# Joint 2-D–3-D Traffic Sign Landmark Data Set for Geo-Localization Using Mobile Laser Scanning Data

Changbin You, Chenglu Wen, *Senior Member, IEEE*, Cheng Wang, *Senior Member, IEEE*, Jonathan Li, *Senior Member, IEEE*, and Ayman Habib

***Abstract*— This paper presents a framework to build a joint 2-D–3-D traffic sign landmark data set for geo-localization using mobile laser scanning (MLS) data. The MLS data include 3-D point clouds and corresponding multi-view images. First, an integrated method, based on a deep learning network and the retro-reflective properties of traffic signs, is developed to accurately extract traffic signs from MLS point clouds. Next, the semantic and spatial properties of the traffic signs (type, location, position, and geometric characteristics) are obtained. Then, a joint 2-D–3-D traffic sign landmark data set is built, and a semantic-spatial organization graph is used to organize the traffic sign data set. Last, based on the traffic sign landmark data set, a geo-localization method for a driving car is proposed to estimate the driving trajectory. It can be used for auxiliary positioning of autonomous vehicles. Experimental results demonstrate the reliability of our proposed method for traffic sign detection and the potential of building 2-D–3-D traffic sign landmark data set for driving trajectory estimation from MLS data.**

***Index Terms*— Point cloud, multi-view images, mobile laser scanning (MLS), traffic sign, joint 2-D-3-D, geo-localization.**

## I. Introduction

IN MODERN cities, as a part of road transportation systems, traffic signs provide important information about the road and the environment to guide, warn, or regulate the behavior of drivers for safer and easier driving. Also, the information on the traffic signs may provide important cues for understanding complex road environments.

Manuscript received March 22, 2018; revised July 6, 2018 and August 27, 2018; accepted August 28, 2018. This work was supported in part by the National Science Foundation of China under Grant 61771413 and Grant U1605254 and in part by the Fundamental Research Funds for the Central Universities under Grant 20720170047. The Associate Editor for this paper was G. Mao. *(Corresponding author: Chenglu Wen.)*

C. You, C. Wen, and C. Wang are with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University, Xiamen FJ 361005, China, and also with the School of Information Science and Engineering, Xiamen University, Xiamen FJ 361005, China (e-mail: youchangbin1407@163.com; clwen@xmu.edu.cn; cwang@xmu.edu.cn).

J. Li is with the MoE Key Laboratory of Underwater Acoustic Communication and Marine Information Technology, School of Information Science and Engineering, Xiamen University, Xiamen FJ 361005, China, and also with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: junli@xmu.edu.cn).

A. Habib is with the Lyles School of Civil Engineering, Purdue University, West Lafayette, IN 47907-1971 USA (e-mail: ahabib@purdue.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2018.2868168

Geo-localization is a problem of determining the geographic location of each query image. It has a wide range of real-world applications such as target tracking, trajectory estimation, navigation, and provides a potential way for auxiliary positioning of autonomous vehicles, etc. Traditional geo-localization approach is to predict the geo-location of a query image by finding its matching ground-level images with known location [1]. However, most of the places do not have ground-level reference images available. Another alternative [2]–[4] is to utilize 3-D objects or model information for more efficient and accurate localization. Following this thread, we aim to explore the possibility to use 3-D objects as landmarks on the street for geo-localization. Compared with other 3-D objects on the street, traffic sign has some good features which are suitable for this application. First, traffic sign is stably located and is always in a fixed location for a long time. Second, traffic sign is distributed with spatial uniformity. For traffic safety, traffic sign should be distributed evenly in geographical position. Last, traffic sign is built discretely and isolated without severe occlusion to ensure to be discerned and cognized exactly by drivers. Thus, there is a potential, with great challenge, to apply traffic sign as landmark for geo-localization.

Several groups have established traffic sign image benchmarks, such as GTSDB/RB [5], [6], BelgiumTSD/TSC [7] and Tsinghua-tencent [8]. These datasets contain tens of thousands of labeled traffic sign images captured in various environments. With access to these huge benchmarks, many researchers have achieved excellent performance in detecting traffic signs and classifying them based on a deep learning network [9]–[11]. Shapenet [12] provides a certain number of 3-D traffic sign models in CAD form. However, there is rarely found real-world traffic sign datasets that provide 3-D data in point cloud form and 2-D data in images. A 2-D-3-D dataset that contains both point cloud and image data of traffic sign can be provided for supervised learning or further exploring the correspondent description of traffic sign in data feature. Thus, detecting traffic signs in MLS point cloud/image data for building a joint 2-D-3-D traffic sign dataset is very demanded.

To build such traffic sign dataset for geo-localization, not only the 3-D point clouds with multi-view images but also object sematic and spatial information should be collected. Traditionally, semantic (sign type) and spatial information (geometric feature) of traffic signs is mainly

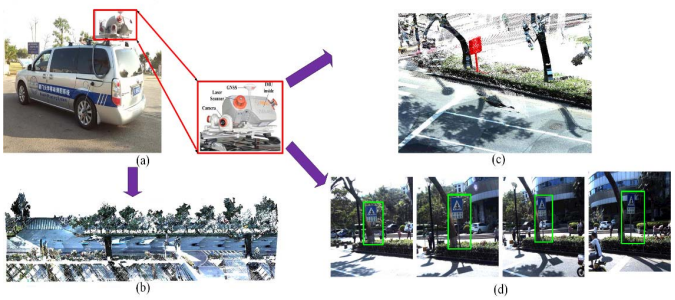



Fig. 1. Illustration of (a) RIEGL VMX-450 system. (b) Large-scale MLS point cloud acquired by RIEGL VMX-450 system. (c) Traffic sign in 3-D point clouds region. (d) Multi-view images of a traffic sign.

collected manually. Because of the resulting huge workload, ensuring real-time and accuracy is difficult. In recent years, more (semi-)automated methods have been developed to obtain sign type [8], [13], [14] and geometric information [15], [16] from traffic signs based on image. However, image is sensitive to lighting conditions, angle of view, etc. Furthermore, accurate spatial geometric information, an important component of traffic sign characteristics, is difficult to calculate from images. In addition, accurate spatial position and geospatial relations among the same or different types of traffic signs cannot be obtained from images.

Light Detection and Ranging (LiDAR) data have been used for applications and research in many ITS applications [17]–[20]. Among the LiDAR systems, Mobile Laser Scanning (MLS), used for 3-D city modeling, provides a cost-effective solution to capture geospatial point clouds with high precision. Because of the high-density, long-range, and high-efficiency of the data acquired by MLS, the system is used to detect and inspect traffic signs for inventory [21], or detect objects like car, pedestrian in real-time for autonomous driving [22]–[24]. Fig. 1 shows a MLS system (RIEGL VMX-450 system), and MLS point cloud with multi-view images acquired by the system. However, for traffic sign detection based on MLS data, current methods, like shape-based [25] or retro-reflective-based [26] methods, are suitable only for dealing with traffic signs in good condition. In our previous works, the retro-reflective properties of sign board [27], [28] and shape-like characteristic [21] were utilized to roughly detect the signs and then multi-condition filtering (e.g. height, evaluation, etc.) was used to finely detect the signs. In [29], a visual phrase dictionary was generated from training data to construct bag-of-visual-phrases representations (BoVPs). Detecting traffic signs was then achieved based on the similarity measures between the BoVPs of the query object and the segmented semantic objects. This method performs well on signs in good condition but cannot work on signs with challenging conditions like deformed shape and tilted or fallen situations caused by human or natural disasters.

In this paper, we propose a framework to build a 2-D-3-D traffic sign landmark dataset and provide a method of geo-localization based on the landmark dataset. The flowchart of the proposed framework is shown in Fig. 2. First, to accurately extract traffic signs from the MLS data, we develop an integrated sign detection method based on a deep learning network and the retro-reflective properties of traffic signs. Next, the semantic and spatial properties of the traffic signs, including type, location, position and geometric characteristics, are obtained. Then, with these properties, a joint 2-D-3-D traffic sign landmark dataset and an organization graph is built for geo-localization. Driving trajectory estimation is provided as a geo-localization test based on the traffic sign landmark dataset.

The contributions of this paper are summarized as follows:

(1) A geo-localization method is proposed to estimate the driving trajectory based on a joint 2-D-3-D semantic-spatial traffic sign landmark dataset. The driving trajectory is estimated by concatenating the image geo-location points obtained by Single-Photo Resection (SPR) with the captured traffic sign image and the correspondent point cloud. The dataset, which includes 3-D point clouds, correspondent multi-view 2-D images, sign types, locations, position and geometric characteristics, can be a potential localization reference and provides a possible solution of auxiliary positioning for autonomous vehicles.

(2) To build traffic sign landmark dataset accurately and rapidly, a robust and rapid traffic sign detection method is developed. Proposed detection method can handle with traffic signs in poor pose condition (e.g. tilted and deformed) and challenging image scenarios. Based on the detection results transferred from traffic sign detection in images, traffic signs with poor poses in point clouds can be successfully detected. Based on the detection results using the retro-reflective properties in point clouds, the mis-detected traffic signs in images due to poor illuminations or partial occlusions can be successfully detected.

The remainder of the paper is organized as follows: Section II reviews the related work. Section III details the proposed traffic sign detection method and the description of the semantic and spatial properties of the detected traffic signs. Section IV introduces our 2-D-3-D traffic sign landmark dataset and driving trajectory estimation. Section V discusses the experimental results. Section VI concludes the work.

## II. Related Work

Types of traffic signs and the extraction of spatial information are important issues in road scene understanding. As a reflection of the function of a traffic sign, it is essential that its type be recognized when gathering associated traffic sign information. Most current works are based on images. Nowadays, because of its high accuracy, one of the most popular methods for traffic sign recognition uses a deep learning network [30], [31]. Due to the lack of informative textures in MLS point clouds, it is difficult to directly recognize extracted traffic sign point clouds, unless they have RGB values. In [27], the type of shape of a traffic sign is obtained by using geometric shape property and the 3-D shape context. Riveiro *et al.* [32] converted 3-D space into a raster image and evaluated the polynomial curves for different types of shapes of signs. However, a detailed type is still unknown. Since MLS systems simultaneously capture multi-view image data along with 3-D point cloud data, the type of traffic signs can be recognized with the assistance of images.

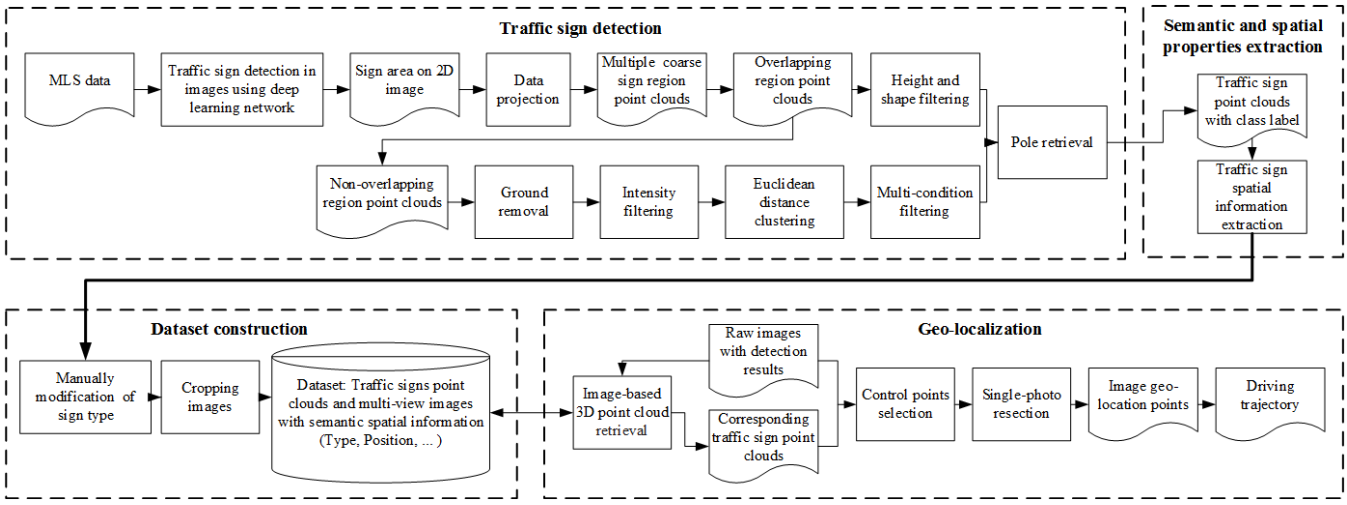



Fig. 2. Flowchart of the proposed framework.

