Mapping and Semantic Modeling of Underground Parking Lots Using a Backpack LiDAR System

Zheng Gong, Jonathan Li, *Senior Member, IEEE*, Zhipeng Luo, Chenglu Wen, *Senior Member, IEEE*, Cheng Wang, *Senior Member, IEEE*, and John Zelek, *Member, IEEE*

***Abstract*— Presented in this paper is a novel method for the mapping and semantic modeling of an underground parking lot using 3D point clouds collected by a low-cost Backpack Laser Scanning (BLS) or LiDAR system. Our method consists of two parts: a Simultaneous Localization and Mapping (SLAM) algorithm based on Sparse Point Clouds (SPC) and a semantic modeling algorithm based on a modified PointNet model. The main contributions of this paper are as follows: (1) a probability frontend framework for the alignment of point clouds using the local point cloud surface variance as the weight of registration, which modifies registration failure caused by the lack of features in sparse point clouds, (2) a robust submap-based strategy for loop closure detection and back-end optimization under sparse point clouds, and (3) a modified PointNet model for classifying the point clouds of underground parking lots into four categories: ceiling, floor, wall, others. Experimental results show that our SPC-SLAM algorithm achieves centimeter-level accuracy (0.09% trajectory error rate) after closed loop processing in a Global Navigation Satellite System (GNSS)-denied underground parking lot, and precision of 84.8% in semantic segmentation.**

***Index Terms*— Backpack LiDAR, point cloud, SLAM, underground parking lots, semantic modeling.**

I. Introduction

PRECISE geolocation of individuals and vehicles has revolutionized commercial logistics and personal travel. Such applications have in common that they are predominantly carried out in open line-of-sight environments that can access Global Navigation Satellite System (GNSS) signals [1]. However, GNSS may be blocked in urban canyons and indoor or underground facilities. With increasing urbanization and rapid development of autonomous driving technology, it is essential to be able to determine locations accurately in these GNSS-denied environments. How to rapidly and accurately locate individuals and vehicles in a huge GNSS-denied underground parking lot is one of the most challenging research topics. In particular, accurate navigation of the rapidly developing autonomous cars in an underground parking lot relies, not only on multiple sensors mounted on the cars, but also on a High-Definition map (HD map) that includes semantic information of the underground parking lot [2], [3].

However, in challenging environments such as indoor, underground, or urban scenarios, the performance of GNSS receivers is jeopardized. One of the current solutions to this problem is to use the LiDAR-based Simultaneous Localization and Mapping (SLAM) algorithm [4], which provides for accurate mapping in some complex environments where GNSS signals become unavailable.

SLAM. SLAM technology was first developed for robotics. Feature-based SLAM uses corners, shapes or contrasts to calculate the position of the sensor. However, this approach falls short in areas, such as tunnels or caves that lack characteristics and objects to locate. By minimizing the difference from frame to frame, measurement-based SLAM uses all points acquired by LiDAR and determines the best position of the sensor. The advantage of this approach is the ability to scan any type of environment. In addition, it is easier to merge this approach with other approaches, such as integrating the Inertial Measurement Unit (IMU) and ground control points to achieve higher accuracy in areas unsuited to SLAM. The main advantage of SLAM for 3D mapping applications is that scanning can be undertaken while in motion with no GNSS signal access. This provides rapid and accurate 3D data acquisition in complex environments, and is particularly effective for indoor mapping where it is impossible to use current GNSS-based mobile mapping solutions. SLAM may very well be the Holy Grail that has eluded indoor mapping applications.

Laser scanning systems. Laser scanning systems have experienced a major evolution in the past decades. After the initial breakthrough of Airborne Laser Scanning (ALS) systems [5], [6], other types of laser scanning systems, most notably Terrestrial Laser Scanning (TLS) [7] and Mobile Laser

Manuscript received February 12, 2019; revised June 3, 2019, August 10, 2019, and October 6, 2019; accepted November 21, 2019. This work was supported in part by the Natural Science Foundation of China under Grant 41871380, Grant 41471379, and Grant 61771413. The work of Z. Gong was supported by the China Scholarships Council under Grant 201806310155. The Associate Editor for this article was J. M. Alvarez. *(Corresponding author: Jonathan Li.)*

Z. Gong and J. Li are with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University, Xiamen 361005, China, with the Fujian Haixi Collaborative Innovation Center for Big Data Applications in Governments, Xiamen University, Xiamen 361005, China, with the School of Informatics, Xiamen University, Xiamen 361005, China, and also with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: gongzheng@stu.xmu.edu.cn; junli@xmu.edu.cn).

Z. Luo, C. Wen, and C. Wang are with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University, Xiamen 361005, China, with the Fujian Haixi Collaborative Innovation Center for Big Data Applications in Governments, Xiamen University, Xiamen 361005, China, and also with the School of Informatics, Xiamen University, Xiamen 361005, China (e-mail: zpluo@stu.xmu.edu.cn; clwen@xmu.edu.cn; cwang@xmu.edu.cn).

J. Zelek is with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: jzelek@uwaterloo.ca).

Digital Object Identifier 10.1109/TITS.2019.2955734

Scanning (MLS) [7] systems emerged. While these three main types of laser scanning systems serve a large number of applications, none can be optimized for fast and flexible scanning in challenging locations. Filling this void are backpack LiDAR systems, which are now evolving towards providing for compact, agile and flexible solutions to map complex environments, in particular, those difficult urban, indoor, and underground environments where GNSS signals are inaccessible [8], [9]. Typically, a backpack LiDAR system comprises a ranging and imaging unit consisting of laser scanner(s) and/or digital cameras, a positioning and navigation unit for georeferencing and a time referencing unit. Accordingly, backpack LiDAR is defined as a technology for capturing point cloud and image data using ranging and imaging sensors that are attached to a moving person [10].

Semantic Segmentation. As a fundamental and challenging task in 3D point cloud understanding, semantic segmentation aims to label every point in a 3D scene. However, because of the properties of point clouds, in practice, semantic segmentation faces many challenges. For example, the points are from a space with a distance metric, meaning that there is rich local structure information among the points, which makes the relationship between points implicit and difficult to be mined. Besides, the sparseness of point clouds renders most of the methods inefficient.

In this paper, point clouds collected by a backpack lasers scanning (PLS) system with two Velodyne VLP-16 laser scanners [11] (see Fig. 1) were used to map a GNSS-denied underground parking lot. A front-to-back framework was presented for indoor localization and mapping. The modified PointNet model [12] was applied to semantic segmentation of the BLS point clouds to obtain four types of objects including ground, wall, ceiling, and others. These four categories of semantic information can be used to support navigation and automated parking of a car in such a GNSS-denied underground parking lot.

