of the point cloud and obtains a dynamic threshold. To extract the correct road markings, the MSTV algorithm further filters out some noise.

To extract road marking point clouds, Soilán et al. (2017) proposed a method based on the Gaussian Mixture Model (GMM). In their method, the intensity distribution of a road that contains road markings can be separated into road surface and road markings that are approximated by Gaussian distributions, with the higher mean distribution representing the intensity distribution of the road marking points. Their method calculates the probability of a point belonging to a road marking by estimating the parameters of the two Gaussian distributions.

Following the extraction process, the road markings are classified into different groups for further applications. Yu et al. (2015) proposed to first use Euclidean distance clustering to group markings into clusters based on the Euclidean distances to their neighbors. First, a voxel-based normalized cut segmentation method was used to group road markings into large and small size road markings. Then, a trajectory-curb line-based method was proposed to classify large-size markings. A Deep Boltzmann Machine (DBM) was used to classify small-size markings. Cheng et al. (2017) proposed a road marking classification method, using four geometric features including area, perimeter, estimated width, and orientation. Because this method uses a simple segmentation strategy, it is difficult to handle markings like text. In addition, it is difficult for these four geometric features to correctly represent an incomplete road marking.

2.2. Point cloud completion

To achieve point cloud completion, scattered data fitting methods, such as Moving Least Squares (Alexa et al., 2003; Wang and Oliveira., 2007), Multi-level Partition of Unity Implicits (Ohtake et al., 2003) or Radial Basis Functions (Carr et al., 2001), etc. were used to smoothly fill holes on surfaces reconstructed from point clouds. Context-based methods (Sharf et al., 2004; Park et al., 2005; Xiao et al., 2007; Savchenko and Kojekine, 2002) are beneficial to recover lost geometric features. Context information is extracted as prior to fit the lost parts. Regularity-based methods have been used to complete point clouds. A missing point cloud is retrieved from other locations based on rules that most objects are regular and symmetrical (Law and Aliaga, 2011; Figueiredo et al., 2017; Thrun and Wegbreit, 2005; Kroemer et al., 2012) and structurally repeated (Zheng et al., 2010; Friedman and Stamos, 2012).

Additional information, such as images, can also be incorporated to assist in the completion process (Li et al., 2011; Xu et al., 2006; Lai et al., 2016; Doria and Radke, 2012). The method of Li et al. (2011) recovered the missing parts by detecting repetitive patterns on each of the depth layers, which are formed by mapping the point clouds to different depth ranges. Transforming a 3D point cloud into a depth image, Doria and Radke (2012) used an image in-painting method to complete the depth image and then transform it back into the point cloud. This method can also be used to complete texture and structure in MLS scans.

Zelek and Lunscher (2017) proposed a deep learning-based method to achieve point cloud completion of a human foot point cloud. The depth map of a half foot is the input into the network, while the depth map of another half is the output.

3. Method

The proposed method is comprised of three main steps for road marking extraction, classification, and completion.

3.1. Modified U-net-based road marking extraction

The usability of current methods is dramatically affected by consistent intensity, density variation and intensity contrast between the road markings and road surface. In this paper, a deep learning-based road marking extraction method, which overcomes the above issues, is proposed to extract the road markings.

Previous studies (Guan et al., 2014; Cheng et al., 2017) have shown that the elevation of a road point cloud contributes little to road marking extraction. Here, elevation refers to the height of a 3D point in the point cloud coordinate system (z-axis coordinates). The road surface is assumed to be parallel to the xoy plane in the point cloud coordinate system we established. Thus, the road is a typical 2D structure. In this paper, the original point clouds have been filtered (Zai et al., 2018), leaving only the road point cloud. In our method, 3D point clouds are first projected onto a horizontal plane and gridded as a 2D image. This projection step greatly reduces computational cost. The average intensity of all points in each grid cell is calculated to represent this grid

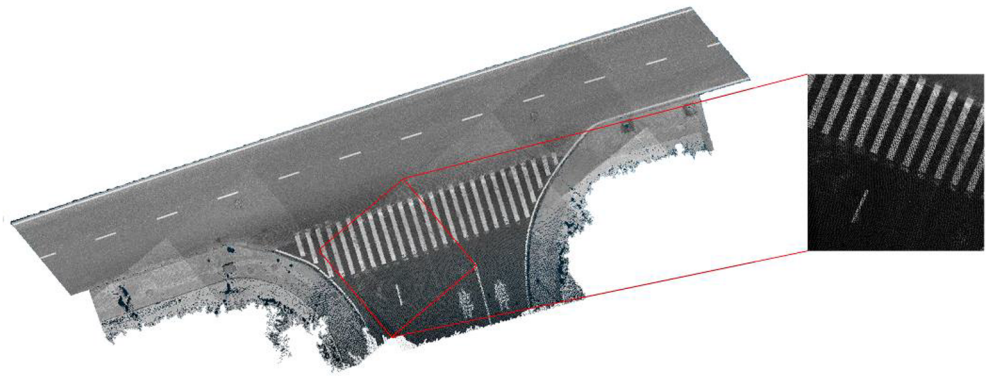


Fig. 2. Intensity image generated by projection.

cell. Fig. 2 shows the intensity image generated by projection.

Because a grid cell either represents some road marking points or represents some non-road marking points, we can consider the road marking extraction task as a binary classification problem. A modified segmentation network U-net (Ronneberger et al., 2015) is used to classify every pixel. U-net (see structure in Fig. 3) is a special encoder-decoder network, which sets the connection between the encoder and decoder. For encoder-decoder networks, the downsampling of the encoder layers usually results in missing many details of the image, especially edge information. It is difficult for the decoder layers to restore these details. The feature maps, which are saved and used in each encoder layer of U-net, contain many details to better guide the decoder. Thus, the outputs of U-net are sharper and more accurate. We consider introducing U-net into road marking extraction, because it can learn not only the difference in intensity from a large number of labeled samples, but also the shape of road markings.

In encoder layers, a 3×3 convolution kernel and ReLU activation function are used. When training, a batch normalization operator is added. Each encoder layer performs two convolutions the second convolution result must be saved. After convolution, the feature maps are down-sampled by 2×2 size and 2×2 stride max pooling. The results are input into the next encoder layer. In each decoder layer, the feature maps are up-sampled by deconvolution, where the size of the deconvolution kernel is 2×2 and the stride is also 2×2. Then, the results of deconvolution are connected with the saved convolution results in corresponding encoder layers. The connected feature maps are run by two convolutions. Finally, the feature maps are converted to a segmentation result by a 1×1 convolution, softmax activation function, and an argmax function.

Compared with the original U-net model, we modified the model by applying the intersection-over-union (IoU), instead of cross-entropy, as a loss function to achieve better performance. In our task, we aim to obtain complete road markings instead of classifying the road surface and marking pixels more accurately. By maximizing IoU, the network extracts more complete road markings in the output. The IoU loss function is defined as follows:

$$\begin{cases} IoU = \frac{RMPixel_{predict} \cap RMPixel_{gt}}{RMPixel_{predict} \cup RMPixel_{gt}} \\ loss_{IoU} = -IoU \end{cases} \quad (1)$$

where $RMPixel_{predict}$ is the pixel that represents road markings in the network output. $RMPixel_{gt}$ is the pixel that represents road markings in the ground truth. The IoU segmentation result is closer to ground truth. Thus, it is reasonable to use $-IoU$ as a loss function to guide network updating parameters. To solve the intensity variation problem, images with different intensity were used as training samples.

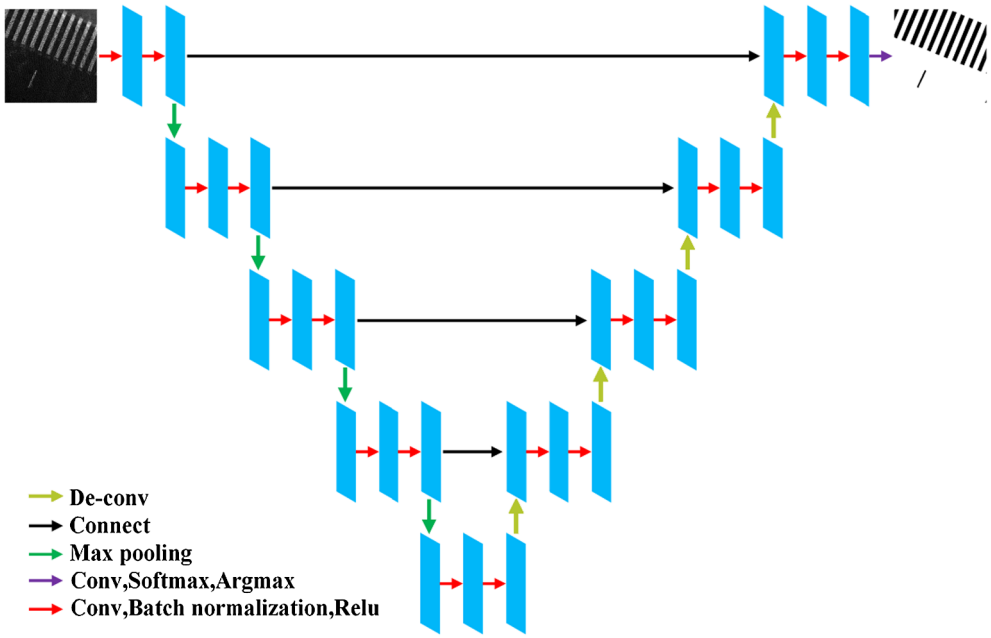


Fig. 3. U-net structure.