It can be achieved by using classifiers, like the Gaussian-Bernoulli deep Boltzmann machine-based hierarchical [29] and SVM classifier [21], [33], to recognize on-image sign area, which is obtained by projecting the 3-D points of each traffic sign onto a 2-D image. However, only traffic signs in good condition that are detected in point clouds can be correctly recognized.

As for traffic sign spatial information, most works focus only on tilt angle [15] or mutations [16] in images. In addition, some information, like 3-D localization [7], is analyzed further. Spatial information about traffic sign is easy to obtain from MLS data regarding the high precision and high-density 3-D data acquired. Automatic inspection processed have been proposed for acquiring the position and placement of traffic signs, including height of the traffic placement of traffic signs poles, planarity of the traffic sign boards, etc. [21], [33]. Thus, combining 2-D images and 3-D data to provide semantic and spatial information, respectively, is significant for ITS tasks.

Recently, indoor RGB-D datasets, for example NYU Depth v2 [34], SUN RGBD [35] and SceneNN [36], have been introduced into object recognition and scene understanding. These datasets provide images and 3-D point clouds (or water-tight mesh models) of the indoor scenes. 2-D-3-D-S [37] is a 2-D-3-D semantic indoor dataset with several different modalities of images, like RGB, depth and semantically annotated 3-D meshes, and point clouds. Compared with outdoor scenes, indoor scenes have smaller areas with controlled lighting. A 2-D-3-D benchmark (PASCAL3-D+ [38]), which provides 2-D-3-D alignments to 12 rigid categories with each category having more than 3,000 object instances, was built to detect 3-D objects in the wild. ObjectNet3-D [39] is a larger scale 2-D-3-D dataset consisting of 100 object categories, including both indoor and outdoor objects. Although these datasets align 2-D objects in images with 3-D CAD shapes and have semantic annotations, they lack the pose and geometric characteristics of each specific object. For different applications, such as street object inventory, object retrieval, and street scene understanding in autonomous driving, spatial information about street objects is important. Building a real-world 2-D-3-D dataset that provides both 3-D point cloud data and 2-D images, especially specific spatial information about each object, is of great significance.

## III. Proposed Framework

### A. Traffic Sign Detection Using Point Cloud and Images

In this section, we present a combination of method based on the deep learning network and retro-reflectivity for traffic sign detection. Fig. 3 shows an illustration of the proposed detection process. The original point clouds are firstly partitioned into several blocks due to the large size of the point cloud data. Then, based on the trajectory data, the 3-D points that are farther than $d$ (20 meters in the paper) from the MLS device are filtered out. In this way, only the objects on both sides of the lane where the MLS device is driven are retained for further processing. Thethreshold $d$ is chosen based on the specific road width. Here, a traffic sign is defined as a highly retro-reflective vertical plane.

In recent years, several public traffic sign image benchmarks have shown superior performance for classification and detection of traffic signs with deep learning methods. In this paper, we use the benchmark, Tsinghua-Tencent 100K [8], which contains 100,000 high resolution traffic sign images annotated with class labels, bounding boxes, and pixel masks, to train two neural networks, YOLOv3 [40] and FCN model [8] (for comparison) for traffic sign detection. Using the above trained models, we detect traffic signs in multi-view images captured by four high-resolution cameras mounted on a RIEGL VMX-450 MLS system (Fig. 1).

After traffic sign detection in images, the proposed method consists of next three steps:

**(1) Coarse traffic sign region transferred from images:** As shown in Fig. 3(a), a series of bounding box results of traffic sign are obtained using detection network. Once an image has any detected traffic signs, we assume traffic signs exist in the neighborhood of the location where the image is captured. To obtain coarse region of those results in point clouds, we first define the neighborhood as a circle with radius $r$ around the image location in the XY-plane.

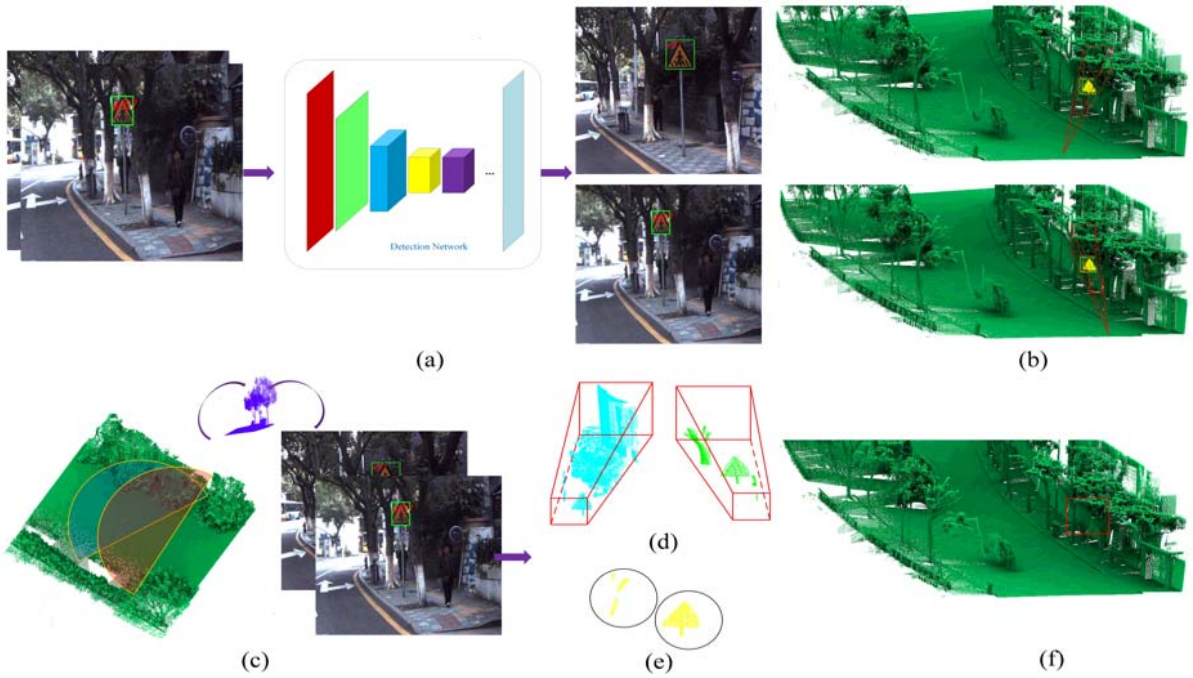



Fig. 3. Illustration of traffic sign detection. (a) Traffic sign detection in images. (b) View of two images in point clouds. (c) Projected points in half circle range onto the images to find coarse sign region point clouds. (d) Coarse traffic sign region point clouds. (e) Traffic sign and other object within overlapping region. (f) Non-overlapping region.

Then, points in point clouds within this circular range are projected onto the image according to the corresponding relationships between the multi-view images and the point clouds. In practice, depending upon the camera direction, only half of the points within the circle need to be projected (see Fig. 3(c)). With the intrinsic and extrinsic parameters given in the MLS system, the corresponding relationships are obtained. A 3-D point in the world coordinate system is transformed into a point in the image coordinate system by applying:

$$s\tilde{m} = A\,[R|t]\,\tilde{M} \tag{1}$$

where $\tilde{M} = [X_w, Y_w, Z_w, 1]^T$ and $\tilde{m} = [u, v, 1]^T$ represent the homogeneous coordinate of a point in world frame and in image frame, respectively; $s$ is a scaling factor; $A$ and $[R|t]$ are the intrinsic and extrinsic camera parameter matrix, respectively. Then, the points, which can be projected onto the bounding box in the image, are regarded as potential points of the traffic sign board. These points are clustered as a coarse traffic sign region, as shown in Fig. 3(d).

**(2) Detection in overlapping regions:** Considering that the multi-view images are acquired simultaneously, a traffic sign can be detected in more than one image. Thus, with multiple detections of the same traffic sign, we obtain more than one coarse area of the sign from different views (Fig. 3(d)). If two or more coarse areas of a potential traffic sign are determined, it is possible to detect accurate potential traffic signs in point cloud data. The overlapping region is extracted by obtaining the indices of points that are projected onto bounding box in the images more than once. After processing all images, overlapping regions with true traffic signs and few other objects are obtained (see Fig. 3(e)). In this way, traffic signs, including those in poor pose conditions within these overlapping regions will be detected. Then, after height and shape filtering for separated clusters, accurate traffic signs are extracted.

**(3) Detection in non-overlapping regions:** After detection in overlapping regions, most of the traffic signs can be extracted. However, in real situations, some traffic signs captured in a single view, and other types of traffic signs, cannot be detected in the images because of a lack of training samples. To this end, we use a retro-reflectivity-based method, which is straightforward in point cloud but with strict constraints, to extract the remaining traffic signs from the non-overlapping regions (see Fig. 3(f)). The non-overlapping regions are obtained by removing overlapping regions from original point clouds.

A voxel-based ground removal method [41] is used first to remove the ground points. Next, the points with an intensity value lower than a threshold, $\omega_c$, are filtered out and an Euclidean distance clustering method is applied to partition the remaining points into separated clusters.

Finally, based on the prior knowledge of traffic signs, thefollowing four conditions are adopted to filter out the objects, except for traffic sign boards: 1) Hit count filtering. A cluster is removed if the number of points, with an intensity value larger than $\omega_I$, is less than $N_I$; 2) Elevation filtering. The difference of the z-coordinates between the segment centroid and the ground points must be at least $H_e$; 3) Height filtering. The difference of the z-coordinates between the highest and lowest points of a cluster must be larger than $H_c$; 4) Shape filtering. Considering that a traffic sign board is normally a flat object, for each cluster, a Principal Component Analysis (PCA) is performed on the covariance matrix of its points. Then, Eigenvalues ($\lambda_1 > \lambda_2 > \lambda_3 > 0$) are used to

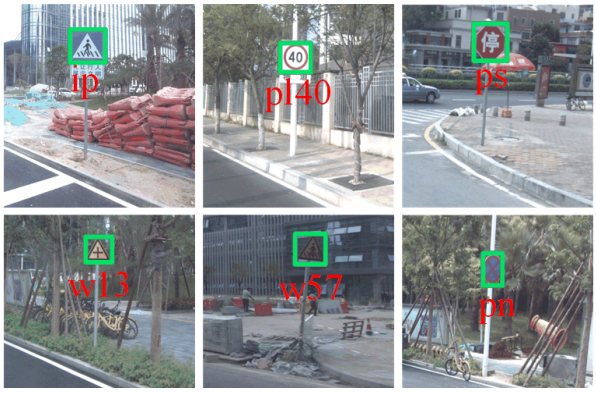



Fig. 4. Illustration of detected traffic signs with bounding box and class label.