The main contributions of our work are as follows:

First, for the geometric characteristics of the indoor structure or underground parking lots, a probabilistic registration algorithm was developed to register the automatically calibrated BLS point clouds from the two VLP-16 laser scanners to improve localization and modeling accuracy. Furthermore, a back-end loop closure detection and optimization algorithm was developed based on the geometric-rigidity-constant assumption between sub-maps to improve the robustness and accuracy of the loop closure detection for raw sparse BLS point clouds.

Second, to guide the training of PointNet, we modified PointNet [12] with a prior module that preserves the previous global structural information of the 3D shape. Through this operation, we determined the coarse shape of an object by its global shape and obtained the object, itself, using certain detailed characteristics. It must be pointed out that, because the prior module is an individual module, the prior module can be inserted into other frameworks.

The rest of the paper is structured as follows: Section II reviews the related work. Section III describes the architecture of our backpack LiDAR system and its calibration, and

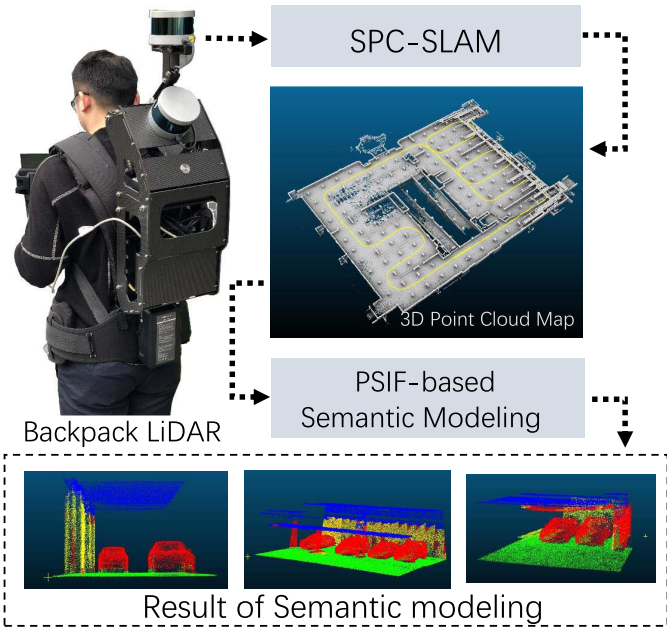


Fig. 1. Pipeline of semantic modeling of underground parking lots: SPC-SLAM algorithm for generating HD maps, PSIF model for semantic segmentation and modeling.

then details the 3D SLAM method and semantic modeling approach. Section IV presents and discusses the experimental results. Section V concludes the paper.

II. Related Work

The first backpack LiDAR prototype was developed in 2010 at the Video and Image Processing Lab, University of California, Berkeley. Their backpack was equipped with five Laser Range Finders (LRFs), one IMU, and two fish-eye cameras [13]. The "Akhka backpack LiDAR system developed in 2013 at the Centre of Excellence in Laser Scanning Research with the National Land Survey of Finland was equipped with a FARO Photon-120 laser scanner and integrated with a GNSS-IMU navigation system and a data recording laptop computer [14]. Researchers at the University of Wurzburg, Germany, presented a backpack LiDAR system, featuring with a SICK LMS 100 laser scanner and a high-end Riegl VZ-400 3D laser scanner [15], which used a 6-DOF-SLAM algorithm to build 3D maps. Researchers at the Swedish University of Agricultural Sciences developed and applied a backpack equipped with a VLP-16, a positioning system, and a computer for system control and logging data for forest inventory [16]. This backpack relies on a commercial stereo vision solution for locations in GNSS-denied environments. Glennie et al. [17], using GPS/GNSS and IMU for accurate pose estimation, developed a LiDAR system equipped with Velodyne HDL-32. Also, for 3D mapping, Wen *et al.* [18] used a LiDAR backpack with a 2D laser scanner, which, using an extended Kalman filter (EKF)-based method, fused data from three 2D laser scanners and an inertial sensor for 6-DOF pose tracking. Filgueira *et al.* [19] Lagüela *et al.* [20] used a LiDAR backpack equipped with a 3D laser scanner,

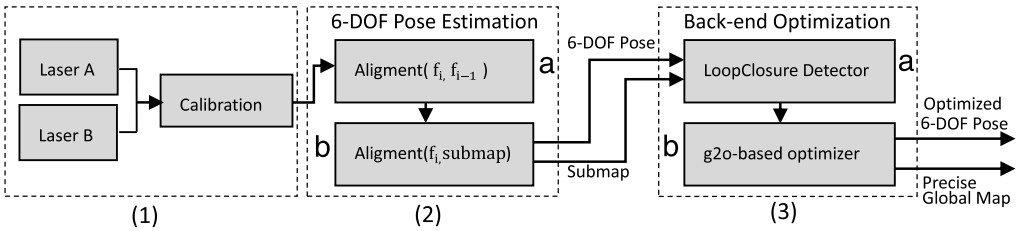


Fig. 2. Block diagram of our localization method: (1) point cloud acquisition and system calibration using an online algorithm, (2) input of calibrated data into our front-end module, (3) generation of a 6-DOF pose and detection of loop-closure and global optimization of the trajectory using a submap-based back-end optimizer, and (3) generation of a global uniform point cloud map.

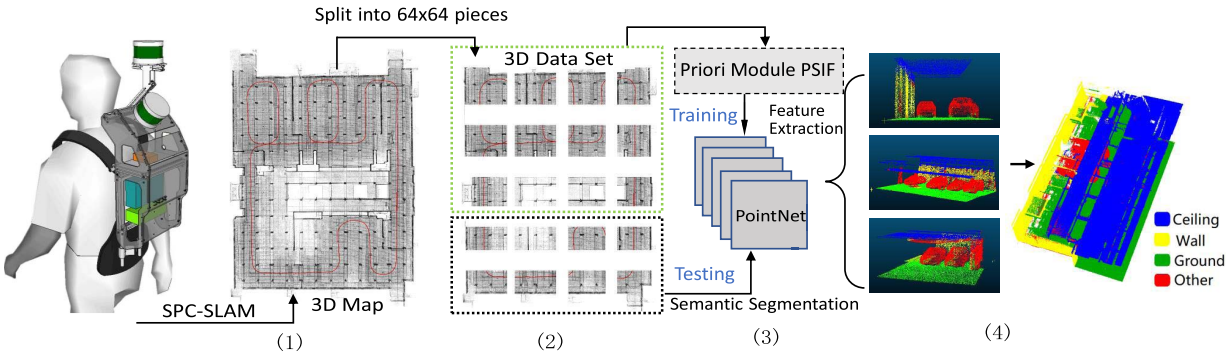


Fig. 3. Flowchart of our semantic modeling algorithm: (1) estimating walking trajectory and reconstructing a 3D map, (2) splitting the 3D map into 64x64 blocks, (3) inputting blocks into PointNet for training and semantic segmentation, and (4) merging the output into a semantic map.

which, using an ICP-based [21] method, fused data from an inertial sensor for 3D mapping.