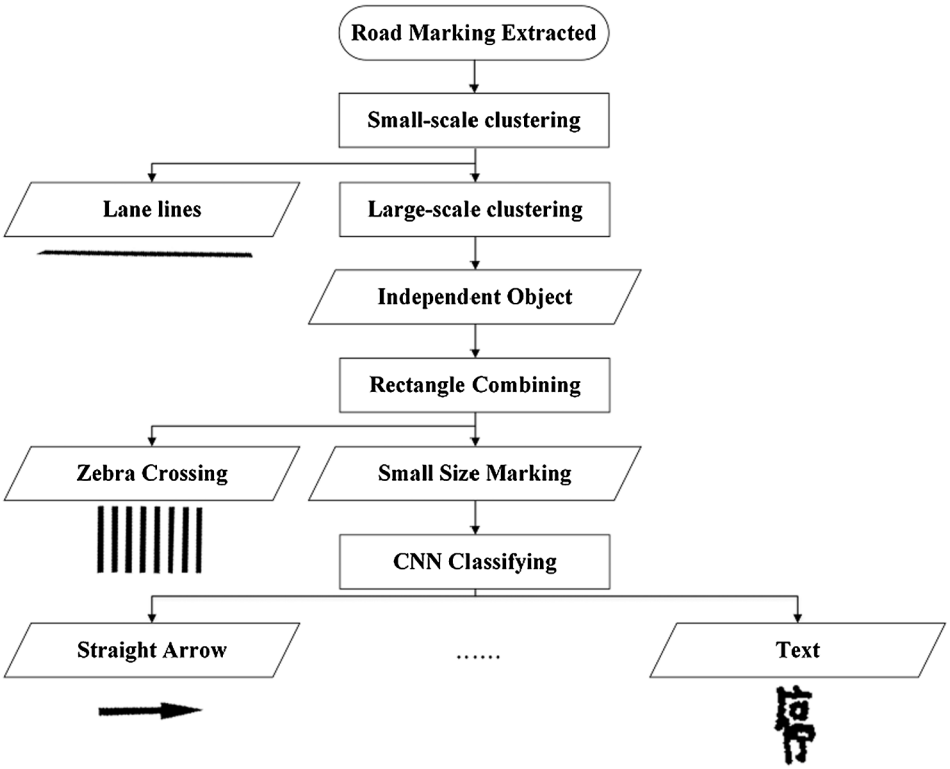


Fig. 4. Flowchart of road marking classification.

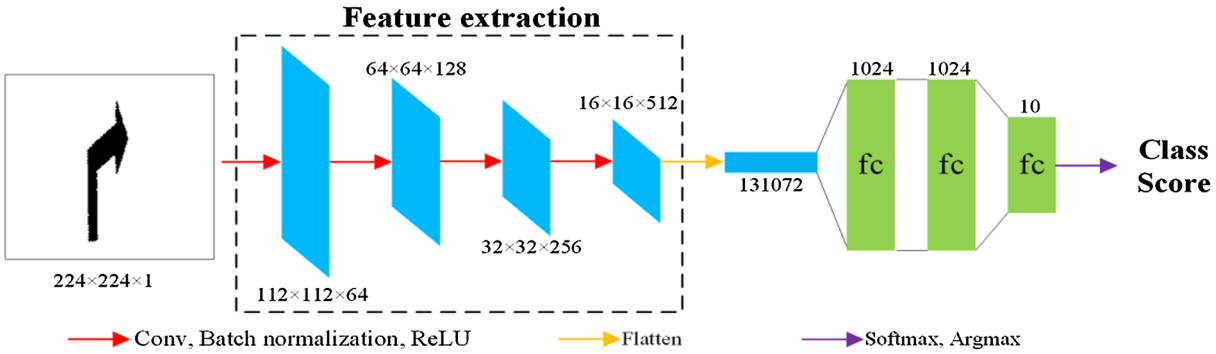


Fig. 5. CNN classifier structure.

3.2. Hierarchical road marking classification

Road marking classification consists of two main steps: (1) a multi-scale clustering algorithm classifies large size road markings; (2) a CNN classifier is trained to classify small size road markings, including straight arrows, turn arrows, and text. The flow chart for road marking classification is shown in Fig. 4.

3.2.1. Multi-scale clustering

Large size road markings include lane lines and zebra crossings. A distance-based Euclidean clustering method segments objects and considers that a series of points, whose distance is less than a set threshold, belong to the same object. First, a small-scale Euclidean clustering is used to segment all extracted objects and to separate the objects as much as possible by setting a small distance threshold (T_s). Although some of the small markings can be incorrectly segmented into multiple objects, lane lines can be effectively separated from other markings. The most significant difference between lane lines and other markings is length; therefore, we classify markings longer than a length threshold, (L), into lane lines. After removing the lane lines, the remaining road markings are classified into independent objects by a large-scale Euclidean clustering, whose threshold is set as T_l. This step correctly segments road markings except the zebra crossing, which is divided into a series of rectangular markings. Finally, some small size road markings are combined into zebra crossings. We compute the center point coordinates and minimum bounding rectangle coordinates for each marking and, using these coordinates, compute the relationships of the positions among the markings. A series of rectangular markings is considered to belong to the same zebra crossing if their bounding boxes are parallel and the distance of their center points, D, is in a particular range.

3.2.2. CNN classifier

After the previously described multi-scale clustering process, lane lines and zebra crossings are removed. The remaining markings are some small size road markings that are grouped into the following ten classes: two classes of dashed lines, four classes of texts, two classes of arrows, a class of diamond, and a class of triangle marking. A CNN classifier (Fig. 5) is designed as follows: A four-layer convolution network, with kernel size 3×3 and stride of one, is used to extract the

features. During the training process, batch normalization is used after each convolution operator. Then, the activation function, ReLU, is added. After feature extraction, three-dimensional feature maps are flattened to one-dimensional feature vectors. Then, the vectors are sent to fully connected layers. The first two fully connected layers contain 1024 nodes, and the activation function is ReLU, also. To prevent overfitting, a dropout operation (ratio = 0.5) is used during training. The last fully connected layer contains ten nodes, and the activation function is softmax, whose output is class score. The image is classified as the category that gets the highest score.

A set of small size marking training samples is prepared manually as templates. The templates are rotated at different angles, and some defects are added to generate additional training samples. Finally, we produced a total of 4000 training samples, with 400 samples in each category.

3.3. Joint learning and context completion

In the past, most research on object completion was rule-based. However, designing rules is complex, and manually designed rules are limited to some specific applications. We propose to integrate a deep learning-based method and a context-based method for road marking completion.

3.3.1. Deep learning-based road marking completion

Deep learning-based image completion showed good performance in some recent studies. Pathak et al. (2016) designed a context-encoder network to achieve image completion. Isola et al. (2017) developed an image translation network, based on an improved conditional generative adversarial network (cGAN). Road marking completion, which translates an image with an incomplete road marking to an image with a complete road marking, can be considered essentially an image translation task. Fig. 6 shows the cGAN model structure on which our road marking completion framework is based.

In this section, the training data here is the same data used to train the CNN classifier. The generator is similar to the U-net encoder-decoder network, and feature connection is added between each encoder and decoder layer. An encoder consists of eight convolution layers; the size of each convolution kernel is 4 × 4 and the stride is two. Especially, the activation function of every encoder layer is Leaky ReLU. A decoder is achieved by eight deconvolution layers; the parameters are the same as those for the convolution layers. The last layer uses a tanh activation function; the other layers use ReLU. When training, batch normalization is used in every layer. The discriminator is a five-layer convolution network to regress the probability that indicates if the output of the generator is real or fake.

During the training process, the generator and the discriminator are successively trained to update their respective network weights. By outputting a fake, realistic image as the input for the discriminator, the generator attempts to cheat the discriminator as much as possible. However, it is expected that the discriminator can distinguish whether the discriminator input is real or generated. The generator and discriminator losses are defined as follows:

$$loss_D = \log(1 - P_{fake}) - \log(P_{real}) \tag{2}$$

$$\begin{cases} loss_G = -\log(P_{real}) + \lambda loss_{L1} \\ loss_{L1} = |y - G(x)| \end{cases} \tag{3}$$

where $loss_D$ is the loss function of the discriminator network, P_{fake} is the probability that the discriminator determines if its input is fake, and P_{real} is the probability that the discriminator determines if its input is real. The generator loss function is $loss_G$; $loss_{L1}$ is the $L1$ distance bound term, y is the ground truth (a completion road marking), x is the input of the generator (an incomplete road marking), G is the generator network, and λ is a hyperparameter. By minimizing $loss_D$ to update the discriminator parameters, we aim to achieve larger P_{real} and smaller P_{fake} values. If the generator tries to cheat the discriminator, small $loss_{L1}$ and large P_{real} should be achieved. Therefore, to update the generator parameters, we minimize $loss_G$.