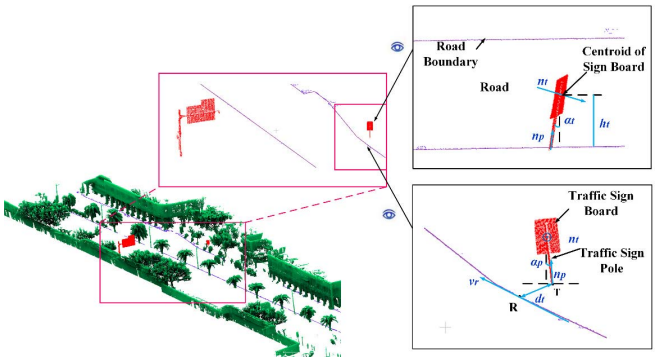



Fig. 5. Illustration of traffic sign spatial properties measurement.

compute linearity as $a_{1D} = (\sqrt{\lambda}_1 - \sqrt{\lambda}_2)/\sqrt{\lambda}_1$ and planarity as $a_{2d} = (\sqrt{\lambda}_2 - \sqrt{\lambda}_3)/\sqrt{\lambda}_1$. If $a_{1D}$ is larger than $S_l$, and $a_{2d}$ is smaller than $S_p$, the cluster cannot be a sign board and is filtered out.

After filtering, a traffic sign pole is retrieved along the downward direction of the extracted traffic sign board. However, a traffic sign may adhere to other objects after retrieving the pole. In this case, to set them apart, we introduce a voxel-based normalized cut segmentation method [29]. The weights on the edges of the complete weighted graph also introduce intensity features of the voxels as follows:

$$w_{ij} = \begin{cases} \exp\left(-\frac{\left\|p_i^{XY}-p_j^{XY}\right\|_2^2}{\sigma_{XY}^2}\right)\cdot\exp\left(-\frac{\left|p_i^{Z}-p_j^{Z}\right|^2}{\sigma_z^2}\right) \\ \quad\cdot\exp\left(-\frac{\left|I_i^n-I_j^n\right|^2}{\sigma_I^2}\right), & \text{if } \left\|p_i^{XY}-p_j^{XY}\right\|_2 \le d_{XY} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $p_i^{XY} = (x_i, y_i)$ and $p_j^{XY} = (x_j, y_j)$ are the coordinates of the centroids on the XY plane. $p_i^Z = z_i$ and $p_j^Z = z_j$ are the $z$ coordinates of the centroids. $I_i^n$ and $I_j^n$ are the interpolated normalized intensities of the points in voxels $i$ and $j$, respectively. $\sigma_{XY}^2$, $\sigma_z^2$ and $\sigma_I^2$ are the variances of the horizontal, vertical and intensity distributions, respectively. Restraining the maximum valid horizontal distance between two voxels is a distance threshold, $d_{XY}$.

### *B. Traffic Sign Semantic and Spatial Properties*

*1) Semantic Property-Sign Type:* Traffic sign semantic property, that is, traffic sign type label, are acquired by a Traffic Sign Recognition (TSR) process. Previous works on TSR using point cloud data are usually based on image recognition using different classifiers after projecting the point cloud of detected signs onto images. In our method, the type of traffic sign point cloud is analogously obtained based on image recognition. We first detect the traffic sign regions in images based on network model. Simultaneously, the sign type is obtained. As shown in Fig. 4, a bounding box with a unique label, which represents the sign type, surrounds the detected traffic sign.

Based on the recognition results of the multi-view images, a sign type is obtained by the types of labels from those images. However, because of the differences in distance or illumination, the model may recognize a traffic sign in a different image as a different type. If all types of labels in multi-view images are randomly transferred to a point cloud, a traffic sign may be given multiple labels. As a result, the type of label of a point cloud is not unique. Considering that recognition performance becomes less reliable with an increase in distance between the location of the traffic sign and the LiDAR sensor, we first re-assign the original score of the probability of the label type for each image, based on distance. Then, to select the most likely type of label for a point cloud, the following selection formulas are used to balance two determining factors: the number of times a single type of label is repeated and the score of the probability of that type of label:

$$\hat{L} = argmax\, W_{L_i} \tag{3}$$

$$W_{L_i} = \omega_1 \frac{N_{L_i}}{\Sigma_1^n N_{L_i}} + \omega_2 \frac{Avrs_{L_i}}{\Sigma_1^n Avrs_{L_i}} \tag{4}$$

where $L_i$ denotes the possible type of label, $i$; $\hat{L}$, computed as the argmax of a weighted function $W_{L_i}$, is the most likely type of label in a point cloud; $N_{L_i}$ is the number of times label type, $L_i$, is repeated. $AvrS_{Li}$ is the average score for the repeated type of labels, $L_i$; $\omega_1$ and $\omega_2$ are the weighted values of the two factors.

Multiple signs, with more than one type in a sign board, cannot be dealt with in this way. In practice, each part of a type is separated in the detection stage and then merged after assigning type labels to all the parts. In addition, traffic signs, which cannot be recognized in the images, are temporarily classified together as unknown types of signs.

*2) Spatial Properties:* Based on our previous work [21], the following parameters (see Fig. 5) are used to describe the spatial properties of the traffic signs:

- **Location:** Location is defined as the coordinates of the centroid of the bottom ring for a traffic sign pole;
- **Position:** (1) the horizontal distance ($d_t$) between the traffic sign locating point (point $T$) and the road boundary point closest to the traffic sign in the horizontal plane (point $R$); (2) the horizontal angle ($\alpha_d$) between the tangent vector of point $R(v_r)$ and the normal vector of the traffic sign board ($n_t$).

- **Geometric characteristics:** (1) height ($h_t$) of the traffic sign above the ground, defined as the height of the centroid of the traffic sign board over the ground; (2) the inclination angle, $\alpha_t$, included between the distribution direction ($n_p$) of the traffic sign pole and the vertical direction with respect to the orientation of the traffic sign board; (3) the inclination angle, $\alpha_p$, included between the distribution direction ($n_p$) of the traffic sign pole and the vertical direction with respect to the profile of the traffic sign board.

Additionally, the planarity of the traffic sign, which is measured by the standard deviation of the laser points on the traffic sign board, can also be calculated. Also, using the combination of features extracted from a traffic sign point cloud and the corresponding image, the visibility of a traffic sign can be estimated as follows [42]:

$$V_s = \frac{1}{N}\sum_{t=0}^{N-1}\sum_{d=1}^{D} w_d M_d(f^t) \tag{5}$$

$$w_d = (w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8) \tag{6}$$

$$\sum_{i=1}^{i=8} w_i = 1 \tag{7}$$

where $f$ is a feature vector that integrates the spatial-related features and the image features, $N$ represents number of scene, $M_d(f)$. $w_d$ is a positive weighted vector for the basis function vector.

## IV. Traffic Sign Landmark Dataset for Geo-localization

### A. Traffic Sign Landmark Dataset Construction

Using the proposed framework, joint multi-view images and 3-D point clouds of traffic signs with semantic and spatial properties were obtained from our data collections.

For some mis-recognized types of signs and traffic signs detected based on point cloud only, the types of some traffic signs are false or unknown. For signs with false or unknown type, we projected the point clouds onto images to obtain the corresponding images, and then manually identified the sign type according to the images. These images were cropped automatically to keep only traffic sign regions. Then, images that are blurred, or contain only the backs of signs, are eliminated. After that, a number was assigned to both the point cloud and the corresponding images to determine the association between them. Each traffic sign point cloud contains more than one multi-view image showing the front of the sign. Next, a traffic sign landmark dataset with 2-D images and 3-D point clouds was built. As shown in Fig. 6, semantic and spatial properties of each traffic sign are associated with 2-D images and a 3-D point cloud, resulting in a complete 2-D-3-D traffic sign landmark dataset. Based on this dataset, the simplified organization graph can be constructed.

Considering the reset or position change of some traffic signs, which lead to changes in semantic and spatial properties of the traffic signs in the dataset, the dataset needs to be updated regularly. If a traffic sign, which has been changed or reset, is explicit, the point clouds, images and spatial properties can be obtained by equipment like static laser scanner, camera and total station. A large-scale update of traffic signs may occur if the roads are rebuilt. In this situation, a new MLS scanning process is needed to accurately update the dataset. The traffic sign dataset can be considered as an independent basic library once it is built using our proposed process. Thus, a new scan for the traffic signs is irrelevant to the dataset.

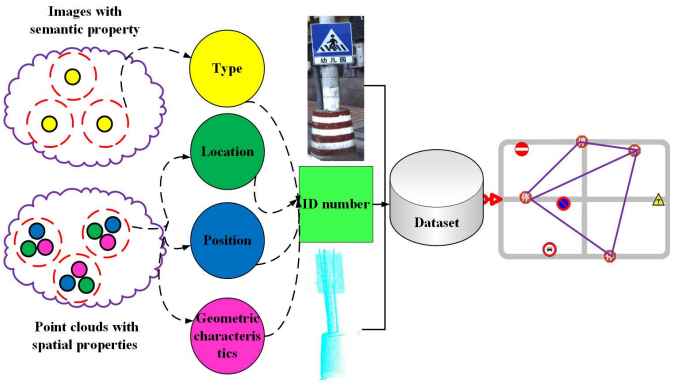



Fig. 6. Illustration of simplified process of a traffic sign dataset and organization graph construction.

### B. Geo-Localization Based on Traffic Sign Landmark Dataset

The traffic sign landmark dataset can be further used for geo-localization of a car with a dashboard camera. Assuming a car is driven on an unknown road, obtaining the accurate geo-location is difficult if based only on the speed and direction of the car, or on the GPS, for which the margin of error is about five to ten meters. But, with some real-world references on the road, the precise location of the car is more likely to be achieved. To estimate the trajectory of the car driven on a road where we have collected traffic sign data, we propose using the images captured by the dashboard camera and the traffic sign landmark dataset. The geo-location of the car at any given time is regarded as the image geo-location captured by the dashboard camera at that time. Thus, estimating the trajectory of the car can be converted into a problem of obtaining geo-locations of a series of consecutive images. This problem, known as Single-Photo Resection (SPR), is a traditional photogrammetric problem that is usually solved by relating 2-D image points to corresponding 3-D object points. In the traffic sign landmark dataset, 3-D traffic sign point clouds are real-world coordinate points. Thus, the key to solve this problem is to build a relationship between the images from the dashboard camera and the point clouds in the traffic sign dataset. Because the point cloud and the corresponding multi-view images in the dataset have been assigned the same ID number, the traffic sign point cloud can be retrieved by finding the most similar image for a given image containing the traffic sign object and then selecting control points to calculate the image geo-location point.

The method consists of the following four procedures:

**(1) Traffic sign detection for images:** First, we use images captured from the dashboard camera to detect traffic signs. Once an image contains a traffic sign, from the traffic sign landmark dataset, we can retrieve all traffic sign point clouds

and related images within a circle with radius, *Rs* (20 meters in this paper), around the initial GPS positioning location. After that, each given image has the corresponding search library for the next procedure.