In terms of SLAM approach, Bailey *et al.* [22] proposed an EKF-based SLAM. Zhang and Singh [23] developed a feature-based front-end odometry approach, which is a state-of-the-art LiDAR-based mapping algorithm in the KITTI benchmark [24]. In recent years, with the advent of high-performance 3D sensors [25], [26], advanced algorithms and applications have been proposed for different sensors, including RGBD-SLAM [27], LiDAR-based SLAM [28]–[30], etc. For the application of parking lots, [31] proposed an approach to autonomous automotive navigation in large-scale parking garages with potentially multiple levels. Han *et al.* [32] proposed a parameterized map-building approach, which enables the high-precision navigation and memory efficient map representation of an underground parking scene. However, these works did not perform precision analysis and semantic information extraction for smart parking.

In terms of the semantic segmentation of 3D point cloud data, Munoz *et al.* [33] used a functional gradient algorithm to learn Associative Markov Network (AMNs) models [34] for 3D point cloud semantic segmentation. Traditionally, the handcrafted descriptors are designed to obtain features. Then, those features are fed to classifiers, e.g. Support Vector Machines (SVMs), to label each point. Typical methods include Spin Image [35], SHOT [36] and RoPS [37]. Although acceptable results are achieved in traditional works, the descriptiveness in existing methods is still far from satisfactory. The main issue lies in insufficiently describing 3D data using statistical information. Each of these methods catches only a portion of the geometric characteristics. However, a 3D shape is comprised of complex topological structures and visible variational geometry. Recently, with the development of Deep Neural Network (DNN), various DNN-based methods, such as VoxelNet [38], OC-Net [39], MV-CNN [40], and PointNet [12], have been employed to deal with this problem. For some datasets, the above DNN-based methods have shown progressive improvement in descriptiveness. However, some important priori knowledge, such as significant global structural information, is rarely considered in these methods. Furthermore, due to the property of back propagation in DNN, the challenge remains to process the 3D point clouds efficiently. Hence, to improve the efficiency and accuracy of DL-based methods, we pursued the possibility of adding the prior information hidden in the 3D point clouds.

III. Method

A. Sensor Calibration

A block diagram of the hardware of our device is displayed in Fig. 2. The LiDAR backpack contains two 16-beam 3D laser scanners (Fig. 4(a)). Each laser scanner consists of 16 individual laser detector pairs over a $30°(-15°to+15°)$ field of view (Fig. 4(b)). One laser scanner is placed horizontally to acquire the point cloud, P_{LaserA}; The other laser scanner is mounted at 45° below the horizontal one to acquire the point cloud, P_{LaerB}. This layout, ensuring full coverage of the 3D space, provides a stable data source for robust pose estimation and greater map detail. Multi-sensor systems require data calibration; whereas, traditional manual calibration, which is time-consuming and has low accuracy

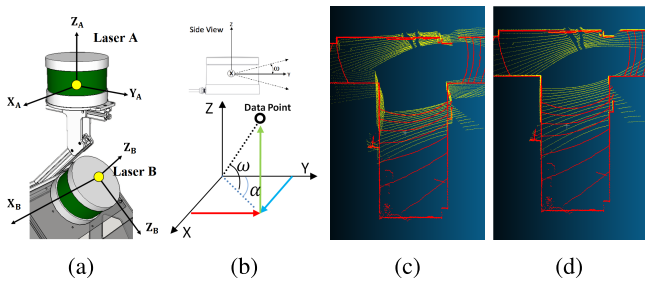


Fig. 4. (a) Coordinates of laser scanners A and B. (b) Coordinates of a data point. (c) LiDAR data before calibration. (d) LiDAR data after calibration.

and stability, basically relies on specially-designed targets. Therefore, we used target-free automatic calibration technology [41], which, to achieve automatic calibration, relies on an isomorphism constraint model of the environment. Based on the assumption that data and LaserA's trajectories T_A^n are synchronized, we transformed the point cloud of laser scanner B (P_B^n) to its location at time, n, in the submap, M, using T_A^n and T_{guess}. Here, T_{guess} is the initial value of calibration matrix estimated by a coarse manual measurement. Then, we used the nearest neighbor search algorithm (Expressed in Eq. (1) by function NN) to find the nearest neighbor point, P_A^n, on the sub map. Finally, we used the environmental consistency constraint to deduce Eq. (1) and obtain T_{cali}^n.

$$\begin{aligned} P_A^n &= NN(M, T_A^n, P_B^n, T_{guess}), \\ T_{cali} &= \arg\min_{T_{cali}} \sum_{i=0} \| P_B^n * T_{cali} - P_A^n \| \end{aligned} \tag{1}$$

Then, we used Eq. (2) to merge the scanners to acquire the global point cloud P_{Global}, where T_{cali} is a transform matrix calculated between the two laser scanners by Eq. (1). Fig. 4(c), 4(d) shows the data before/after automatic calibration.

$$P_{global} = P_{LaserA} + T_{cali} * P_{LaserB} \tag{2}$$

B. 6-DOF Pose Estimation

Once the calibrated point clouds are obtained from the dual laser scanners, they are combined into a 6-DOF pose estimation algorithm for system pose tracking and the incremental building of submaps (Fig. 2(2)). Zhang and Singh [23] used selected feature points to determine a transformation matrix between point cloud frames. When the geometric features are deficient, the stability of the algorithm is easily affected. Inspired by the Generalized-ICP (g-ICP) [42] algorithm, to achieve a more accurate and robust estimation, we introduced a probability framework (Eq. (4)) to the point cloud alignment by using the local point cloud surface variance as the weight of registration, which modifies the registration failure caused by the lack of features in the sparse point clouds. Then we used this alignment twice to calculate a 6-DOF pose, that is, frame to frame alignment step and frame to submap alignment step, based on this probability framework and oc-tree based approximate nearest neighbors search. The detailed description of the algorithm is as follows:

$$f_i^0 = \mathbf{T}_{i-1} \cdot f_i \tag{3}$$

$$\Delta\mathbf{T}_i = \arg\min_{\Delta\mathbf{T}_i} \sum_{j=1}^{n} d_j{}^{(\Delta\mathbf{T}_i)^T} (C_j^{f_i^0} + \Delta\mathbf{T}_i C_j^{f_{i-1}^0} \Delta\mathbf{T}_i^T)^{-1} d_j^{(\Delta\mathbf{T}_i)} \tag{4}$$

$$\begin{bmatrix} f_i^0 \\ \mathbf{T}_i{}' \end{bmatrix} = \Delta\mathbf{T}_i \cdot \begin{bmatrix} f_i^0 \\ \mathbf{T}_{i-1} \end{bmatrix} \tag{5}$$