3.3.2. Context-based completion

The cGAN network above is used to complete some small size road markings, after which we complete other road markings based on context information related to the road marking design rules (Fig. 7). Usually, the lane lines on both sides of a road are continuous solid lines, but, for reasons mentioned above, they are sometimes broken. If it is broken, a lane line can be connected. In addition, the incremental spaces in the dashed lines must be consistent, and the distance between parallel dashed lines must be uniform. In this context, some missing road markings are completed. We first calculate the center point of each dash within the dashed line. Then, the distance between the center points of adjacent dashes and the distance between the parallel dashed lines are computed. Next, the average value and standard deviation of distance are computed. Last, the position of the missing dashed line is located if the standard deviation is unusual.

After completing the road markings on the images, to reduce unnecessary loss of precision, we transform the images back into a 3D point cloud. The images were obtained from a point cloud projection process; therefore, we find the corresponding point cloud according to the position of each pixel. When the road marking is complete, some pixels without corresponding points were added. For these pixels, we use the Inverse Distance Weighted (IDW) interpolation algorithm to compute the coordinates of the points. The IDW algorithm uses an inverse of distance as a weight to calculate a new variable. If a pixel has no corresponding points, we randomly generate the x, y coordinates of several points at this position, then use their neighboring points to

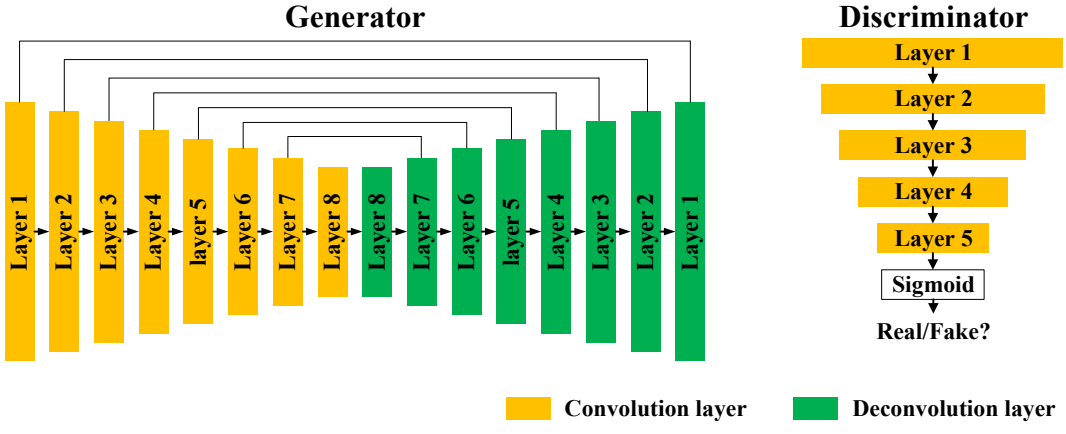


Fig. 6. cGAN structure.

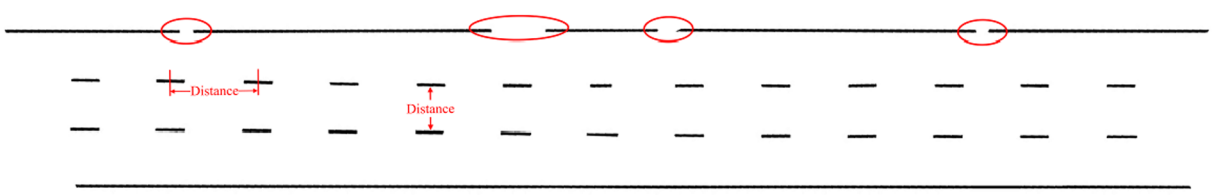


Fig. 7. Road marking completion based on road marking design rules.

compute the z coordinates with IDW as follows:

$$\begin{cases} z = \frac{\sum_{i=1}^{n} w_i z_i}{\sum_{i=1}^{n} w_i} \\ w_i = \frac{1}{(x-x_i)^2 + (y-y_i)^2} \end{cases} \quad (4)$$

where (x_i, y_i, z_i) is the coordinate of the ith neighbor point.

4. Dataset

The data in this study includes highway data, urban data and underground garage data. The study area for the MLS data is located in Xiamen, China. A section of highway data and urban data are measured by a RIEGL VMX-450 system. In addition, using an own assembled backpacked laser scanning system (Wang et al., 2018), we collected some road surface point clouds of an underground garage. Some point cloud data in the study data are of low-quality, such as intensity variation, density variation and low-intensity contrasts in part of the data.

To create a 3D road marking dataset, we manually labeled some data. At first, all road markings were manually annotated on the intensity projection images. The ground truth data were saved as binary matrices. Then, according to the position of each 2D grid cell, the 2D ground truth was restored to 3D point cloud ground truth. Finally, all the road markings were segmented as independent objects by a clustering algorithm. The category of each marking and the 3D coordinates of each point are given. The average point density of the MLS data is about 5000 points/m^2, and the precision is 8 mm. The average point density of the underground garage data is about 1800 points/m^2, and the precision of the data is 3–5 cm. The number of different road markings is listed in Table 1. Because of the small number of road markings in certain categories, we merged some categories. When classifying, the merged categories are still distinguished as different categories.

5. Results and discussion

In this section, the whole framework is evaluated from the aspect of the road marking extraction, classification, and completion steps and then compared with other algorithms. The results of each aspect are shown, and the reasons for errors are analyzed.

When gridding point clouds, we used a grid cell of 4 cm × 4 cm for highway, urban, and underground garage data. This grid cell size effectively preserves the details of the road markings and greatly reduces the amount of data that must be processed. The selection of the grid cell size is related to the density of the data. In practice, a larger grid cell size can be considered when applied to sparser data.

Table 1
Number of road marking in dataset from MLS system and backpacked system.

Category	Number from MLS system	Number from backpacked system
Dashed line	719	65
Text	19	/
Straight arrow	8	27
Turn arrow	18	41
Diamond	14	/
Triangle	5	/
Lane line	262	200
Crossing	32	23

Two models were trained in the road marking extraction experiment. One was trained with 3000 samples for processing highway and urban data. The other was trained with 1000 samples for underground garage data. In the classification experiment, 4000 samples were used to train one model to classify the road markings in highway, urban, and underground garage data. Because the categories are similar in these three scenes, the same training data, as used for classification, was used in the completion experiment. The complete road markings were the ground truth.

We used different hyper parameters for different models. The initial learning rate, batch size and epochs are 0.001, 4, and 300, respectively for the U-net model; 0.0001, 32, and 40, respectively for the small-size road marking classifier; 0.0002, 4, and 300, respectively for the cGAN model. These three models were trained by the Adam optimizer. The momentum term is 0.5. The design of the network structure depends on experience; therefore, the parameters were selected through multiple experiments. For example, when selecting the size of the convolution kernel, it is common to use a small size kernel (e.g. a 3×3 convolution kernel) repeatedly to expand the receptive field (Szegedy et al., 2016).

5.1. Intensity-invariant road marking extraction

Road marking extraction is evaluated through the following performance metrics:

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$F1_{score} = \frac{2 \times Precision \times Recall}{Precison + Recall} \quad (7)$$

where TP, FP and FN are the number of true positives, false positives and false negatives, respectively.

5.1.1. Highway scene

Because highways are regularly maintained, the roads are clean, and the markings are in good condition. Thus, road surface point clouds in a highway scene are of high-quality. In a highway scene, most road markings are dashed lines and continuous lane lines; the number of other types of road markings is very small, which indicates that the distribution of different types of samples in the data is not uniform. In the road marking extraction step, we only performed binary classification in the hope of reducing the impact in the case where the data is not uniform; however, the results of marking extraction in a highway scene show that impact still exists. As seen in Fig. 8, some markings (arrows) are incomplete; thus, marking extraction is not ideal. However, small incompletions do not affect the classification step. We deal with them in the completion step.

5.1.2. Urban road scene

Urban road scenes are very different from highway scenes. Possibly, some road surfaces have some dirt, and some road markings are badly worn, which makes it extremely difficult to extract road markings in an urban scene.

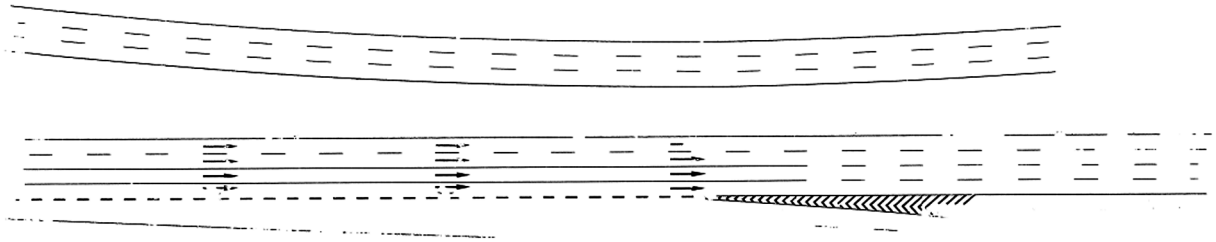


Fig. 8. Extracted road markings in a highway section.