**(2) Image-based 3-D point cloud retrieval:** The goal is to find the 3-D point cloud that is most similar to a given image. Image retrieval is determined based on the similarity of the measurements of the image features. The most common image features include color, shape, and texture features, like RGB, Histogram of Oriented Gradient (HOG), Scale-invariant feature transform (SIFT), etc. Compared with these traditional features, the feature extracted from the convolution network has proved to be effective in many applications like detection and classification. Thus, to increase the accuracy of retrieved results, a public pre-trained network model, ImageNet-VGG-f [43], consisting of eight layers, five of which are convolutional and the last three fully-connected, is used to extract image features. After comparing the feature vector of the given image with the feature of each image in the search library, the traffic sign point cloud of the given image is found from among the N returned point clouds and corresponding images. Since the point clouds and images in the search library are from the neighborhood of the initial GPS positioning location, many unnecessary interferences have been filtered out, which greatly improves the accuracy of the retrieved results and reduces the retrieval time.

**(3) Calculating geo-location of an image:** With the image and related point cloud, SPR is used to determine the six Exterior Orientation Parameters (*EOP's*) associated with the image, including the image geo-location $(X_o, Y_o, Z_o)$ and three-dimensional rotation $(\varphi, \omega, \kappa)$ of the camera. Using at least three non-collinear conjugate points (control points) in a least squares adjustment based on the collinearity equations, the traditional method takes the following form:

$$x - x_p = -f\frac{a_1(X-X_o)+b_1(Y-Y_o)+c_1(Z-Z_o)}{a_3(X-X_o)+b_3(Y-Y_o)+c_3(Z-Z_o)} \quad (8)$$

$$y - y_p = -f\frac{a_2(X-X_o)+b_2(Y-Y_o)+c_2(Z-Z_o)}{a_3(X-X_o)+b_3(Y-Y_o)+c_3(Z-Z_o)} \quad (9)$$

where $x_p$, $y_p$ are the coordinates of the principal point; $f$ is the focal length of the camera; $a_i$, $b_i, c_i$ (i = 1, 2, 3) are the elements of the rotation matrix $R = R_\varphi R_\omega R_\kappa = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}$. In our method, we use a collinearity equations-based approach, where control points are selected interactively. The control points of the traffic sign point cloud, initially set as the corner points of the 3-D bounding box of the sign board and lowest point of the pole, are then modified manually. The corresponding control points in the image are selected accordingly. For a series of control points, the optimal $p$ for the *EOP's* is computed as the argmax of an error function $E(p)$ that is accumulated by inserting the coordinates of the control points into the collinearity equations point by point:

$$E(p) = \sum_{i=1}^{n} \left\| x_i - x_o + f\frac{u_i}{w_i} \right\|^2 + \left\| y_i - y_o + f\frac{v_i}{w_i} \right\|^2 \quad (10)$$

$$p^* = arg\min_p E(p) \quad (11)$$

TABLE I
DESCRIPTIONS OF THE THREE SELECTED DATA COLLECTIONS

| Data collections | Size (GB) | Length(km) | Point number (million) | Image number |
|---|---|---|---|---|
| HDR | 14.89 | 5.03 | 571 | 3011 |
| WPR | 14.06 | 4.45 | 522 | 2572 |
| XHR | 33.70 | 11.13 | 1030 | 8220 |

TABLE II
PARAMETERS USED IN METHOD

| $r$ | $\omega_c$ | $\omega_l$ | $N_l$ | $H_e$ (m) | $H_c$ (m) |
|---|---|---|---|---|---|
| 30 | 40000 | 55000 | 285 | 1.5 | 0.7 |
| $S_l$ | $S_p$ | $\sigma_{XY}^2$ (m) | $\sigma_z^2$ (m) | $\sigma_l^2$ | $d_{XY}$ (m) |
| 0.85 | 0.11 | 2 | 10 | 1.0 | 5 |

where $n$ is the number of control points; $x_i$, $y_i$ represent the coordinates of the image point $i$ that corresponds to the object point $(X_i, Y_i, Z_i)$. $u_i, v_i, w_i$ represent $a_1(X_i - X_o) + b_1(Y_i - Y_o) + c_1(Z_i - Z_o)$,$a_2(X_i - X_o) + b_2(Y_i - Y_o) + c_2(Z_i - Z_o)$ and $a_3(X_i - X_o) + b_3(Y_i - Y_o) + c_3(Z_i - Z_o)$, respectively. With one of the Quasi-Newton methods (BFGS), the optimal $p$ for the *EOP's* is solved by minimizing the error function $E(p)$.

**(4) Result check:** Because of the possibility that results may be erroneously retrieved, it may be incorrect to consider, as the final result, only the geo-location, which is calculated using the most similar traffic sign point cloud with the given image. Thus, once the geo-location is determined, it should be checked by the initial GPS positioning location. If the error between the geo-location and the GPS positioning location is greater than a preset threshold (4 meters in 2-D Euclidean space in this paper), the next similar point cloud should be iteratively calculated until the image geo-location is within the allowable range of error. For a series of the image geo-location points, the trajectory of the car is estimated by linearly concatenating the trajectory points.

## V. EXPERIMENTS AND RESULTS

### A. System and Data

In this study, MLS data, including point clouds and images, were acquired by a RIEGL VMX-450 system (Fig. 1(a)). The accuracy of the acquired point clouds is within 8 mm, and the precision is within 5 mm. To obtain traffic sign landmarks by our traffic sign detection method, three data collections were selected from the following sources: the surveys of Huandao Road (HDR), Wenping Road (WPR) and Xiahe Road (XHR), Xiamen, China, conducted in December 2013, August 2016, and September 2016, respectively. These roads represent a typical urban road environment with a considerable number of traffic signs. The XHR survey, which revealed many fallen trees, was acquired after the attack of Typhoon Meranti. In XHR survey, some traffic signs, which are deformed or tilted, are a great challenge for traffic sign detection. A detailed description of these three data collections is given in Table I.

### B. Traffic Sign Detection

*1) Parameter Analysis:* Table II gives the parameters used in our traffic sign detection method. Table III lists the influence

TABLE III
INFLUENCE OF EACH PARAMETER ON THE RESULTS

| Possible influence | $r$ | | $\omega_c$ | | $N_I$ | | $H_e$ | | $H_c$ | | $S_l$ | | $S_p$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $S$ | $L$ | $S$ | $L$ | $S$ | $L$ | $S$ | $L$ | $S$ | $L$ | $S$ | $L$ | $S$ | $L$ |
| Precision decreasing | | | | | ✓ | | ✓ | | ✓ | | | ✓ | ✓ | |
| Recall decreasing | ✓ | | ✓ | ✓ | | ✓ | | ✓ | | ✓ | ✓ | | | ✓ |
| Time consuming | | ✓ | | | | | | | | | | | | |
| Completeness decreasing | | | | ✓ | | | | | | | | | | |

TABLE IV
PRINCIPAL COMPONENT ANALYSIS OF SIGN BOARDS

| Property | Maximum | Minimum | Average |
|---|---|---|---|
| Linearity | 0.853 | 0.116 | 0.455 |
| Planarity | 0.860 | 0.115 | 0.477 |

of each parameter on the results if it is set too small ($S$) or too large ($L$).

The radius of projection circular range, $r$ is set, based on the largest distance between the MLS camera and a traffic sign that can be detected in an image. The coarse point cloud region of detected traffic sign in image may cannot be transferred from image if $r$ is set too small. If $r$ is set too large, it will take too much time to project points onto image to find coarse traffic sign region. In this paper, we set $r$ to be 30 meters.

Intensity filtering value, $\omega_c$ is set, based on the intensity values of the traffic sign points. If $\omega_c$ is set too small, it will be difficult to partition the remaining points into separated object clusters after the initial intensity filtering. Some traffic sign boards may be mis-detected because they clustered with other objects, resulting in matching the filtering conditions. If $\omega_c$ is set too large, points with not-high intensity in sign board will be filtered out, causing the incompleteness of the sign board and mis-detecting. In this paper, we set $\omega_c$ to be 40000.

Intensity threshold, $\omega_I$ is usually set about 50000 to 60000 depending on the high intensity value of points on sign board. In this paper, we set $\omega_I$ to be 55000. The number of points with an intensity value larger than $\omega_I$, $N_I$ is set according to the number of minimum sized boards. If $N_I$ is set too small, some false objects with few high intensity points will be retained. If $N_I$ is set too large, some sign boards have only few high intensity points due to retro-reflective material deterioration, may be filtered out. In this paper, we set $N_I$ to be 285.

Elevation threshold, $H_e$ and height threshold, $H_c$ are determined directly based on local traffic facility construction standards. Once $H_e$ and $H_c$ are set too small, some objects like traffic cone and license plate are falsely detected as sign board. In addition, low elevation traffic signs will be filtered out if $H_e$ is set too large. If $H_c$ is set too large, triangle signs whose board height is short will be filtered out as well. In this paper, we set $H_e$ and $H_c$ to be 1.5 and 0.7 meters, respectively.

After analyzing the principal component of 197 traffic sign boards, as shown in Table IV, linearity threshold, $S_l$ is set as the maximum value of linearity, and planarity threshold, $S_p$ is set as the minimum value of planarity. If $S_l$ is set too small, it may miss some big sign boards whose linearity is large, or if $S_l$ is set too large, light pole or trunk may be erroneously regarded as board. If $S_p$ is set too small, it may retain some board-like objects, or if $S_p$ is set too large, it may miss some small signs whose planarity is small. In this paper, we set $S_l$ and $S_p$ to be 0.85 and 0.11, respectively.

In addition, the variances of the horizontal, vertical and intensity distributions, respectively, $\sigma_{XY}^2$, $\sigma_z^2$, $\sigma_I^2$, and the distance threshold restraining the maximum valid horizontal distance between two voxels, $d_{XY}$, the optimal parameter configurations used in the voxel-based normalized cut segmentation method are set according to [29].

Thus, these parameters should be anatomized and reasonably set before experiments. However, they are not fixed values in any context. For point clouds collected using different systems, $\omega_c$, $\omega_I$ and $N_I$, which are related to the attributes of density and intensity, should be reset accordingly. For traffic signs with different design standards, $H_e$, $H_c$, $S_l$ and $S_p$ should be modified accordingly.

*2) Quantitative Assessment:* To quantitatively assess the accuracy of the proposed traffic sign detection method, detection results were given using the three selected data collections. We compared the extracted traffic signs with the manually labeled reference data and adopted the following four indices as follows:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{12}$$

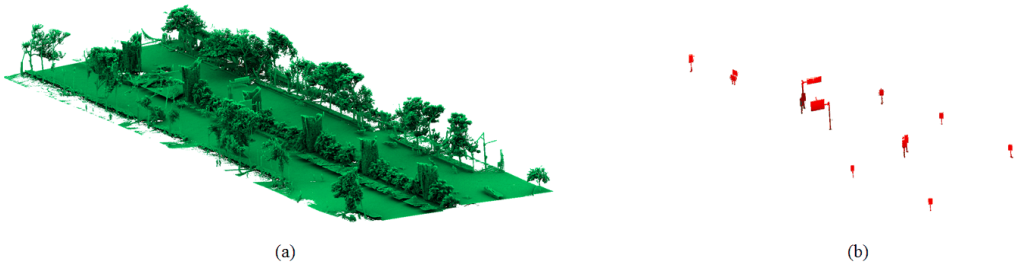
$$\text{Precision} = \frac{TP}{TP + FP} \tag{13}$$

$$\text{Quality} = \frac{TP}{TP + FN + FP} \tag{14}$$

$$F1 \text{ measure} = \frac{2 * TP}{2 * TP + FN + FP} \tag{15}$$

where TP, FN, and FP denote the numbers of true positives, false negatives, and false positives, respectively.