Step 1, Frame-to-Frame Alignment. In this frame-to-frame alignment (Eq. (3), Eq. (4), and Eq. (5) below), we use the data of the two adjacent frames, f_i, f_{i-1}, from the two calibrated laser scanners to align the point clouds and use the resultant transform matrix, $\mathbf{T}_i{}'$, as a rough estimate of odometry. First, the current frame, f_i, is aligned with the global coordinate system, f_i^0, within the previous global transform matrix $\mathbf{T}_{i-1}$. Here d_j is the distance between two transformed frames.

Second, g-ICP, a robust algorithm for sparse point cloud data registration is used to pair align f_i^0 with the previous frame, f_{i-1}^0, and obtain the transform matrix, $\Delta\mathbf{T}_i$, to align the current frame, f_i^0, with the global coordinate system and update the temp global transform matrix, $\mathbf{T}_i{}'$. $C_j^{f_i^0}$ and $C_j^{f_{i-1}^0}$ are covariance matrices associated with the measured point clouds, f_i and f_{i-1}, according to the probabilistic model of the g-ICP. Here, to calculate the variance of every point, we use the nearest thirty points as the local neighborhood.

$$\Delta\mathbf{T}_i{}' = \arg\min_{\Delta\mathbf{T}_i{}'} \sum_{j=1}^{n} d_j{}^{(\Delta\mathbf{T}_i')^T} (C_j^{f_i^0} + \Delta\mathbf{T}_i' C_j^s \Delta\mathbf{T}'^T_i)^{-1} d_j^{(\Delta\mathbf{T}_i')} \tag{6}$$

$$\begin{bmatrix} f_i^0 \\ \mathbf{T}_i \end{bmatrix} = \Delta\mathbf{T}_i{}' \cdot \begin{bmatrix} f_i^0 \\ \mathbf{T}_i{}' \end{bmatrix} \tag{7}$$

$$\begin{aligned} f_{i-1}^0 &= f_i^0, M_i' = M_{i-1}' + f_i^0, \\ &where \quad i \in 0 - m \end{aligned} \tag{8}$$

Step 2, Frame-to-Map Alignment. To reduce cumulative error and more accurately estimate attitude, we use the re-match strategy (Step 2). As shown in Eqs. (6), (7) and (8), on the basis of Step 1, we use the FLANN Search Algorithm [43] to retrieve the nearest neighbor, s, of f_i^0 in submap, M_n'. M_n' is defined in Eq. (8), and m is the size of the submap.

Second, we re-use g-ICP to pair align f_i^0 with the nearest neighbor, s, and obtain the transform matrix, $\Delta T_i{}'$, to fine tune the current frame, f_i^0, to the global coordinate system and update the final global transform matrix, $\mathbf{T}_i$. $C_j^{f_i^0}$ and C_j^s are covariance matrices associated with the measured point clouds, f_i, and the nearest point cloud in submap, s, according to the probabilistic model of the g-ICP. The flow diagram of the front-end algorithm is described by Algorithm 1.

C. Loop Closure Detection and Global Optimization

Because of algorithmic error and data noise, the registration error of 3D mapping accumulates when the system traverses a long distance, which causes the global map to be staggered on the loop at point of closure (Fig. 7(a)). To minimize the cumulative error and generate a globally

Algorithm 1 (T_i, M_i')=$PoseEstimation(f_n)$: Caculate Global Transform T_i and M_i' from n Dual LiDAR Calibreated Frame f_n

Input: LiDAR data frame f_i
Output: $T_i : [T_1, T_2, \ldots, T_n], submap : M_i$
for $i = 0; i \leq n; i \neq s$ **do**
  // STEP.1: Frame to Frame match ;
  $f_i^0 = T_{i-1} \cdot f_i$;
  $\Delta T_i = GeneralizedICP(f_i^0, f_{i-1}^0)$;
  $f_i^0 = applyTransform(f_i^0, \Delta T_i)$;
  // STEP.2: Frame to Map match ;
  if $i == 0$ **then**
    $Push(Submap, f_i^0)$;
  else
    $f_{neighbor} = ANN(Supmap, f_i^0)$
    $\Delta T_i{}' = GeneralizedICP(f_i^0, f_{neighbor})$;
    $f_i^0 = ApplyTransform(f_i^0, \Delta T_i{}')$;
    $T_i = T_i{}' \cdot Global_{i-1}$;
    $M_i' = M_{i-1}' + f_i^0$;
  $f_{i-1}^0 = f_i^0$;
  $M_{i-1}' = M_i'$;
return (T_n, M_n');

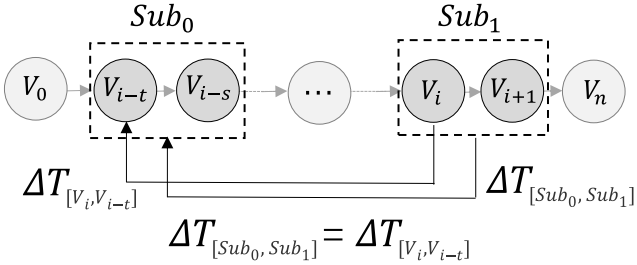


Fig. 5. Structure of pose graph.

consistent map, fusing local submaps into a single geometrically consistent 3D global map must be considered. How to create a closed loop path and submap is described in Fig. 5. Once submaps are created and the loop closure is detected, using the submap-based back-end optimization algorithm based on g2o [44], the submaps are combined into a single geometrically consistent 3D map.