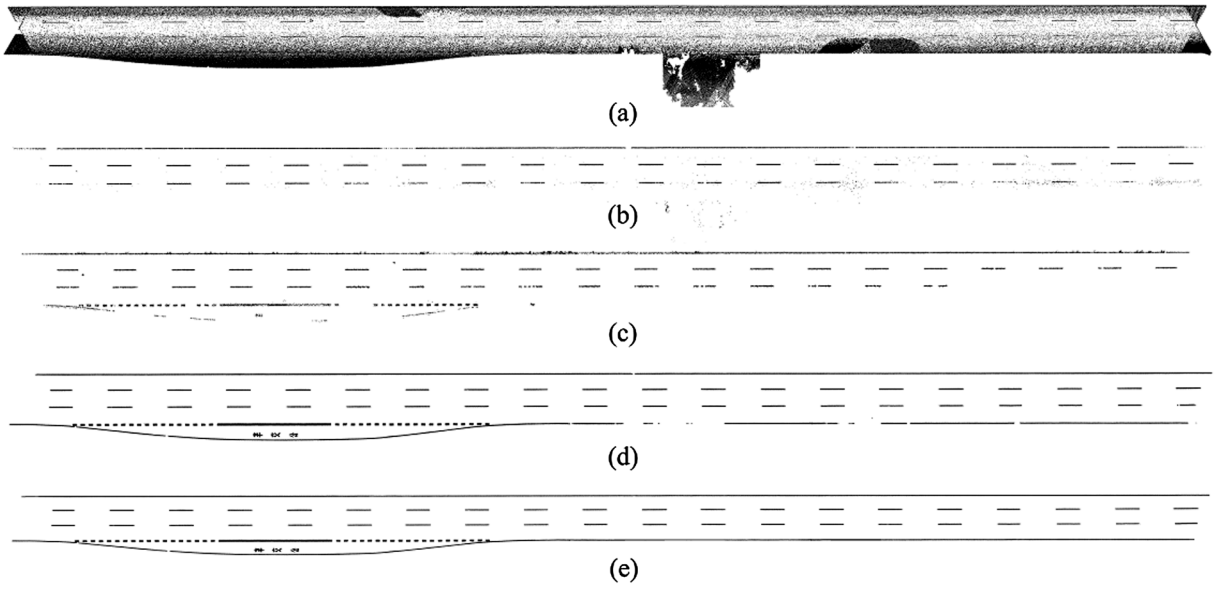


Fig. 9. Extracted road markings on a straight road section: obtained by the methods of (a) Cheng et al. (2017), (b) Yang et al. (2012), (c) Soilán et al. (2017), (d) ours, and (e) ground truth data.

We compared different algorithms using a low-intensity contrast point cloud and analyzed the results. The efficiency was compared with the methods of Cheng et al. (2017), Yang et al. (2012), and Soilán et al. (2017). Fig. 9 shows the results on a straight road section with low-intensity contrast. In this case, the method of Cheng et al. (2017) and the method of Yang et al. (2012), which calculate a global intensity threshold to extract road markings, do not yield a good result. Fig. 10(a) presents the intensity distribution in this example. The intensity of the road marking points on the side of the road is lower than the intensity of the non-marking points in the center of the road. Therefore, it is impossible to choose a suitable threshold range to distinguish between marking points and non-marking points, causing the results of the method of Yang et al. (2012) to miss part of the road markings. Similarly, because the method of Soilán et al. (2017) uses a GMM with two single Gaussian distributions, it is difficult to extract road marking points with that method.

Fig. 10(b) shows the intensity histogram for this example. The red curve is the intensity histogram for the marking point cloud. The blue curve is the intensity histogram for the non-marking point cloud. As shown in Fig. 10, the number of marking points is far smaller than the number of non-marking points. In addition, the intensity ranges for the marking and non-marking points are 30,000–48,000 and 23,000–40,000, respectively. There is a large overlap between the ranges. However, in this situation, commonly used binarization algorithms, the Otsu algorithm (Otsu, 1979) for example, cannot calculate an effective threshold. Thus, the method of Cheng et al. (2017), an Otsu-based method, does not perform well. Table 2 presents the performance of the different methods used in this example. The precision, recall and F1-score for the method of Cheng et al. (2017) are 18.52%, 28.03%, and 22.30%, respectively. There are many noises in these results; therefore, the precision and recall

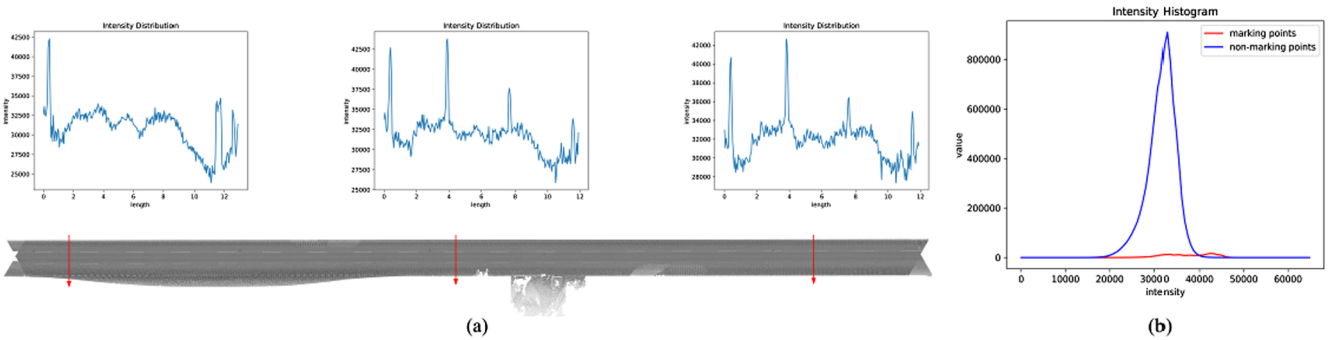


Fig. 10. (a) Intensity distribution of Fig. 9. (b) Intensity histogram of Fig. 9.

Table 2
Extraction result comparison with other methods.

Method	Our dataset			TUM-MLS dataset		
	Precision	Recall	F1-score	Precision	Recall	F1-score
Cheng et al. (2017)	18.52%	28.03%	22.30%	58.39%	42.52%	49.21%
Yang et al. (2012)	92.31%	33.45%	49.11%	68.22%	**82.60%**	74.72%
Soilán et al. (2017)	94.37%	40.24%	56.42%	70.79%	64.22%	67.35%
Ours (cGAN-based)	90.15%	82.33%	86.06%	82.56%	76.48%	79.40%
Ours (U-net-based)	**95.97%**	**87.52%**	**91.55%**	**89.12%**	81.31%	**85.04%**

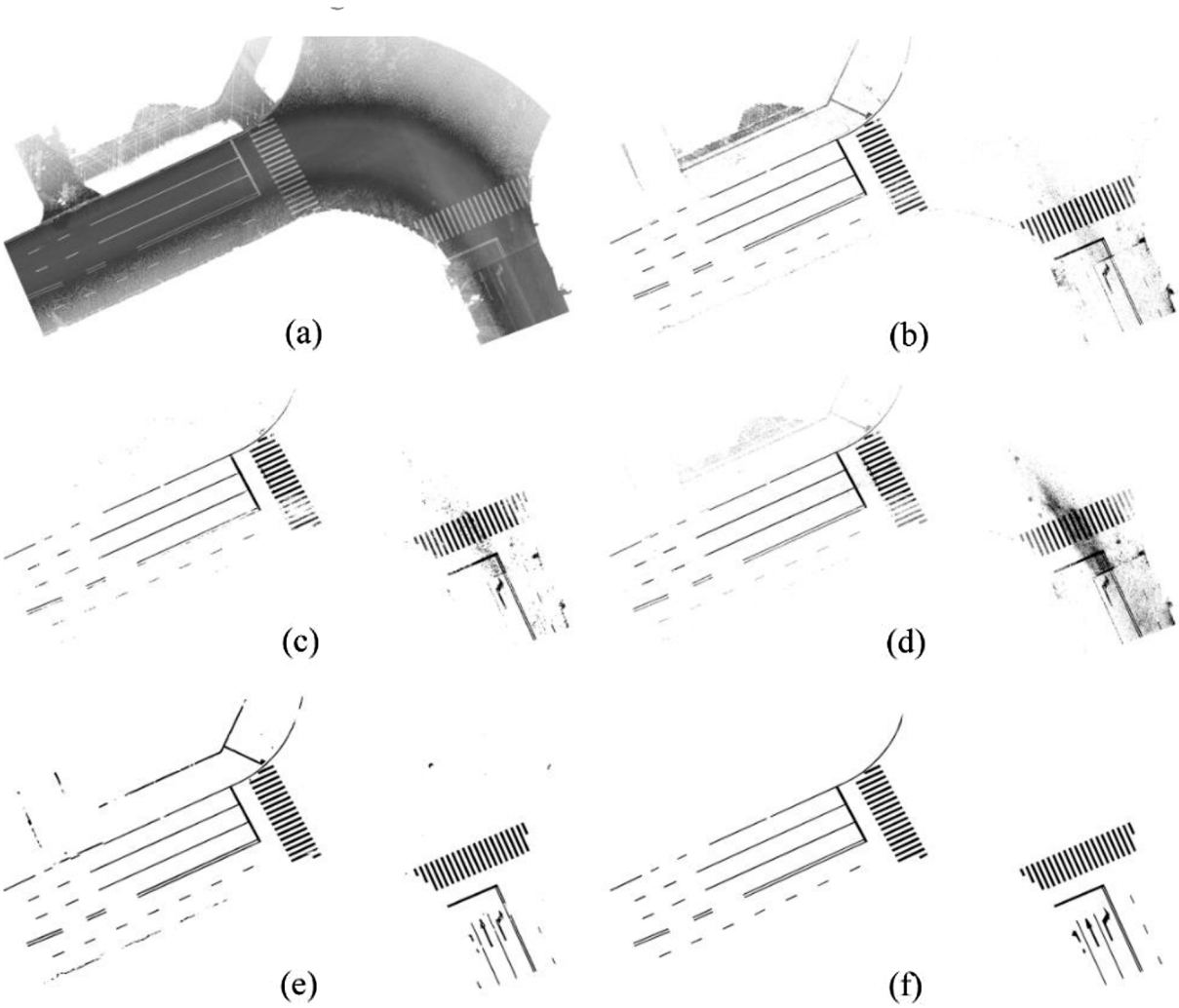


Fig. 11. Intensity image of a road intersection surface (a) and the extracted road markings using the methods of (b) Cheng et al. (2017), (c) Yang et al. (2012), (d) Soilán et al. (2017), (e) ours, and (f) ground truth.