Fig. 7 shows an example of part of the traffic sign detection results in 3-D point clouds. Details of the quantitative evaluation results are given in Table V. The average precision, recall, F1-measure, and quality of the proposed traffic sign detection algorithm achieved (YOLOv3-based/FCN-based) using the three selected data collections are 0.949/0.940, 0.931/0.963, 0.940/0.951, and 0.887/0.907, respectively. The results indicate the proposed method efficiently detects traffic signs using MLS data. Comparing with the FCN-based method, the YOLOv3-based method for traffic sign detection in images has higher accuracy. Thus, after transferring from images, fewer FPs exist in overlapping region point clouds, leading to higher precision of YOLOv3-based method. However, it has lower recall than the FCN-based method for image object detection. Once traffic sign results cannot be transferred from images, the detection result is from retro-reflective-based method,



(a) (b)

Fig. 7. Illustration of a part of traffic sign detection results. (a) Raw point cloud. (b) Detected traffic signs.

TABLE V
QUANTITATIVE EVALUATION OF DIFFERENT TRAFFIC SIGN DETECTION METHODS

| Dataset | Method | Ground truth | TP | FN | FP | Precision (%) | Recall (%) | F1-measure (%) | Quality (%) |
|---|---|---|---|---|---|---|---|---|---|
| **HDR** | Ours(YOLOv3-based) | 138 | 131 | 7 | 8 | **94.24** | 94.93 | 94.58 | 89.73 |
| | Ours(FCN-based) | | 136 | 2 | 10 | 93.15 | **98.55** | **95.77** | **91.89** |
| | IB | | 136 | 2 | 14 | 90.67 | **98.55** | 94.44 | 89.47 |
| | PB | | 130 | 8 | 12 | 91.55 | 94.20 | 92.86 | 86.67 |
| | GB | | 134 | 4 | 16 | 89.33 | 97.10 | 93.06 | 87.01 |
| **WPR** | Ours(YOLOv3-based) | 125 | 115 | 10 | 6 | **95.04** | 92.00 | 93.50 | 87.79 |
| | Ours(FCN-based) | | 120 | 5 | 9 | 93.02 | 96.00 | **94.49** | **89.55** |
| | IB | | 123 | 2 | 15 | 89.13 | **98.40** | 93.54 | 87.86 |
| | PB | | 114 | 11 | 10 | 91.94 | 91.20 | 91.57 | 84.44 |
| | GB | | 115 | 10 | 16 | 87.79 | 92.00 | 89.84 | 81.56 |
| **XHR** | Ours(YOLOv3-based) | 266 | 246 | 20 | 12 | 95.35 | 92.48 | 93.89 | 88.49 |
| | Ours(FCN-based) | | 251 | 15 | 11 | **95.80** | 94.36 | **95.08** | **90.61** |
| | IB | | 264 | 2 | 29 | 90.10 | **99.25** | 94.45 | 89.49 |
| | PB | | 243 | 23 | 13 | 95.02 | 93.23 | 94.12 | 88.89 |
| | GB | | 246 | 20 | 21 | 92.13 | 92.48 | 92.31 | 85.71 |
| **KITTI** | Ours(Retro-reflective-based) | 75 | 67 | 8 | 7 | 90.54 | 89.33 | 89.93 | 81.73 |

which is applied directly in point clouds. Thus, YOLOv3-based method achieved lower detection recall than FCN-based. In conclusion, with the advent of faster and more accurate on-image detection networks, transferring detection results from image will attain higher precision and recall.

In addition, the following three typical methods were compared with our method: an intensity-based (IB) method [28], a pole-based (PB) method [21], and a Gaussian Mixture Model-based (GB) method [33]. Shown in bold in Table V are the best experimental results, which indicate that our proposed method outperforms the other methods for detecting traffic signs. Comparatively, the IB method achieved better performance than the PB and GB methods. In the PB method, pole-like objects (linear structures) are extracted first based on a PCA analysis. However, the linearity threshold in PCA cannot be self-adapted and is always pre-defined in accordance with the common thin poles. Thus, the PB method may not be able to deal with a huge pole or pole structure loss.

The large number of reflective objects in the city streets (e.g. planar metallic surfaces and some pedestrian reflective clothing) in our selected datasets strongly influence the precision of the methods based on retro-reflective properties for urban road environment. The PB and GB method uses retro-reflective properties to detect traffic signs. In this case, some sign boards, whose retro-reflective properties are partially lost due to material deterioration, may be erroneously filtered out. Also, in the GB method, a Gaussian Mixture Model with two components was trained to analyze the distribution of the intensity of both non-reflective and retro-reflective points to set them apart. Training samples strongly affect performance. The IB method, especially, is very sensitive to reflective objects. Although the IB method has the highest recall among the compared methods, because of the increase in FPs, there is a decrease in precision and global quality.

To further test the performance of our method, we reconstructed a part of road scene point clouds using a series of single frames and ground truth poses from the odometry benchmark (one of KITTI dataset [44]), which was collected by a Velodyne multi-beam laser scanner. Due to the lack of camera extrinsic parameters, the detection point clouds cannot be transferred from image detection results. Thus, we only applied the retro-reflective-based method to this data. Considering the elevation and size of traffic signs in this dataset are lower and smaller than ours, the parameters $N_I$, $H_e$ and $H_c$ in the experiment were modified to be 150, 1.3, and 0.5, respectively and other parameters remained the same. As show in Table V, the precision, recall, F1-measure, and quality in KITTI dataset are 0.905, 0.893, 0.899, and 0.817, respectively. Thus, our method also achieved satisfying results on KITTI dataset.

*3) Traffic Sign Detection in Challenging Conditions:* To test our traffic sign detection method under challenging conditions, three scenes with traffic signs in poor pose condition (Fig.8(a)) were selected from the XHR dataset (collected after Typhoon Meranti). Most of the existing sign damage is deformation of the sign boards. In our previous work [45], we demonstrated the feasibility of our method. The detection results of the traffic signs with challenging 3-D form, generated by our

TABLE VI
COMPUTING TIME OF EACH PROCESSING STEP AT THE TRAFFIC SIGN DETECTION STAGE

| Method | Dataset | Steps | | | | Total computing time (min) |
|---|---|---|---|---|---|---|
| | | Detection in images (s) | Coarse traffic sign region transferring from images (s) | Detection in overlapping region (s) | Detection in non-overlapping region(s) | |
| YOLOv3-based | HDR | 91.19 | 511.60 | 18.23 | 183.13 | 13.40 |
| | WPR | 78.40 | 588.93 | 25.95 | 194.23 | 14.79 |
| | XHR | 250.45 | 791.15 | 45.53 | 313.22 | 23.34 |
| FCN-based | HDR | 18878.43 | 1374.40 | 18.87 | 160.78 | 340.54 |
| | WPR | 16593.24 | 1571.74 | 71.03 | 180.65 | 306.94 |
| | XHR | 52585.63 | 3419.27 | 95.56 | 285.16 | 939.76 |

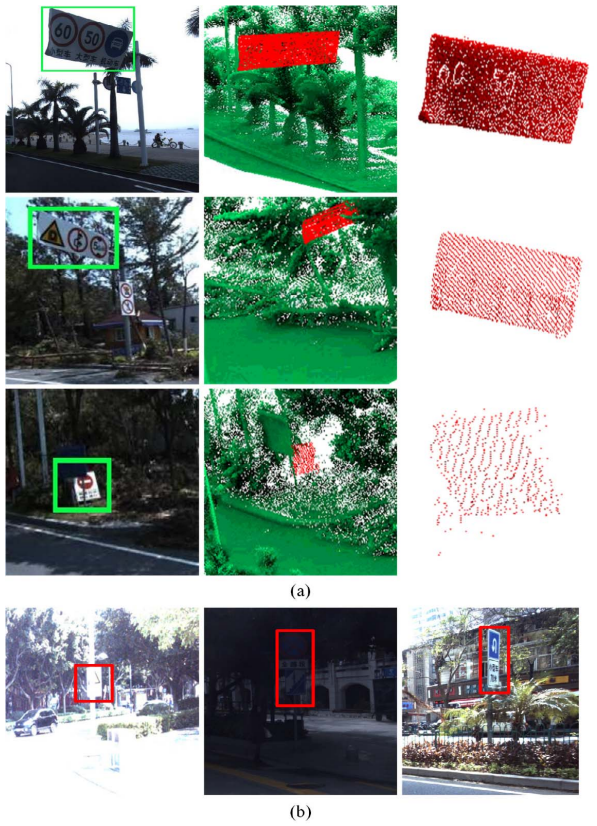


Fig. 8. Some examples of traffic sign in challenging conditions detected. (a) Challenging 3-D form. Images of traffic signs in poor pose conditions (left side), Point clouds of traffic signs in poor pose conditions (middle) and detection results (right side). (b) Challenging image scenarios: strong illumination (left side), poor illumination (middle) and large viewpoint (right side).

detection method, are given in the right side of Fig. 8(a). The results indicate our proposed method performs well and achieves good results for traffic sign detection under challenging 3-D forms. These traffic signs in poor pose conditions are difficult to be detected in point clouds because of retro-reflective material and geometrical property loss. With the combination of traffic sign detection results in images, which are not associated with the conditions of traffic sign in 3-D form, our method effectively deals with the failure detection of traffic signs in point clouds.

In addition, we considered the effects of challenging image scenarios such as strong illumination, poor illumination and large viewpoint (shown in Fig. 8(b)) in traffic sign detection performance. It is still very challenging to effectively detect traffic signs in above image scenarios using image-based methods. However, since point clouds are more immune to environmental conditions, we can detect those traffic signs directly based on the retro-reflective properties in point clouds. The two methods are complementary. Thus, the integrated method is robust to traffic sign detection for different situations.

TABLE VII
LABELING RESULTS OF TRAFFIC SIGN POINT CLOUDS

| Dataset | Accuracy (%) (Without selection formulation) | Accuracy (%) (With selection formulation) |
|---|---|---|
| HDR | 89.44 | 91.55 |
| WPR | 87.13 | 90.10 |
| XHR | 91.93 | 95.97 |
| Average | 89.50 | 92.15 |

*4) Time Performance:* In our experiments, we used a trained YOLOv3 (and FCN for comparison) for traffic sign detection in images with a Linux PC with an Intel Core (TM) i5-4460 CPU and two NVIDIA Titan Z GPUs with 12GB memory. To reduce the time cost of detecting images, we partitioned the images in each dataset into four groups and ran them in parallel. The obtaining image detection results were then used for transferring coarse traffic sign region in point clouds, which is the next step of the remaining proposed framework. The remaining proposed framework, coded with C++, was run on a personal computer configured with an Intel(R) Core (TM) i7-6700 CPU 3.4 GHz and a RAM of 16 GB.

As shown in Table VI, the transfer of coarse traffic sign regions from images required most of the total processing time except detection in images. Total computing time (not including data acquisition time) for traffic sign detection (YOLOv3-based and FCN-based) in the HDR, WPR, and XHR datasets were about 13.40/340.54, 14.79/306.94, and 23.34/939.76 min, respectively. For YOLOv3-based method, the computing time of detection in images was greatly shorten compared with FCN-based method. The overall computational efficiency of the traffic sign detection was greatly improved.