As seen in Eq. (9), V represents the six vector parameters $[(\phi_i, \theta_i, \psi_i), (x_i, y_i, z_i)]$ of the graph vertex. Eq. (10) defines the general optimization equation for Posegraph [44], where $z_{i,j}$ represents the mesured value between i, j, and Ω represents the error weight matrix; $e(V_i, V_j, z_{i,j})$ is an error function that calculates the similarity between (V_i, V_j) and $Z_{i,j}$. According to Kummerle et al. [44], the error function is defined as Eq. (11), where the function $t2v$ represents the transformation of the pose matrix into a vector.

$$V_i = [R_i, T_i] = [(\phi_i, \theta_i, \psi_i)^T, (x_i, y_i, z_i)^T]^T \tag{9}$$

Algorithm 2 $(Flag, H_{icp})$=$LoopClosureDetecter(T_n, \delta, e)$: Detect the Loop Closure from T_n Using the Control Parameters δ, e, and calculate submap's transform H_{icp}.

Input: T_n, δ, e (Here $\delta = 0.5$, $e = 0.02$)
Output: H_{icp}, $Flag$
Flag=0;
for $i = 0; i \leq n; i \neq s$ **do**
  while $FindClosePostation(T_i, \delta) == true$ **do**
    $referTraj.Push(T_i)$
  if $referTraj.size > 0$ ***and*** $alignTraj.size > 0$ **then**
    $M_{sub}^A \leftarrow BuildMap(referTraj)$;
    $M_{sub}^B \leftarrow BuildMap(alignTraj)$;
    for $i \leftarrow 0$ *to* $M_{sub}^A.Size$ **do**
      $(M_{sub}^N \leftarrow KNN\ (M_{sub}^A, M_{sub}^B)$;
    $(H_{icp}, err_{icp}) \leftarrow Align\ (M_{sub}^N, M_{sub}^A)$ **if** $err_{icp} < e$ **then**
      Flag=1;
return $(Flag, H_{icp})$;

$$F(V) = \sum_{\langle i,j \rangle \in C} e(V_i, V_j, z_{i,j})^T \Omega_{i,j} e(V_i, V_j, z_{i,j}),$$

$$V^* = \arg\min F(V) \tag{10}$$

$$e(V_i, V_j, z_{i,j}) = t2v(Z_{i,j}^{-1}, (V_i^{-1}, V_j)) \tag{11}$$

In multi-beam laser scanner SLAM applications, because of the sparsity of sensor data, it is difficult to ensure the correctness and robustness of loop closure detection in single frame data. To solve this problem, we propose a closed loop detection based on local sub map and pose graph construction. Submaps and their corresponding trajectories are rigid constraints. Based on this assumption, we match submaps on a point of loop closure and obtain a transform matrix as a new measurement, which provides a new constraint for global optimization (See Fig. 5). The main idea to avoid closed loop calculation errors caused by single sparse point clouds encompasses the rigid transformation properties of the trajectory: $\Delta T_{Sub_0, Sub_1} = \Delta T_{V_i, V_{i-t}}$. The specific steps are as follows:

First, the sub map, M_n', and the corresponding trajectory, T_n, are obtained from the 6-DOF pose estimation. Then, the distance between poses is used as the starting condition for loop closure detection. In other words, once the distance between a current and a past pose is detected as being less than the set threshold, δ, the closest point sets, T_c, are extracted and closed loop detection (Algorithm 2) is begun.

Second, the score of the submap registration is used as the basis for successive loop closure detection. The derivation of this process is giving by Eq. (12). Once the closed loop is detected, these two sub-maps, M_{sub}^A, M_{sub}^B, are aligned, and the new transform, H_{icp}, is added and combined with the current posture, V_j, as the new measurement, V_j'. The new edge, $z_{i,j}$, is caculated with loop closure point, V_i, and added it to the pose graph. Finally, using Eq. (12), the global consistency trajectory is calculated and used to reconstruct a high accuracy

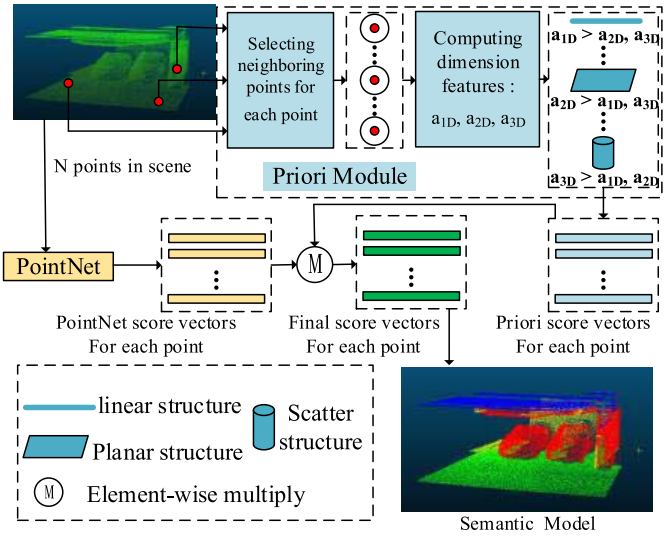


Fig. 6. Flowchart of semantic modeling algorithm (1) prior information extraction module, (2) global information mining module, and (3) feature fusion module.

global map.

$$\begin{aligned} z_{i,i+1} &= V_i^{-1} \cdot V_{i+1}, \\ z_{i,j} &= \Delta T_{V_i,V_j} = V_i^{-1} \cdot V_j{}', \ while \\ V_j{}' &= H_{icp} \cdot V_j \Rightarrow z_{i,j} = V_i^{-1} \cdot H_{icp} \cdot V_j \end{aligned} \tag{12}$$

D. Semantic Modeling for Underground Parking Lots

Considering its excellent results with indoor scene segmentation, we adopt the PointNet scheme as our basic module (See Fig. 6). PointNet takes the indoor 3D point clouds as input, and labels every point. However, likes other DNN-based method, PointNet seldom takes the prior information into account. Therefore, considering there is much prior geometric structure information in an underground garage, to guide the training of PointNet, we proposed an extra prior module, Priori Prior Structure Information Filter (PSIF), which preserves the prior global shape information of 3D shapes.

By utilizing the PSIF module and the basic PointNet module, we propose our model, Prior-based PointNet Model (PPM). As shown in Fig. 6, PPM consists of three key parts: prior information extraction module, global information mining module, and feature fusion module. The prior module provides the rich prior 3D shape information. So it can be considered as a guide to speed up the training. Furthermore, the prior module can provide more supplementary information to improve the performance of the model. Besides, the structure of the global feature extraction module is based on the PointNet, which has been proved powerful in describing the global 3D shape. Additionally, feature fusion module merges the global feature and prior feature to obtain the final fusion feature by operating an element-wise multiply.