rates are low. The methods of Soilán et al. (2017) and Yang et al. (2012) achieve high precision (94.37% and 92.31%, respectively), but the recall rates (40.24% and 33.45%, respectively) and F1-scores (56.42% and 49.11%, respectively) are insufficient. As shown in Fig. 9, a lane marking, some dashed lines, and some text are missing in Fig. 9(b); a lane is missing in Fig. 9(c). By achieving precision of 95.97%, recall of 87.52% and F1-score of 91.55%, our method performs better than the other methods.

Instead of a simple threshold segmentation algorithm, we used a modified U-net to extract the road markings. For road marking extraction, the shape of the marking is also an important information; however, all threshold segmentation algorithms based on statistics ignore shape information. A CNN-based segmentation network completes road marking extraction through a series of convolution kernels. During the learning of the convolution kernels, changes in intensity information and the shape of the road markings are both considered. Because our method is less sensitive to intensity distribution, it outperformed other methods.

In addition, we developed a cGAN-based method for road marking extraction (see Table 2). To extract road markings, a cGAN model was trained to achieve the transform from the original images to labeled images. The cGAN model was trained over 300 epochs. Loss gradually decreased and eventually stabilized over a small range of changes. The average precision, recall, and F1-score of the cGAN model is 90.15%, 82.33%, and 86.06%, respectively. The U-net model performed better than the cGAN model on this task. It was mainly manifested in

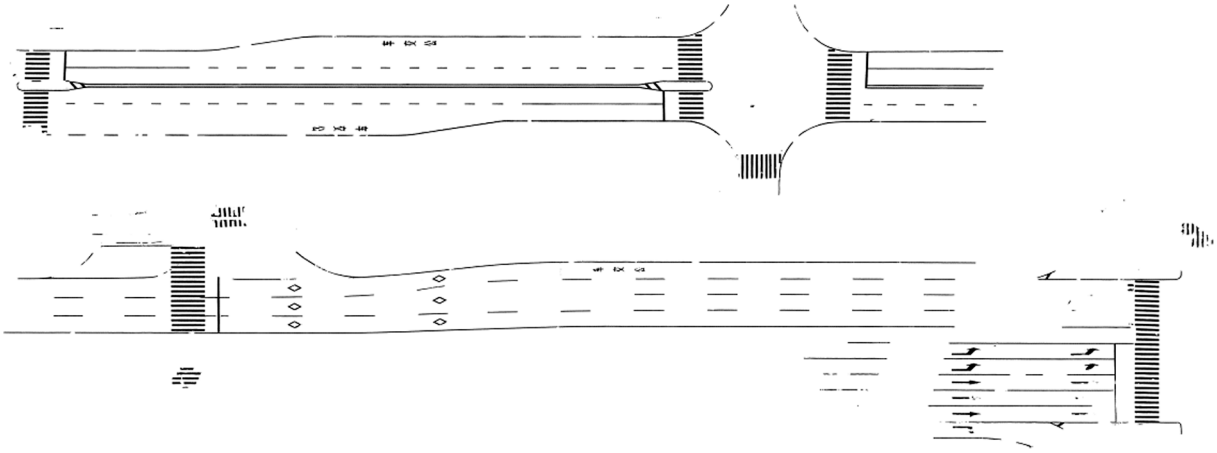
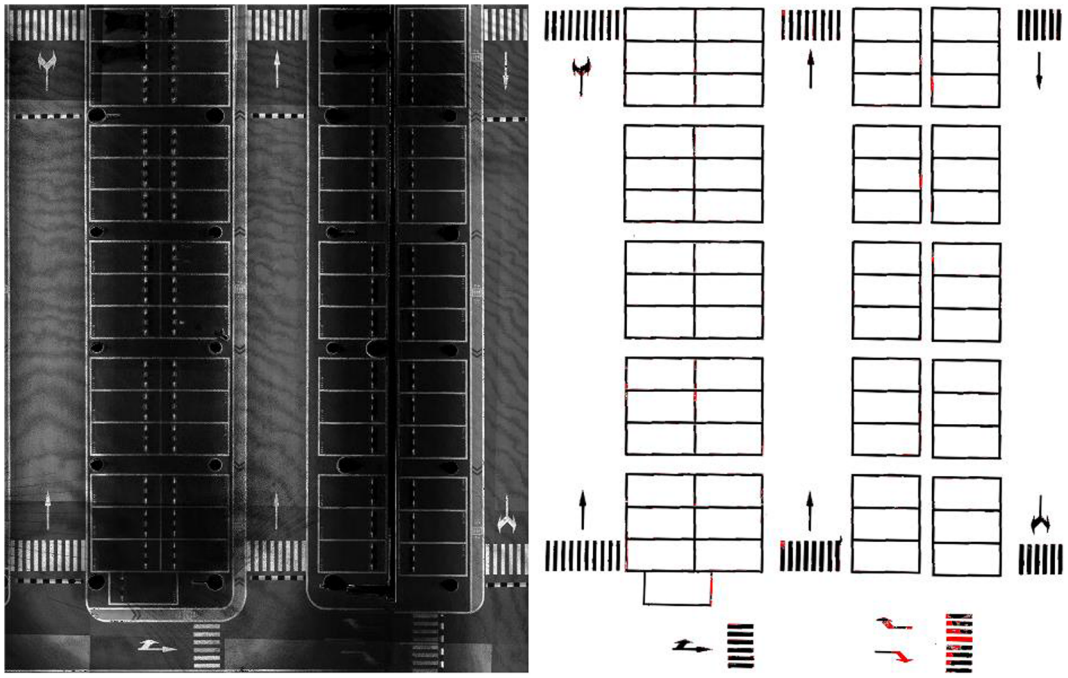


Fig. 12. More extraction results in an urban road section.



(a) (b)

Fig. 13. Road marking extraction from underground garage data. (a) The raw data. (b) Black pixels are extraction results and red pixels are ground truth.

Table 3
Parameters in multi-scale clustering.

Name	Definition	Value (m)
T_s	Small-scale Euclidean clustering threshold	0.12
T_l	Large-scale Euclidean clustering threshold	0.5
L	The minimum length of lane line	7.5
D	The distance of rectangular marking centers	0.8–1.2

Table 4
Error rate of road marking classification.

Road scene	scenes			Average rate
	Urban road	Highway	Underground	
Error rate	2.41%	6.01%	3.79%	4.07%

incomplete extraction of road marking details. The reason for the lower accuracy of cGAN model may be that the discriminator is redundant in this task, and the loss function cannot directly guide the extraction.

Fig. 11 shows the road marking extraction results at a crossroad. In this data, it contains both old (uneven and worn) road surface and newly constructed (even) road surface. The intensity characteristics of these two road surfaces are quite different. We manually labeled the ground truth of this region based on image and point cloud. Other algorithms have obvious noise in the extraction results for this special part. In particular, the results of the method of Soilán et al. (2017) show that two Gaussian models do not fit such data distributions. However, our method achieves very good results, even on an uneven road surface. Fig. 12 shows more urban road extraction results.