### C. Sign Type Recognition

For classifying the true type of a sign, the following are of equal importance: two determining factors of selection formulation, times a single type of label type is repeated, and the probability score of the type of label. Thus, the weight value $\omega_1$ and $\omega_2$, are set at 0.5. In practice, the performance of traffic sign recognition mainly depends on the deep learning network and training samples. In [8], network performance is evaluated in detail. Achieved accuracy and

Fig. 9. Illustrations of parts of traffic sign point clouds and related images in our landmark dataset.

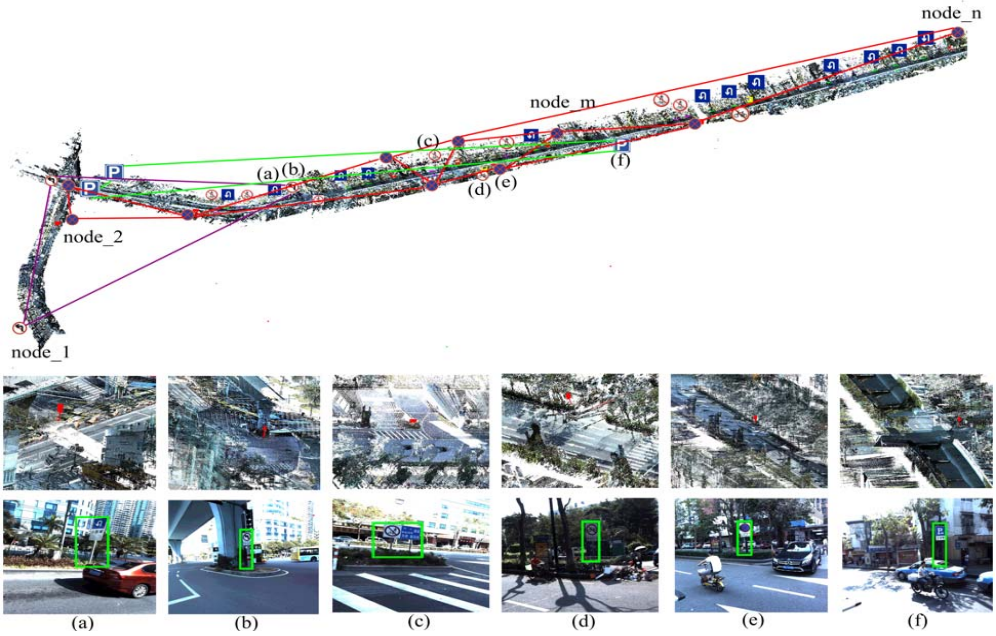



Fig. 10. Illustration of the organization graph with traffic sign landmarks. (a)–(f) Representative traffic signs of each category.

recall are 88.0% and 91.0%, respectively. To evaluate the performance of sign recognition, 335 traffic sign point clouds with signs (except those of unknown type) were selected from the detection results for comparison with true labels from the images.

As shown in Table VII, by reassigning the scores for the types of labels and using the selection formulation, we achieve an average accuracy of 92.15%, which to a certain degree, is an improvement when compared with the result without formulation selection. False labeling occurs when all the images containing same traffic sign, recognized by the network, obtain false label types.

### *D. Dataset Visualization and Organization Graph*

The dataset consists of 39 sign types, 1,306 2-D images captured from different views, and 442 3-D point clouds, along with their semantic and spatial properties. Currently, two parameters of semantic properties, traffic sign planarity and visibility, are not considered in the dataset. Fig. 9 shows some example point clouds and related images in our dataset.

Based on this dataset, an organization graph for traffic signs of the same type was built (see Fig. 10). The node of the graph represents a recognized traffic sign with semantic and spatial attributes. The edge between nodes is the Euclidean distance between the two signs in 3-D space. In conformance with the semantic property, traffic signs of the same type are connected. In conformance with the global spatial property (i.e. the traffic sign location in the real world), the traffic signs are connected to a specific road area. In conformance with the local spatial properties (i.e., traffic sign positions), the relationship of a traffic sign to the local road environment is provided.

The traffic sign landmarks along with semantic and spatial information are exported to a Geographic Information System (GIS) vector layer. Fig. 11 shows the traffic sign landmarks with semantic and spatial properties clearly and intuitively visualized over an orthophoto in GIS under the visual interface. The semantic and spatial properties can be easily acquired by clicking the traffic sign in the interface. With the combination of big traffic data like traffic

TABLE VIII
mAP (%) RESULTS OBTAINED BY USING DIFFERENT SIMILARITY MEASUREMENT DISTANCE IN TWO DATASETS

| Dataset | Euclidean | Manhattan | Chebyshev | Cosine | Hamming | Correlation |
|---|---|---|---|---|---|---|
| Ours | 57.70 | 57.63 | 44.39 | **59.95** | 38.76 | 59.57 |
| [46] | 34.08 | 37.52 | 21.03 | **43.27** | 18.33 | 43.26 |

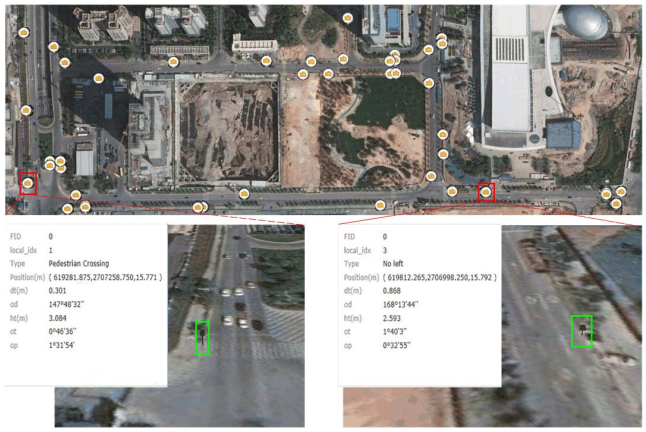


Fig. 11. Traffic sign landmarks with semantic and spatial propeties visualized over an ortophoto in GIS.

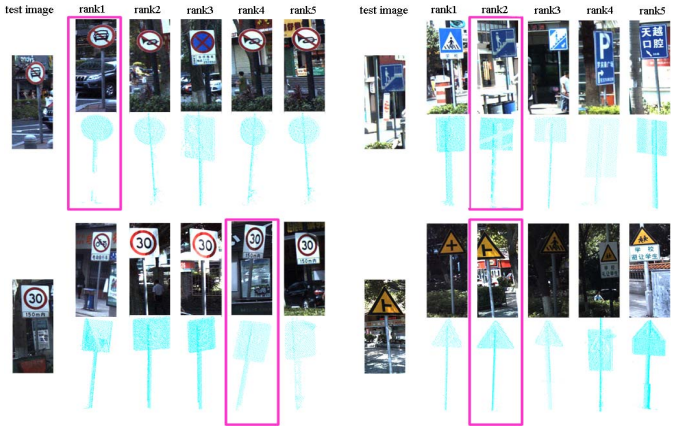



Fig. 12. Example of 3-D traffic sign point cloud retrieval. Pink boxes are the true point cloud.

flows or traffic accidents, it might be useful for analyzing whether the distribution distance between two same type traffic signs is reasonable, to guide road designation, or standardization installation and maintenance update of traffic signs. For example, if a road section with few speed limit signs or other warning signs in long distance often has traffic accidents, it would be urgent for traffic management departments to add more related warning signs.

### *E. Image-Based 3-D Point Cloud Retrieval in Dataset*

To find the optimal similarity measurement distance for image retrieval, we chose 28 categories, about 415 images with different street views, from our dataset as the search library. Also, we used another dataset [46], containing 40 categories of animals, with 100 images in each category to seek the optimal similarity measurement. In addition, half of the images from each category were randomly selected as retrieval images to test the measurement of several distances. We used mean average precision (mAP) to judge the performance of their retrieval. The mAP is computed as follows:

$$\text{mAP} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{R}\sum_{r=1}^{R} Precision(M_{nr}) \qquad (16)$$

where $N$ is the total number of retrieved images and $R$ is the number of the retrieved images in each category; $M_{nr}$ represents the set of retrieval results ranked from the maximum result to image, $d_r$. Table VIII shows that measuring similar distances differently has a great impact on the retrieval results. Because of its larger search library, performance in [46] is worse than for our dataset. Among these distances, the cosine distance has the highest mAP: 59.95% and 43.27% for our dataset and [46], respectively. Thus, cosine distance is used as our measurement of the similarity in distance for further retrieval tasks. Due to the difficulty in matching morphological features between an image and a point cloud, it is impossible

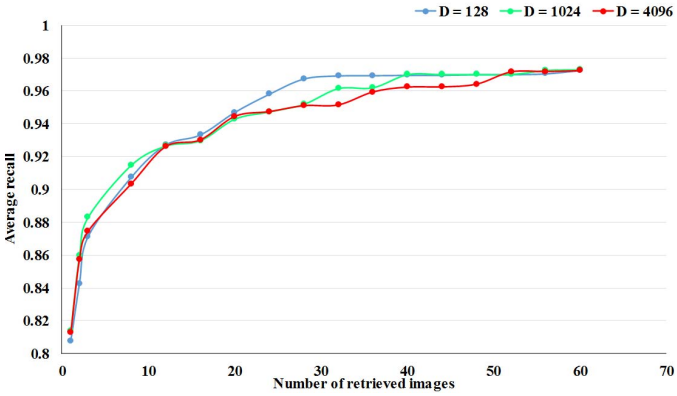



Fig. 13. Average recall using 128, 1024 and 4096 dimensions feature with different number of retrieved images.

to manually select the most similar among the hundreds of 3-D point clouds that depend only on a single image. To this end, based on the image retrieval method, we provide operator with the top N ranked 3-D point clouds and a related image from our dataset for each randomly captured traffic sign image from a street. Then, the operator selects those that are close from among the N returned point clouds. Fig. 12 shows some 3-D point cloud retrieval examples using the image-based retrieval method.

To evaluate the performance of the method, six main categories with 37 subcategories (1195 images) were selected from our dataset. A single traffic sign contains at least two images captured from different views, with the same ID number assigned to each. Once any one of the retrieved top N images has the same ID number as the retrieval image, we conclude that the traffic sign image has successfully retrieved its closest image and 3-D point cloud. Obviously, as the number of retrieved image increases, the recall rises. As illustrated in Fig. 13, using different dimension features of an image

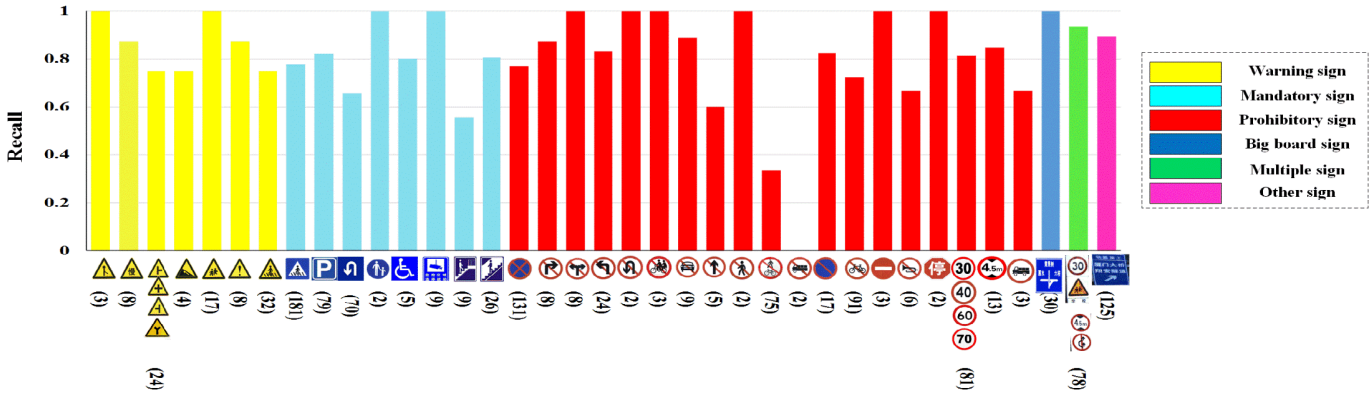



Fig. 14. Recall for 37 categories that have images with a 3-D point cloud in our dataset. The number of 2-D images for each category is shown in the brackets.