The preliminary inference of geometric structure plays an important role in the processing of 3D point cloud data. For example, rough geometry structure can depict the outline of a 3D shape, thus preliminarily confirming the classification of the 3D object. Because a 3D point cloud is comprised of a set of points, its geometric structure consists of the relationship between points. Therefore, the shape of a 3D point cloud depends on the connection between points. The relationship between points and some of the results obtained are reported in [45], [46]. The most typical result is the dimension feature, that is, the relationship between the properties of the point set and the geometry of the 3D point cloud.

Dimensionality features. For a given point set $P = \{p_1, p_2, \ldots, p_n\}$, where n denotes the number of points, a Principal Components Analysis (PCA) is performed on the point set P to obtain three eigenvalues $(\lambda_1, \lambda_2, \lambda_3)$, such as $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$, and three eigenvectors $(\overrightarrow{v_1}, \overrightarrow{v_2}, \overrightarrow{v_3})$. Then the standard deviation σ_i along an eigenvector i is denoted as follows:

$$\forall i \in [1, 3], \sigma_i = \sqrt{\lambda_i} \tag{13}$$

The 3D shape of the point clouds is then represented by an oriented ellipsoid. Three geometrical features are introduced in order to describe the linear (a_{1D}), planar (a_{2D}) or scatter (a_{3D}) behaviors [45] as follows:

$$a_{1D} = \frac{\sigma_1 - \sigma_2}{\sigma_1}, a_{2D} = \frac{\sigma_2 - \sigma_3}{\sigma_1}, a_{3D} = \frac{\sigma_3}{\sigma_1} \tag{14}$$

According to the relationship between the distribution of points and the three eigenvalues [45], if $\sigma_1 \geq \sigma_2$ and σ_3, a_{1D} is therefore larger than a_{2D} and a_{3D}, and a linear structure is represented by point set P. If $\sigma_1 \simeq \sigma_2 \geq \sigma_3$, a_{2D} is the largest one, and a planar structure is represented by point set P. Finally, $\sigma_1 \simeq \sigma_2 \simeq \sigma_3$ means that a_{3D} is larger than a_{1D} and a_{2D}, and a scatter structure is represented by the point set P.

Apart from the above dimensionality features, we also introduce the omnivariance feature O, which is also derived from the standard deviation as follows:

$$O = \prod_{i \in [1,3]} \sigma_i \tag{15}$$

The omni-variance feature, O, proportional to the ellipsoid volume, allows for characterizing the shape of the neighborhood, and, in particular, enhancing whether one or two eigenvalues are prominent.

Priori Module: Priori Structure Information Filter (PSIF). For a given point set $P = \{p_1, p_2, \ldots, p_n\}$, combined with the above three dimensionality features and the omnivariance feature, we define the priori module as a filter, which is actually a priori structure probability score vector $S^p = \left\{S_1^p, S_2^p, \ldots, S_m^p\right\}$, where m is is the number of categories, and S_i^p is the priori structure probability score of categories i.

The assumption is that, for 3D point clouds belonging to the same class, the distribution of probability scores follows a normal distribution. This assumption is in line with the actual situation. Then, the vector, S^p is computed by a transformation F, as follows:

$$F(a_{1D}, a_{2D}, a_{3D}, O) = S^p = \left\{S_1^p, S_2^p, \ldots, S_m^p\right\}, \tag{16}$$

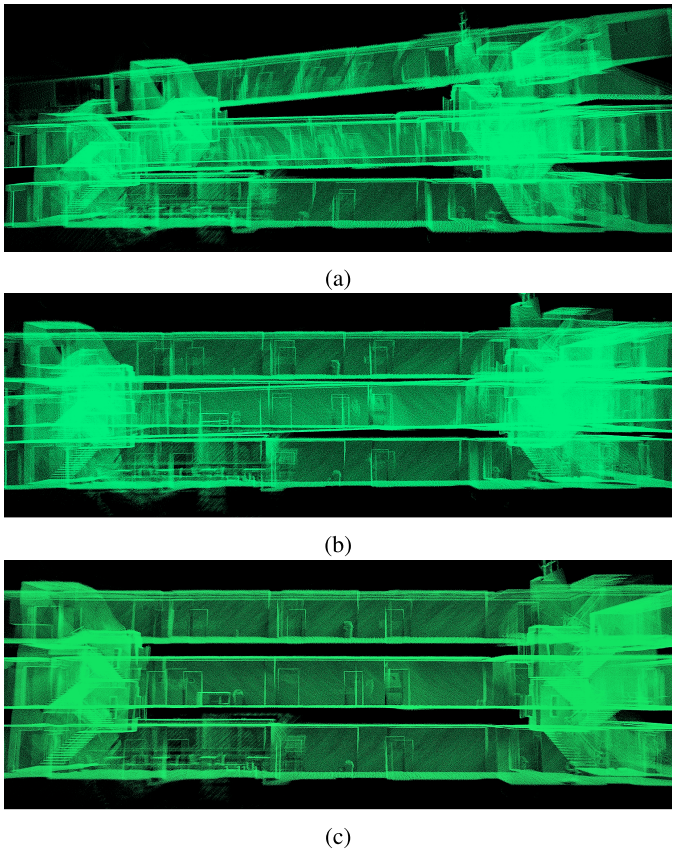


Fig. 7. Comparison of mapping results obtained using (a) LOAM algorithm, (b) our SPC-SLAM algorithm before optimization, and (c) our SPC-SLAM algorithm after optimization.

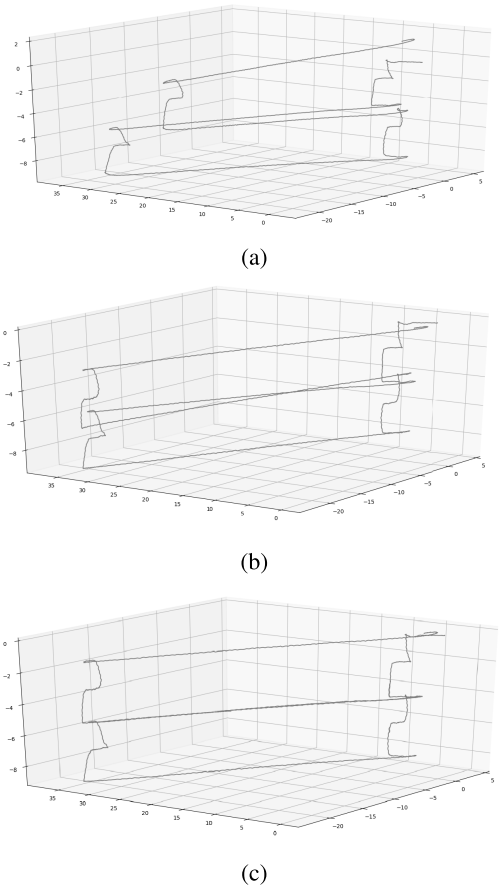


Fig. 8. Comparison of trajectory results obtained using (a) LOAM algorithm, (b) our SPC-SLAM algorithm before optimization, and (c) our SPC-SLAM algorithm after optimization.