In addition, we compared our method with the methods of Cheng et al. (2017), Yang et al. (2012), and Soilán et al. (2017) using the TUM-MLS point cloud dataset of an urban scene (Gehrung et al., 2017). We selected a 400 m long road section from this dataset for testing. Table 2 lists the road marking extraction results using the extraction model

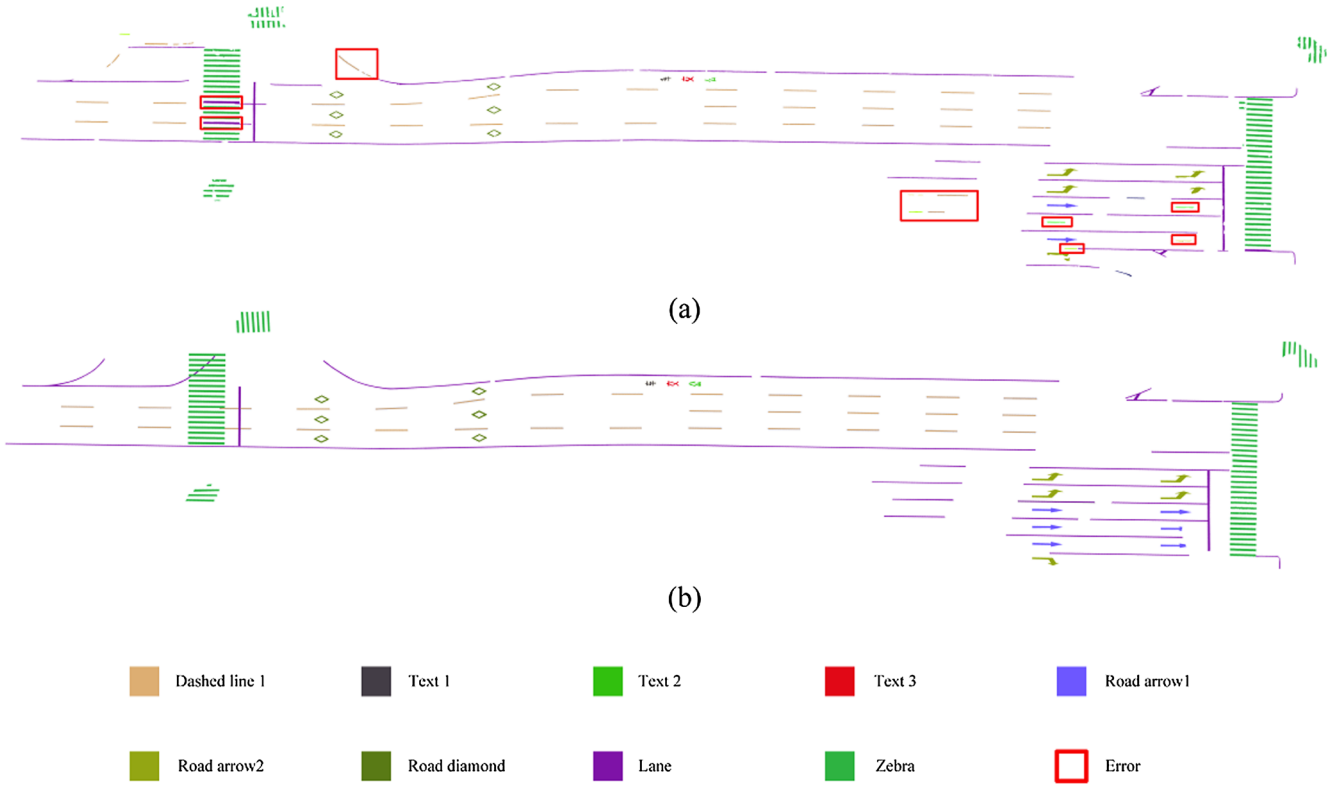


Fig. 14. Road marking classification on an urban road. (a) Classification results. (b) Ground truth.

trained with our dataset on TUM-MLS. The results in Table 2 for precision, recall, and F1-score, using our U-net-based method with the TUM-MLS dataset (89.12%, 81.31% and 85.04%, respectively) clearly show that our method outperforms the other methods on the TUM-MLS dataset. The TUM-MLS data was collected in Germany, where the road marking standard is different from the Chinese standard. Because the extraction model was not retrained with the TUM-MLS dataset, a decline in the precision and recall rates was observed.

5.1.3. Underground road scene

We tested our method on an underground garage scene of about 2000 m^2. The quality of the data, attained from a backpack system, is lower than the quality of the data from the RIEGL VMX-450 system. The point density of the garage data from the backpacked system is lower than the density of the MLS data. However, the issue of lower density can be ignored by gridding point clouds. We calculate the average intensity value within each grid cell to represent the intensity value of the grid cell. CNN considers not only intensity contrast, but also shape information. The results in Fig. 13 indicate the proposed method performs well on the underground garage scene. In Fig. 13 right, red pixels are ground truth and black pixels are extraction results. Although a few zebra crossings and arrow markings are missing, most road markings were extracted correctly.

5.2. Hierarchical road marking classification

Listed in Table 3 are several parameters with values used in multi-scale clustering. These parameters are selected according to the actual situation of the road markings. We obtained the appropriate thresholds, T_s, T_l and D, by measuring the distance between different road markings and observed the minimum length of the lane line to determine L. Because road marking standards vary from country to country, these parameters, when applied to different data, must be adjusted, and the network must be retrained.

The results of the proposed hierarchical road marking classification are evaluated by an error rate, defined as follows:

$$Error \quad rate = \frac{N_{error}}{N_{marking}} \tag{8}$$

where N_{error} is the number of error pixels and $N_{marking}$ is the number of road marking pixels. We tested our method using urban road, highway, and underground road data. As shown in Table 4, the average error rate of road marking classification is 4.07% for three test scenes.

5.2.1. Urban road scene and highway scene

Road marking classification for an urban road scene is shown in Fig. 14. This example is a typical urban road, containing lane lines, dashed lines, text, and arrow markings, etc. Different road markings are labelled with different colors. Erroneous results are enclosed in red boxes. Most road markings can be correctly classified from the classification results. Incorrect results are mainly some lane lines and arrow markings. The continuous lane lines, broken during the previous road marking extraction, look very similar to dashed lines. Thus, some lane lines are incorrectly classified as dashed lines. Similarly, because of errors in the extraction process, the arrow markings are classified as dashed lines. However, this problem can be alleviated by road marking completion. In addition, Fig. 15 shows a highway scene. The error rate of road marking classification is 2.14% and 6.01% for urban road and highway test scenes, respectively. In Fig. 14 and Fig. 15, different road markings are labeled with different colors. Dashed lines 1 and 2 have different lengths. Texts 1, 2, 3 and 4 are different Chinese characters. Road arrows “1” are straight arrows. Road arrows “2” are turn arrows.

We also tested our classification algorithm on the TUM-MLS dataset. Because this dataset does not provide any road marking labels, the classification model was not trained by the TUM-MLS dataset. Also, the shape of the road markings varies greatly from country to country. Large-size road markings are classified correctly by the multi-scale

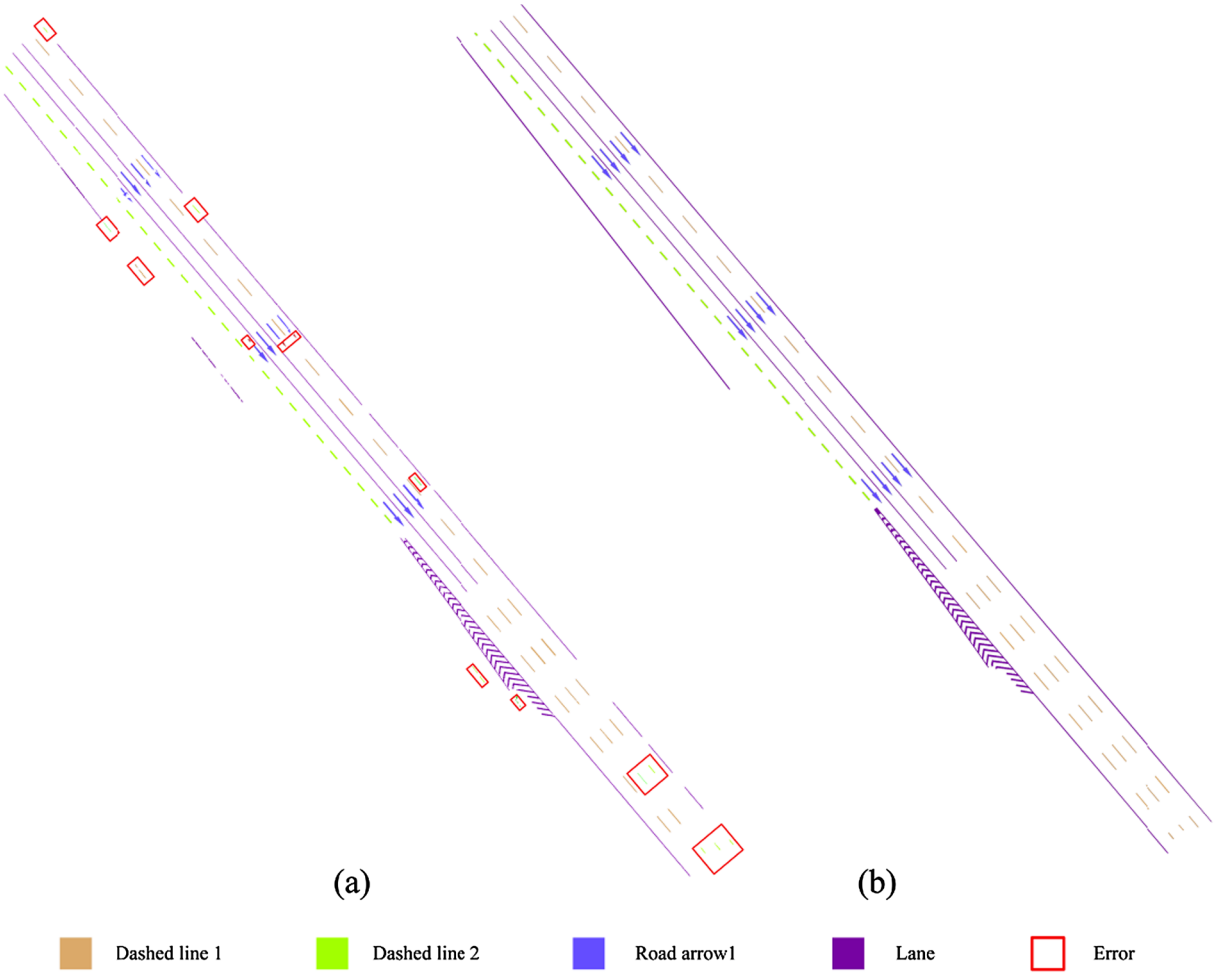


Fig. 15. Road marking classification on a highway road. (a) Classification results. (b) Ground truth.

clustering algorithm. However, the CNN classifier, which is used for small-size road marking classification, does not perform well.