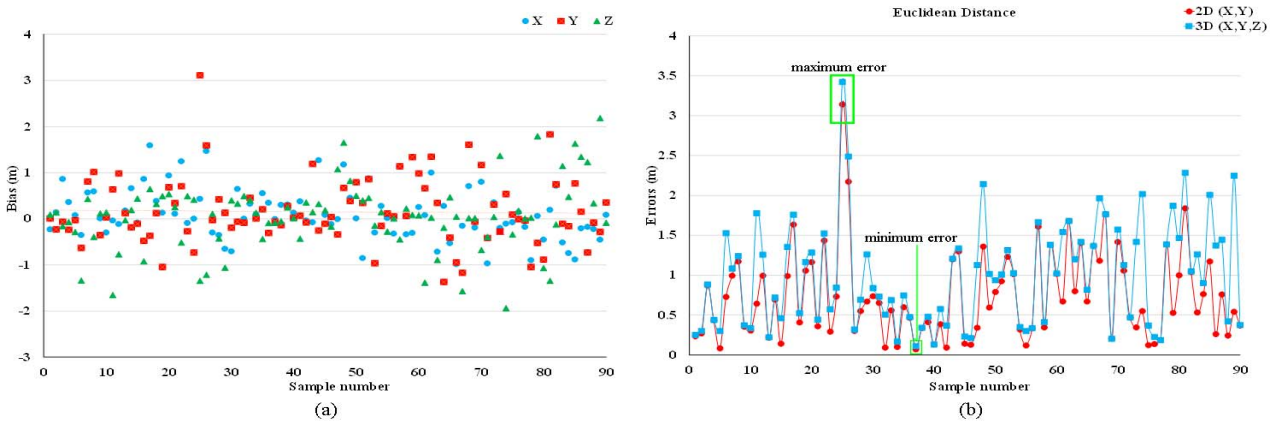



Fig. 15. Illustration of SPR results of 90 sample images compared with ground truth. (a) Biases in the $x$, $y$ and $z$ directions. (b) Distance errors in 2-D and 3-D Euclidean space (Green boxes are maximum and minimum distance errors).

influences the recall rate. Thus, suitable dimension features selected can improve the recall rate and time efficiency of the retrieval method. When using 128 dimensions, the recall reaches 97% with 30 images returned. Taking comprehensive complex operator selection and time efficiency into consideration, we finally used 128-dimension features, reduced from the original output 4096-dimension features by PCA. Then, using each image, we conducted a test to determine if the top retrieved image and point cloud were the same as those in the traffic sign with the retrieved ones. Fig. 14 shows the recall results for each category. Thus, in our dataset, the image-based 3-D point cloud retrieval method attains satisfying results.

### *F. Geo-Localization of a Driving Car*

To demonstrate the feasibility of trajectory estimation, 90 sample images, which are related to 44 traffic sign point clouds, captured in three road parts from constructed traffic sign landmark dataset were selected. The trajectory recorded by MLS system is regarded as the ground truth. Before SPR, *IOP's* (the principal point $x_o$, $y_o$ and the focal length of camera $f$) and initial *EOP's* should be set at first. In general, *IOP's* can be obtained from the instruction manual of camera. As for initial *EOP's*, image geo-location ($X_o$, $Y_o$, $Z_o$) were set as $X_o = \sum_1^n X_i/n$, $Y_o = \sum_1^n Y_i/n$, $Z_o = 0$. Three rotations $(\varphi, \omega, \kappa)$ were all set as 0. In SPR, it only needs one control point to minimize the error function $E(p)$. But the introduction of more control points strengthens the solution of the SPR. Thus, we selected five control points and used four of them each time to find out the image geo-location. It obtained five results. We chose the one having minimum $E(p)$ as result.

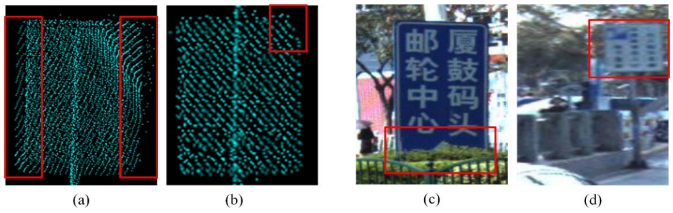



Fig. 16. Some examples of low quality of traffic sign point clouds [(a) and (b)] and images [(c) and (d)] causing difficulty in selecting control points. (a) Rough edges. (b) Missing a corner. (c) Occlusion. (d) Blurred edges.

To assess the accuracy of the experimental results with respect to the ground truth, the biases between the two groups of coordinates were analyzed. As shown in Fig. 15(a), most biases in the $x$, $y$, and $z$ directions were within $\pm$1m, on the 90 selected image geo-locations. Fig. 15(b) shows the distance errors between estimated point and ground truth point in 2-D and 3-D Euclidean space. The distance errors are defined

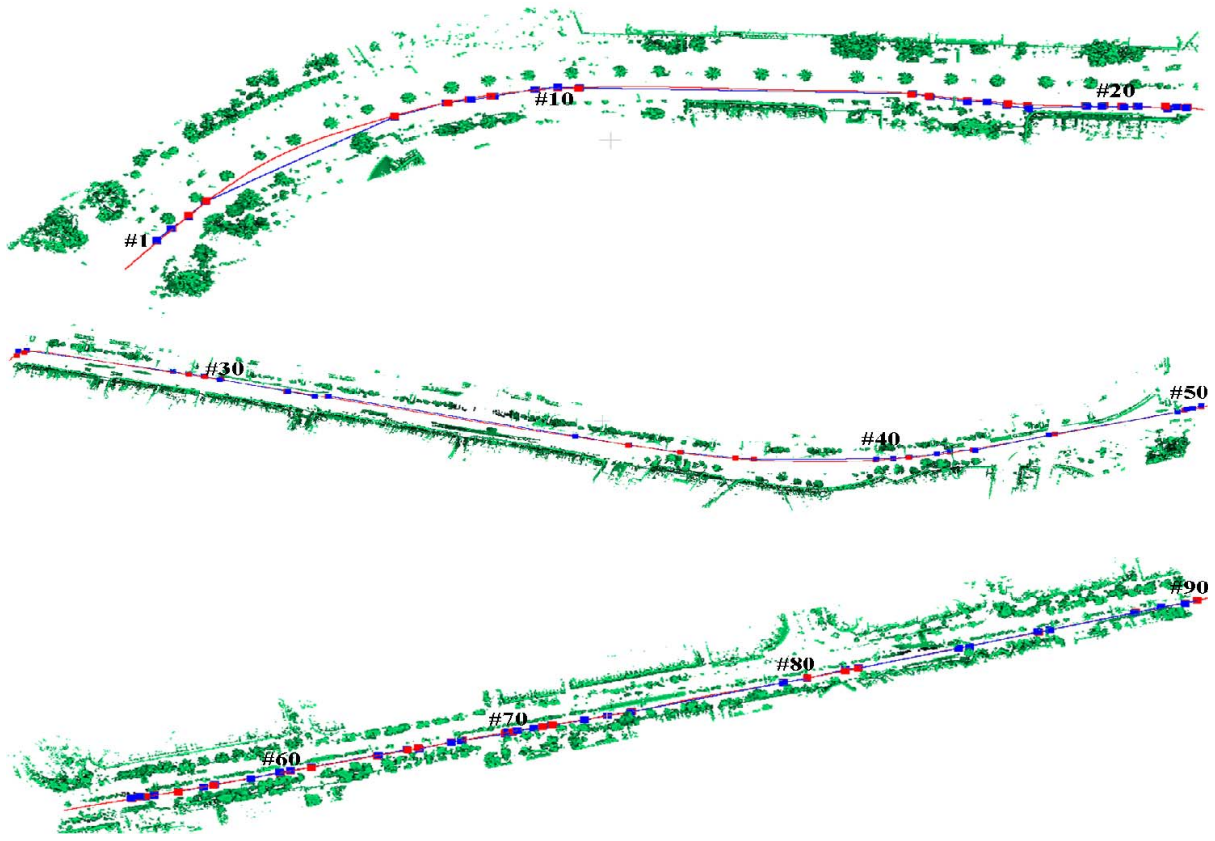



Fig. 17. Illustration of three estimated trajectories (in blue) compared with the baseline trajectories (in red). The squares with different number represent the ground truth (in red) and the estimated (in blue) geo-location of the sample images.

as follows:

$$\text{Error}_{2d} = \sqrt{(x_e - x_g)^2 + (y_e - y_g)^2} \tag{17}$$

$$\text{Error}_{3d} = \sqrt{(x_e - x_g)^2 + (y_e - y_g)^2 + (z_e - z_g)^2} \tag{18}$$

where $x_e$, $y_e$, $z_e$ are the coordinates of the estimated point; $x_g$, $y_g$, $z_g$ are the coordinates of the ground truth point.

In 2-D Euclidean space, it attained an average, maximum, minimum distance errors of 0.733m, 3.138m and 0.065m, respectively. In 3-D Euclidean space, it attained an average, maximum, minimum distance errors of 0.989m, 3.416m and 0.105m, respectively. The maximum distance errors occurred in the image numbered 25. The reason is that the corresponding traffic sign point cloud is with low quality (as shown in Fig. 16(a)), causing difficulty in selecting control points. Therefore, the results of this approach were strongly affected by the quality of point clouds and images. Fig. 16 shows some low-quality examples of traffic sign point clouds and images. Except for images with occlusion (Fig. 16(c)) or blurred edges (Fig. 16(d)), the images captured in night time, rain or misty weather may also lead to the failure of selecting correct control points. In addition, manpower to select control points is time-consuming, low efficiency, and easy to introduce human error. Considering worse accuracy of GPS positioning, the approach for estimating driving trajectory achieved acceptable results.

The estimated trajectory (in blue) and the ground truth trajectory (in red) are given in Fig. 17. Red squares represent ground truth geo-location of sign image, and blue squares represent estimated geo-location of sign image by our method. If a long-curved road only has an estimated point on both sides, such as a part curved road between sample number 1 and 10, the linearly connecting trajectory loses curvature of the curve. On the straight road and curved roads with multiple estimated image geo-location points, the driving trajectories obtained from linearly concatenating the image geo-location points has a good coincidence with ground truth trajectories.

## VI. Conclusions

In this paper, we proposed a novel framework for building a joint 2-D-3-D traffic sign landmark dataset from MLS data for geo-localization. By using the integrated method based on a deep learning network and retro-reflective properties in our method, we achieved experimental results that demonstrate the effectiveness of automated detection and type recognition for traffic sign from MLS data. Using detected traffic sign point clouds aligned with images and the extracted semantic and spatial properties of the traffic signs, we built a 2-D-3-D traffic sign landmark dataset. Also, based on the landmark dataset for an intuitive and effective guide to update the maintenance of traffic signs, we built a semantic-spatial organization graph of traffic signs. In addition, the driving car geo-localization test conducted using the traffic sign landmark dataset demonstrates the potential application to driving trajectory estimation. It shows a great effectiveness and feasibility

in auxiliary positioning of ordinary and autonomous vehicles. In future work, we will expand the landmark dataset and develop an automatic SPR method for more efficient and accurate geo-localization.