where

$$\begin{aligned} S_i^p &= 1 + f_i^D(x_i) + f_i^O(O) \\ &= 1 + \frac{exp(-\frac{(x_i-\mu_i^D)^2}{2(\delta_i^D)^2})}{\delta_i^D\sqrt{2\pi}} + \frac{exp(-\frac{(O-\mu_i^O)^2}{2(\delta_i^O)^2})}{\delta_i^O\sqrt{2\pi}} \end{aligned} \tag{17}$$

where $x_i \in (a_{1D}, a_{2D}, a_{3D})$, and μ_i^D, μ_i^O, δ^D, δ^O are parameters that to be estimated. It must be pointed out that x_i is determined by the priori knowledge of the i^{th} category. For example, because the structure of a wall is represented by a planner structure, $x = a_{2D}$.

After obtaining the priori structure probability score vector S_p, we explicitly insert it into the results, $S_{PointNet}$, generated by PointNet. In this paper, we apply element-wise multiplication to S_p and $S_{PointNet}$, to obtain the following final score vector, S_{final}:

$$S_{final} = S_{PointNet} \cdot S_p \tag{18}$$

IV. Experimental Results

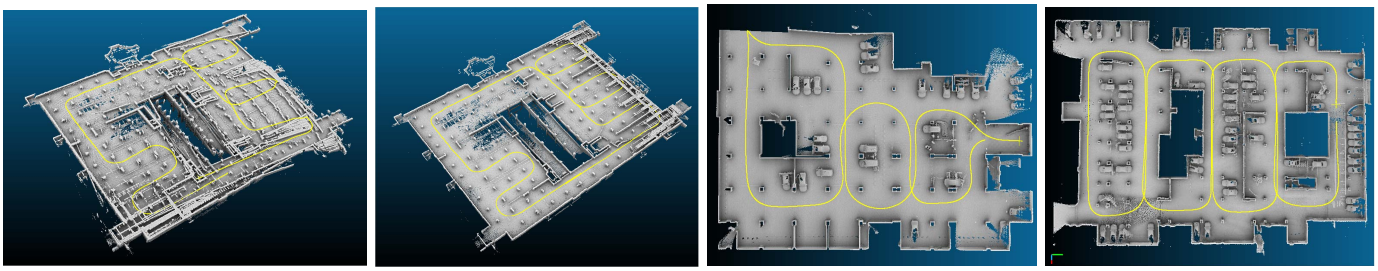
To quantitatively evaluate the accuracy and correctness of the proposed method on indoor mapping, two measurements, including (1) point cloud quality evaluation, (2) trajectory error analysis, were given. While for semantic segmention, we using (3) accuracy and (4) Intersection-Over-Union (IoU) for evaluation. We will deatils the definition of each evaluation method and the experimental results below.

TABLE I
Data Set Acquired by Our Backpack and its Properties

Data Set	Dimensions ($m \times m$)	Number of Scans	Travel Distance (m)
$FLOOR_4$	$52 \times 3 \times 4$	1730	33.6
$FLOOR_5$	$52 \times 3 \times 4$	3737	35.3+36.1
$FLOOR_6$	$52 \times 3 \times 4$	1895	37.2
$FLOOR_{4-6}$	$52 \times 3 \times 15$	7362	142.2
$ParkingLot\#1$	$54 \times 74 \times 6.5$	9.16×10^6	316.2
$ParkingLot\#2$	$55 \times 78 \times 7$	7.32×10^6	215.5
$ParkingLot\#3$	$82 \times 100 \times 7$	1.16×10^7	421.6

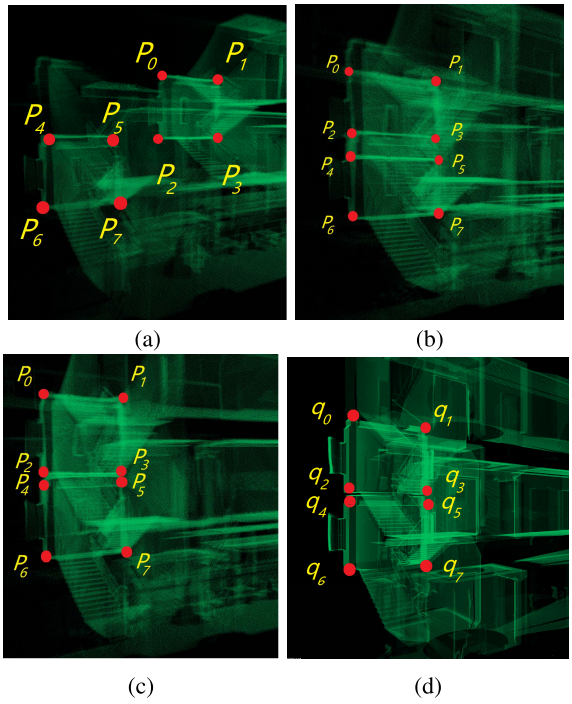
A. Experimental Setup

In the first experiment, our backpack was used to collect the point clouds of an indoor corridor inside a multi-storey building and over three underground parking lot scenes. The properties of all data set are shown in Table I. For the sake of the backend closed-loop requirement, we walked in the



(a) (b) (c) (d)

Fig. 9. Comparison of mapping results in data *ParkingLot*#1 obtained using (a) LOAM algorithm and (b) our SPC-SLAM algorithm after optimization. (c) and (d) are other two data (*ParkingLot*#2, *ParkingLot*#3) generated by our method (yellow line: walking trajectory).



(a) (b) (c) (d)

Fig. 10. Point cloud quality evaluation and comparison: mapping results obtained using (a) LOAM algorithm, (b) our SPC-SLAM algorithm before optimization, (c) our SPC-SLAM algorithm after optimization, and (d) ground truth data.

'8' pattern, which constituted three closed loop points. The indoor environment consisted of three layers of corridors, with each corridor about 40 m long and 3 m wide. The total distance in corridor (including the loop) was about 200 m. Fig. 7 shows maps built from this representative indoor environment: a long, narrow corridor and a large lobby. Figs. 9(b)–9(d) shows the point cloud map details and the trajectory (yellow curve) generated by SPC-SLAM. The data were collected by a synchronous trigger and pre-calibrated with our automated-calibration algorithm.