5.2.2. Underground road scene

Fig. 16 shows the classification results with underground data. In general, our method, although also performing well with the underground data example, produces incorrect results when classifying some incomplete road markings. Shown as errors enclosed by red boxes in Fig. 16, a group of zebra crossings, an arrow marking and some parking lines are classified as dashed lines, because while extracting road markings, these were not completely extracted. They were judged as some small-size road markings by the multi-scale clustering algorithm and were sent to the CNN classifier. The classifier considered them all independent dashed lines. The error rate of road marking classification is 3.79% for underground garage scene.

5.3. Road marking completion result

In the road marking completion step, some incomplete road markings are completed. First, some small markings (e.g. arrow markings) are handled by cGAN. Then, based on their continuity, the broken lane lines are connected. Last, some missing markings (e.g. dashed lines) are added into a road, based on the regularity of the positional relationship between road markings. Fig. 17 shows the results of the completion experiment on scenes A and B. We evaluated this experiment by calculating the precision and recall of extraction and the error rate of classification after completion. The performance is shown in Table 5.

As indicated in Table 5, the performance metrics have changed. For the example in Fig. 17(a), extraction precision increased from 92.23% to 93.38%; recall increased from 82.60% to 83.87%; the error rate of classification dropped from 2.41% to 0. For the example in Fig. 17(b), the precision of road marking extraction increased from 90.30% to 90.77%; the error rate of classification dropped from 6.01% to 4.31%; but the recall of extraction dropped from 87.56% to 86.04%. As seen in Fig. 17(b), some incorrectly classified lane lines were filtered out during the completion process. The recall of road marking extraction represents the completeness of the extraction result. Therefore, in the example in Fig. 17(b), the completeness of the extraction result is reduced because of the decrease in the number of road markings. Also, the completion results, using only the cGAN model, are given in Table 5. For both scene A and B, the precisions are improved by introducing the context-based completion. The recall values drop slightly because more points are introduced into the results.

5.4. Computational efficiency

We ran our programs on a desktop PC with an Intel® E5-2683 CPU @ 2.00 GHz and Nvidia Titan X GPU. We divided all the data into 300-meter segments. The average extraction, classification, and completion time per data segment are 3.25 s, 132.12 s, and 2.47 s, respectively. More than 90% of the runtime is consumed on the multi-clustering algorithm. However, under the acceleration of the GPU, the networks are quickly executed.

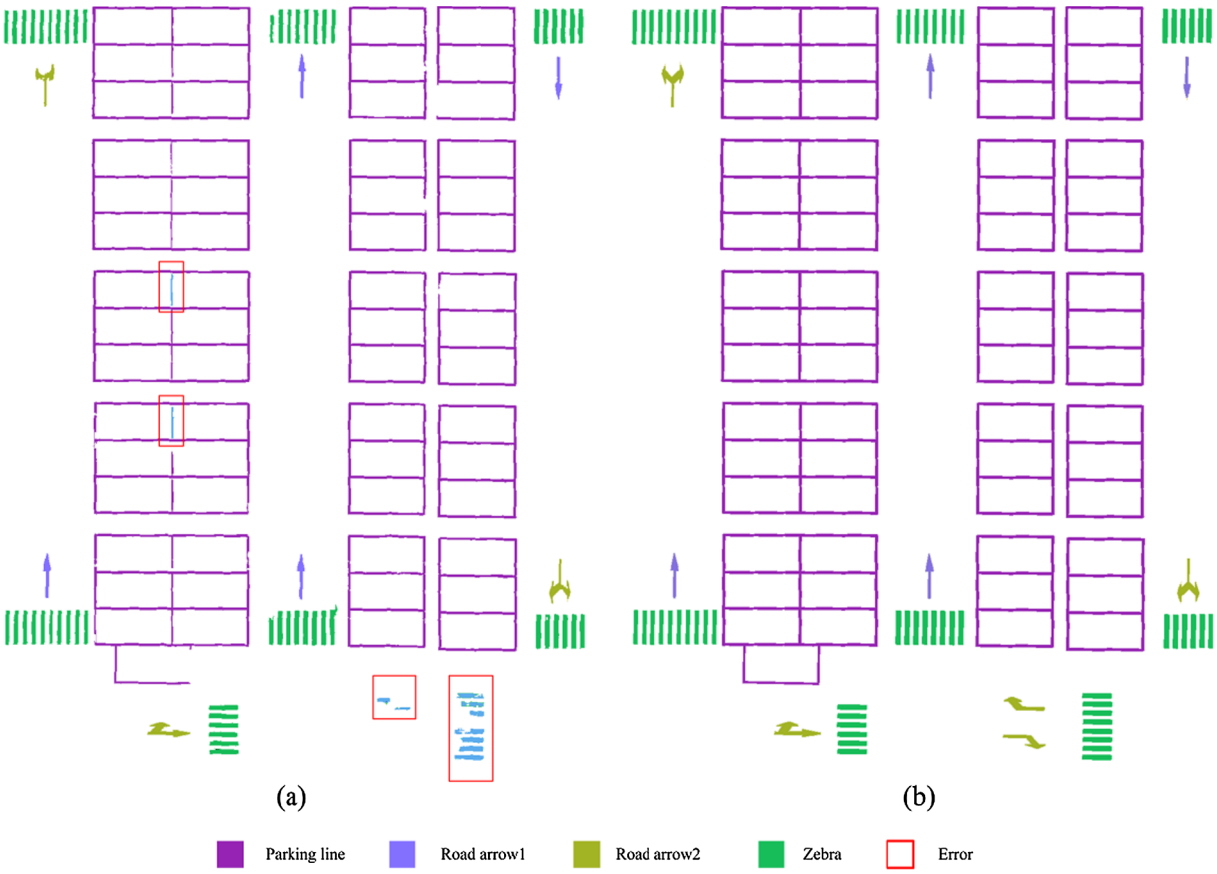


Fig. 16. Road marking classification on underground road. (a) Classification results. (b) Ground truth.

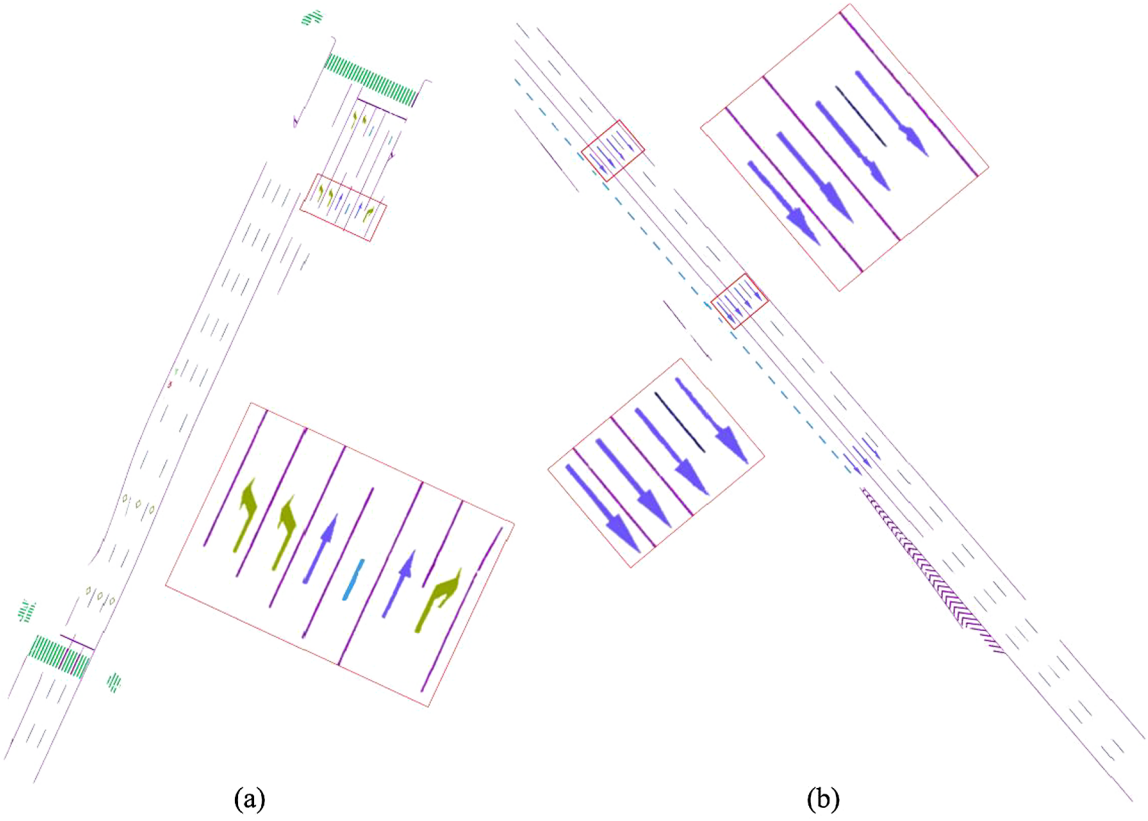


Fig. 17. Results of road marking completion. (a) Scene A. (b) Scene B.