## REFERENCES


[1] J. Hays and A. A. Efros, "IM2GPS: Estimating geographic information from a single image," in *Proc. IEEE CVPR*, Jun. 2008, pp. 1912–1920.

[2] S. Ramalingam, S. Bouaziz, P. Sturm, and M. Brand, "Geolocalization using skylines from omni-images," in *Proc. IEEE CVPR*, Sep. 2009, pp. 23–30.

[3] A. Irschara, C. Zach, J.-M. Frahm, and H. Bischof, "From structure-from-motion point clouds to fast location recognition," in *Proc. IEEE CVPR*, Jun. 2009, pp. 2599–2606.

[4] A. Ruta, Y. Li, and X. Liu, "Intelligent Geolocalization in urban areas using global positioning systems, three-dimensional geographic information systems, and vision," *J. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 3–12, Jan. 2010.

[5] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German traffic sign detection benchmark," in *Proc. Int. Joint Conf. Neural Netw.*, 2013, pp. 1–8.

[6] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German traffic sign recognition benchmark: A multi-class classification competition," in *Proc. Int. Joint Conf. Neural Netw.*, 2011, pp. 1453–1460.

[7] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view traffic sign detection, recognition, and 3D localisation," in *Proc. IEEE WACV*, Apr. 2009, pp. 1–8.

[8] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proc. IEEE CVPR*, Jun. 2016, pp. 2110–2118.

[9] J. Zhang, M. Huang, X. Jin, and X. Li, "A real-time Chinese traffic sign detection algorithm based on modified YOLOv2," *J. Algorithm.*, vol. 10, no. 4, p. 127, Nov. 2017.

[10] R. Qian, B. Zhang, Y. Yue, Z. Wang, and F. Coenen, "Robust Chinese traffic sign detection and recognition with deep convolutional neural network," in *Proc. Int. Conf. Natural Comput.*, 2015, pp. 791–796.

[11] M. M. Lau, K. H. Lim, and A. A. Gopalai, "Malaysia traffic sign recognition with convolutional neural network," in *Proc. IEEE Int. Conf. (DSP)*, Jul. 2015, pp. 1–8.

[12] Z. Wu *et al.*, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE CVPR*, Jun. 2015, pp. 1912–1920.

[13] L. Hazelhoff, I. Creusen, and P. H. N. De With, "Robust detection, classification and positioning of traffic signs from street-level panoramic images for inventory purposes," in *Proc. IEEE WACV*, Jan. 2012, pp. 313–320.

[14] R. Qian, Y. Yue, F. Coenen, and B. Zhang, "Traffic sign recognition with convolutional neural network based on max pooling positions," in *Proc. Int. Conf. (ICNC-FSK)*, 2016, pp. 578–582.

[15] J. Abukhait, I. Abdel-Qader, J. S. Oh, and O. Abudayyeh, "Occlusion-invariant tilt angle computation for automated road sign condition assessment," in *Proc. IEEE Int. Conf. (EIT)*, May 2012, pp. 1–6.

[16] L. Hazelhoff, I. Creusen, and P. H. N. de With, "Mutation detection system for actualizing traffic sign inventories," in *Proc. IEEE Int. Conf. (VISAPP)*, Jan. 2014, pp. 705–713.

[17] H. Wang *et al.*, "Object detection in terrestrial laser scanning point clouds based on Hough forest," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 10, pp. 1807–1811, Oct. 2014.

[18] Y. Yu, J. Li, H. Guan, and C. Wang, "Automated extraction of urban road facilities using mobile laser scanning data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2167–2181, Aug. 2015.

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

[20] F. Wu *et al.*, "Rapid localization and extraction of street light poles in mobile LiDAR point clouds: A supervoxel-based approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 2, pp. 292–305, Feb. 2017.

[21] C. Wen *et al.*, "Spatial-related traffic sign inspection for inventory purposes using mobile laser scanning data," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 27–37, Jan. 2016.

[22] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE CVPR*, Jul. 2017, pp. 6526–6534.

[23] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. Waslander. (2018). "Joint 3D proposal generation and object detection from view aggregation." [Online]. Available: https://arxiv.org/abs/1712.02294

[24] Z. Wang, W. Zhan, and M. Tomizuka. (2018). "Fusing bird view LIDAR point cloud and front view camera image for deep object detection." [Online]. Available: https://arxiv.org/abs/1711.06703

[25] B. Yang and Z. Dong, "A shape-based segmentation method for mobile laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 81, pp. 19–30, Jul. 2013.

[26] C. Ai and Y. Tsai, "Critical assessment of an enhanced traffic sign detection method using mobile LiDAR and INS technologies," *J. Transp. Eng.*, vol. 141, no. 5, art. no. 04014096, May 2015.

[27] S. Weng, J. Li, Y. Chen, and C. Wang, "Road traffic sign detection and classification from mobile LiDAR point clouds," in *Proc. ISPRS Int. Conf.*, 2015, p. 9910A.

[28] P. Huang, M. Cheng, Y. Chen, H. Luo, C. Wang, and J. Li, "Traffic sign occlusion detection using mobile laser scanning point clouds," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2364–2376, Sep. 2017.

[29] Y. Yu, J. Li, C. Wen, H. Guan, H. Luo, and C. Wang, "Bag-of-visual-phrases and hierarchical deep models for traffic sign detection and recognition in mobile laser scanning data," *ISPRS J. Photogram. Remote Sens.*, vol. 113, pp. 106–123, Mar. 2016.

[30] D. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural Netw.*, vol. 32, pp. 333–338, Aug. 2012.

[31] Y. Zhu, C. Zhang, D. Zhou, X. Wang, X. Bai, and W. Liu, "Traffic sign detection and recognition using fully convolutional network," *Neurocomputing*, vol. 214, pp. 758–766, Nov. 2016.

[32] B. Riveiro, L. Díaz-Vilariño, B. Conde-Carnero, M. Soilán, and P. Arias, "Automatic segmentation and shape-based classification of retro-reflective traffic signs from mobile LiDAR data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 1, pp. 295–303, Jan. 2016.

[33] M. Soilán, B. Riveiro, J. Martínez-Sánchez, and P. Arias, "Traffic sign detection in MLS acquired point clouds for geometric and image-based semantic inventory," *ISPRS J. Photogramm. Remote Sens.*, vol. 114, pp. 92–101, Apr. 2016.

[34] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 746–760.

[35] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in *Proc. IEEE CVPR*, Jun. 2015, pp. 567–576.

[36] B.-S. Hua, Q.-H. Pham, D. T. Nguyen, M.-K. Tran, L.-F. Yu, and S.-K. Yeung, "SceneNN: A scene meshes dataset with annotations," in *Proc. Int. Conf. 3D Vis.*, 2016, pp. 92–101.

[37] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese. (2017). "Joint 2D-3D-semantic data for indoor scene understanding." [Online]. Available: https://arxiv.org/abs/1702.01105

[38] X. Yu, R. Mottaghi, and S. Savarese, "Beyond PASCAL: A benchmark for 3D object detection in the wild," *Appl. Comput. Vis.*, vol. 12, no. 10, pp. 75–82, Mar. 2014.

[39] Y. Xiang *et al.*, "ObjectNet3D: A large scale database for 3D object recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 160–176.

[40] J. Redmon and A. Farhadi. (2018). "YOLOv3: An incremental improvement." [Online]. Available: https://arxiv.org/abs/1804.02767

[41] Y. Yu, J. Li, H. Guan, C. Wang, and J. Yu, "Semiautomated extraction of street light poles from mobile LiDAR point-clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1374–1386, Mar. 2014.

[42] S. Wu, C. Wen, H. Luo, Y. Chen, C. Wang, and J. Li, "Using mobile LiDAR point clouds for traffic sign detection and sign visibility estimation," in *Proc. IEEE IGARSS*, Jul. 2015, pp. 565–568.

[43] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional networks," in *Proc. BMVC*, Sep. 2014.

[44] A. Geiger, P. Lenzm, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE CVPR*, Jun. 2012, pp. 3354–3361.

[45] C. You, C. Wen, H. Luo, C. Wang, and J. Li, "Rapid traffic sign damage inspection in natural scenes using mobile laser scanning data," in *Proc. IEEE IGARSS*, Jul. 2017, pp. 6271–6274.

[46] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 453–465, Mar. 2014.

**Changbin You** received the B.Eng. degree in electronic and information engineering from Fuzhou University, Fuzhou, China, in 2016. He is currently pursuing the M.Sc. degree with the School of Information Science and Engineering, Xiamen University, Xiamen, China. His research interests include computer vision, machine learning, and 3-D point cloud processing.

**Chenglu Wen** (M'14–SM'17) received the Ph.D. degree in mechanical engineering from China Agricultural University, Beijing, China, in 2009. She is currently an Associate Professor with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Information Science and Engineering Xiamen University, Xiamen, China. She has co-authored about 50 research papers published in refereed journals and proceedings. Her current research interests include 3-D point cloud processing, machine learning, and robot vision. She is the Secretary of the ISPRS WG I/6 on Multi-Sensor Data Fusion (2016–2020). She is an Associate Editor of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS.

**Cheng Wang** (M'11–SM'16) received the Ph.D. degree in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2002. He is currently a Professor and the Associate Dean of the School of Information Science and Engineering and the Executive Director of Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University, Xiamen, China. He has co-authored over 150 papers. His current research interests include remote sensing image processing, mobile LiDAR data analysis, and multi-sensor fusion. He is the Chair of the ISPRS Working Group I/6 on Multi-Sensor Integration and Fusion (2016–2020) and a Council Member of the China Society of Image and Graphics.

**Jonathan Li** (M'00–SM'11) received the Ph.D. degree in geomatics engineering from the University of Cape Town, Cape Town, South Africa. He is currently a Professor with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Information Science and Engineering, Xiamen University, Xiamen, China. He is also a Professor with the Departments of Geography and Environmental Management and Systems Design Engineering, University of Waterloo, Waterloo, ON, Canada. His current research interests include information extraction from LiDAR point clouds and from earth observation images. He has co-authored over 400 publications. He chairs the ISPRS Working Group I/2 on LiDAR-, Air- and Spaceborne Optical Sensing Systems (2016–2020) and the ICA Commission on Sensor-Driven Mapping (2015–2019). He is an Associate Editor of the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS and the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING.

**Ayman Habib** received the M.Sc. and Ph.D. degrees in photogrammetry from The Ohio State University, Columbus, OH, USA, in 1993 and 1994, respectively. He is currently a Professor in geomatics at the Lyles School of Civil Engineering, Purdue University, West Lafayette, IN, USA. His research interests include the fields of terrestrial and aerial mobile mapping systems, modeling the perspective geometry of nontraditional imaging scanners, automatic matching and change detection, automatic calibration and stability analysis of low-cost digital cameras, utilizing low-cost imaging systems for infrastructure monitoring and biomedical applications, incorporating analytical and free-form linear features in various photogrammetric orientation procedures, object recognition in imagery, UAV-based 3-D mapping, and integrating photogrammetric data with other sensors/data sets (e.g., GPS/INS, GIS databases, multispectral sensors, and LiDAR).