Table 5
Completion performance.

Scene	Precision			Recall			Error rate		
	Before	cGAN	cGAN + context	Before	cGAN	cGAN + context	Before	cGAN	cGAN + context
A	92.23%	92.95%	**93.38%**	82.60%	**84.21%**	83.87%	2.41%	0.53%	**0.00%**
B	90.30%	90.47%	**90.77%**	87.56%	**88.21%**	86.04%	6.01%	4.96%	**4.31%**

6. Conclusions

This paper deals with the problems associated with the threshold-based road marking extraction algorithms. Such problems lead to lack of robustness when handling 3D dense point clouds acquired by MLS systems, most notably because of its varying reflective intensity and point density as well as low intensity contrast between road markings and its surrounding road surface. In this paper, we have proposed a three-stage approach to automated extraction, classification, and completion of road markings in noisy, unstructured, 3D dense MLS point clouds. At the extraction stage, we developed a neural network-based algorithm for automated extraction of road markings. Our experiments demonstrated that our method was able to extract hidden features (e.g. intensity changes, marking shape information) automatically with considerably improved performance in road marking extraction. The precision, recall, and F1-score obtained using the low-quality point cloud datasets achieved 95.97%, 87.52% and 91.55%, respectively.

At the classification stage, we have developed a hierarchical algorithm to handle road markings with different sizes. Unlike those methods mainly based on manually designed classification rules, we have taken advantage of unsupervised multi-scale clustering to obtain large-size road markings (e.g., lane lines, pedestrian crossings), while distinguish small-size road markings using a novel supervised CNN-based classifier. The average error rate of road marking classification that we obtained was 4.07%.

At the completion stage, we have developed a conditional GAN-based algorithm to reduce the effect of the incompleteness of road markings in point clouds. Our results have shown that the reconstruction of even small-size road markings looks very promising. Furthermore, we have also developed a context-based algorithm to handle lane lines and missing road markings. Our results indicated that the recall was increased, while the precision was decreased slightly. Overall, our experimental results demonstrated very promising performance in extraction, classification, and completion of road markings in point clouds. The focus of our future work will be placed on the development of end-to-end algorithms for road marking extraction, classification, and completion.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grants No. 41471379, 61771413, and U1605254) and the Fundamental Research Funds for the Central Universities (Grant No. 20720170047).

References

Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T., 2003. Computing and rendering point set surfaces. IEEE Trans. Visual Comput. Graphics 9 (1), 3–15.
Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C., Evans, T.R., 2001. Reconstruction and representation of 3D objects with radial basis functions. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, pp. 67–76.
Cheng, M., Zhang, H., Wang, C., Li, J., 2017. 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.
Doria, D., Radke, R.J., 2012. Filling large holes in lidar data by inpainting depth gradients. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 65–72.
Figueiredo, R., Moreno, P., Bernardino, A., 2017. Automatic object shape completion from 3D point clouds for object manipulation. In: Proceedings of the 12th International Joint Conference on Computer Vision, pp. 565–570.
Friedman, S., Stamos, I., 2012. Online facade reconstruction from dominant frequencies in structured point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1–8.
Gehrung, J., Hebel, M., Arens, M., Stilla, U., 2017. An approach to extract moving objects from MLS data using a volumetric background representation. ISPRS Ann. Photogram., Remote Sens. Spatial Inf. Sci. 4, 107–114.
Guan, H., Li, J., Yu, Y., Ji, Z., Wang, C., 2015. Using mobile LiDAR data for rapidly updating road markings. IEEE Trans. Intell. Transp. Syst. 16 (5), 2457–2466.
Guan, H., Li, J., Yu, Y., Wang, C., Chapman, M., Yang, B., 2014. Using mobile laser scanning data for automated extraction of road markings. ISPRS J. Photogramm. Remote Sens. 87, 93–107.
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.
Kroemer, O., Amor, H.B., Ewerton, M., Peters, J., 2012. Point cloud completion using extrusions. In: Proceedings of International Conference on Humanoid Robots, pp. 680–685.
Lai, P.J., Huang, Y.L., Chien, S.Y., 2016. Surface-based background completion in 3D scene. In: IEEE Global Conference on Signal and Information Processing, pp. 1218–1222.
Law, A.J., Aliaga, D.G., 2011. Single viewpoint model completion of symmetric objects for digital inspection. Comput. Vis. Image Underst. 115 (5), 603–610.
Lee, S., Kweon, I.S., Kim, J., Yoon, J.S., Shin, S., Bailo, O., Kim, N., Lee, T.H., Hong, H.S., Han, S.H., 2017. VPGnet: vanishing point guided network for lane and road marking detection and recognition. In: Proceedings of the International Conference on Computer Vision, pp. 1965–1973.
Li, Y., Zheng, Q., Sharf, A., Cohen-Or, D., Chen, B., Mitra, N.J., 2011. 2D–3D fusion for layer decomposition of urban facades. In: Proceedings of the International Conference on Computer Vision, pp. 882–889.
Ma, L., Li, Y., Li, J., Wang, C., Wang, R., Chapman, M.A., 2018. Mobile laser scanned point-clouds for road object detection and extraction: a review. Remote Sens. 10 (10), 1531.
Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., Seidel, H.P., 2003. Multi-level partition of unity implicits. ACM Trans. Graphics 22 (3), 463–470.
Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE Trans. Syst., Man, Cybernetics 9 (1), 62–66.
Pan, X., Shi, J., Luo, P., Wang, X., Tang, X., 2018. Spatial as deep: spatial CNN for traffic scene understanding. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 7276–7283.
Park, S., Guo, X., Shin, H., Qin, H., 2005. Shape and appearance repair for incomplete point surfaces. In: Proceedings of the International Conference on Computer Vision, pp. 1260–1267.
Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A., 2016. Context encoders: feature learning by inpainting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2536–2544.
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.
Savchenko, V., Kojekine, N., 2002. An approach to blend surfaces. In: Vince, J., Earnshaw, R. (Eds.), Advances in Modelling, Animation and Rendering. Springer, London, pp. 139–150.
Sharf, A., Alexa, M., Cohen-Or, D., 2004. Context-based surface completion. ACM Trans. Graphics 23 (3), 878–887.
Soilán, M., Riveiro, B., Martínez-Sánchez, J., Arias, P., 2017. Segmentation and classification of road markings using MLS data. ISPRS J. Photogramm. Remote Sens. 123, 94–103.
Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826.
Thrun, S., Wegbreit, B., 2005. Shape from symmetry. In: Proceedings of the International Conference on Computer Vision, pp. 1824–1831.
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.
Wang, J., Oliveira, M.M., 2007. Filling holes on locally smooth surfaces reconstructed from point clouds. Image Vis. Comput. 25 (1), 103–113.
Wu, T., Ranganathan, A., 2012. A practical system for road marking detection and recognition. In: Proceedings of IEEE Intelligent Vehicles Symposium, pp. 25–30.
Xiao, C., Zheng, W., Miao, Y., Zhao, Y., Peng, Q., 2007. A unified method for appearance and geometry completion of point set surfaces. Visual Comput. 23 (6), 433–443.
Xu, S., Georghiades, A., Rushmeier, H., Dorsey, J., McMillan, L., 2006. Image guided

geometry inference. In: IEEE Symposium on 3D Data Processing, Visualization, and Transmission, pp. 310–317.
Yan, L., Liu, H., Tan, J., Li, Z., Xie, H., Chen, C., 2016. Scan line-based road marking extraction from mobile LiDAR point clouds. Sensors 16 (6), 903.
Yang, B., Fang, L., Li, Q., Li, J., 2012. Automated extraction of road markings from mobile LiDAR point clouds. Photogramm. Eng. Remote Sens. 78 (4), 331–338.
Yu, Y., Li, J., Guan, H., Jia, F., Wang, C., 2015. Learning hierarchical features for automated extraction of road markings from 3-D mobile LiDAR point clouds. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 8 (2), 709–726.
Zai, D., Li, J., Guo, Y., Cheng, M., Lin, Y., Luo, H., Wang, C., 2018. 3-D road boundary extraction from mobile laser scanning data via supervoxels and graph cuts. IEEE Trans. Intell. Transp. Syst. 19 (3), 802–813.
Zelek, J., Lunscher, N., 2017. Point cloud completion of foot shape from a single depth map for fit matching using deep learning view synthesis. In: Proceedings of the International Conference on Computer Vision Workshop, pp. 2300–2305.
Zheng, Q., Sharf, A., Wan, G., Li, Y., Mitra, N.J., Cohen-Or, D., Chen, B., 2010. Non-local scan consolidation for 3D urban scenes. ACM Trans. Graphics 29 (4), 94–101.