B. Quality Evaluation for 3D Point Cloud

In this experiments, we used sparse 3D point clouds, to compare our method with LOAM [23], the state-of-the-art 3D mapping algorithm. LOAM is a feature-based mapping algorithm that provides high-precision front-end odometry. To evaluate the trajectory accuracy and mapping quality of our method, another point cloud data set was collected in the same environments by a Riegl VZ-1000 as a ground truth map. The VZ-1000 remained stationary at a few different places in the indoor environment. To provide millimeter level accuracy for each survey point, several high-density static scans were manually stitched together and automatically refined by Riegl software.

For segmentation algorithms verification, the setting of hyperparameters for PointNet is similar to that in [12]. More specially, we used dropout with keep ratio 0.7 on the last fully connected layer. The decay rate for batch normalization starts at 0.5 and is gradually increased to 0.99. In addition, we used Adam Optimizer with initial learning rate 0.001, momentum 0.9 and batch size 32. The learning rate is divided by 2 every 20 epochs. Besides, we trained our model with TensorFlow and a NVIDIA Tesla P100 GPU.

In the first evaluation experiment, we manually selected n control points, $Q = [q_1, q_2, ...q_n]$ from the mapping results of our algorithm and manually selected corresponding survey points, $P = [p_1, p_2, ...p_n]$, from the high-density map of the VZ-1000. Fig. 10(a)–10(d) illustrates this process from the results of different approaches. After manually matching corresponding point sets, the distances between a point set and a corresponding point in the second point set were considered trajectory errors, e_t, as follows:

$$e_t(p_i, q_i) = \frac{\sum_{i=0, j=0, i<j}^{n} ||p_i - p_j||^2}{\sum_{i=0, j=0, i<j}^{n} ||q_i - q_j||^2} \tag{19}$$

As shown in Table II, two groups of mapping errors, calculated by control points for LOAM or our method, are indicated by $err_{cp_1}$and err_{cp_2}, respectively. The mapping errors for our method are smaller than those for LOAM, especially for indoor multi-storey structures ($FLOOR_4 - FLOOR_6$), where the lowest mapping error rate (0.09%) is achieved for $FLOOR_5$. This result is reasonable because, in indoor environments, the feature-based method, with symmetrical or weak features, is less exact than our g-ICP global matching method. In Table II, we also give the processing frame rate (frame per sec., FPS) of each algorithm. It can be seen that the time efficiency of our algorithm in the front-end result is close to that of LOAM. In the data with closed-loop points, we did not use multi-thread for closed-loop detection and optimized, so the time efficiency is reduced.

TABLE II
MAPPING ERROR AND TIME EFFICIENCY ANALYSIS OF LOAM AND OUR METHOD

Data\ Method	LOAM				SPC-SLAM(Front-end)				SPC-SLAM(optimized)			
	err_{cp_1}	err_{cp_2}	err_rate	FPS	err_{cp_1}	err_{cp_2}	err_rate	FPS	err_{cp_1}	err_{cp_2}	err_rate	FPS
$FLOOR_6$	8.9918	8.9946	**27.03 %**	9.12	0.0653	0.1102	**0.34%**	8.37	0.0294	0.065	**0.24%**	8.37
$FLOOR_5$	2.0516	2.3076	**7.65 %**	8.78	0.2944	0.1482	**1.39%**	8.91	0.0053	0.0297	**0.09%**	8.91
$FLOOR_4$	1.0564	1.0804	**3.11 %**	10.37	0.1056	0.0658	**0.74%**	10.31	0.0136	0.0156	**0.12%**	10.31
$FLOOR_{4-6}$	4.0204	3.8976	**12.6 %**	9.53	0.2461	0.1758	**0.82%**	9.96	0.0494	0.105	**0.15 %**	5.33
$ParkingLot\#1$	0.8246	1.3273	**2.71%**	9.25	0.3284	0.1648	**0.58%**	8.22	0.1123	0.0856	**0.32%**	4.21
$ParkingLot\#2$	0.7327	0.8256	**1.44%**	8.66	0.1223	0.2241	**0.61%**	11.37	0.0982	0.1187	**0.26%**	2.87
$ParkingLot\#3$	1.6643	2.8256	**5.47%**	10.23	0.1764	0.1648	**0.49%**	10.09	0.0572	0.0411	**0.19%**	1.33

C. Trajectory Error Analysis

It is difficult to obtain accurate trajectory ground truth in a narrow environment (e.g, a multi-storey building). Therefore, by using the distance between point cloud frames to analyze trajectory error, we introduce the concept: Interframe Error Curve (IEC).

The error analysis, using the rigid transform property of a point cloud, reflects the relative trajectory error by calculating the point cloud error and calculating the variance according to the data fluctuation, thereby assessing the robustness of the final algorithm and completing our experimental veri cation. The main idea of IEC is as follows: First, we must acquire the trajectory, $T_i (i \in (0 \sim n-1))$, for analysis. Each point cloud frame, f_i, is registered to the corresponding position by T_i, as follows:

$$f_i{}' = T_i * f_i, \quad i \in (0 \sim n-1) \tag{20}$$

Then, we caculate interframe error $e_{f'}$, as follows:

$$e_{f'}(f_i{}', f_{i+1}{}') = \sum_{j=0}^{m} ||p_j - s_j||^2$$
$$where \quad p_j \in f_i{}', q_j \in f_{i+1}{}', s_j = NN(q_j, p_j) \tag{21}$$

In Eq.(21), p_j, q_j are points in point cloud frames $f_i{}'$, $f_{i+1}{}'$. To reduce the matching error caused by outliers, we use the nearest neighbor algorithm, NN function, to extract adjacent points, s_j, of corresponding points, p_j, as matching points. Fig. 11(a)–11(d)shows the interframe error curves for LOAM and SPC-SLAM across parts of scan pairs. As shown in the four scenes, the errors for SPC-SLAM are significantly lower than those for LOAM, which indicates a more robust trajectory estimation for SPC-SLAM and proves the accuracy of our algorithm.

For loop closure detection, we used data $FLOOR_{4-6}$ and $ParkingLot_{\#1-\#3}$ to compare the precision-recall (P-R) curve of our algorithm and M2DP [47] (Fig. 12(a)–12(d)). As can be seen from the figure, compared with the underground parking lot data, the closed-loop detection of indoor data is unstable when the recall decreases. However, due to the use of sub-map, our algorithm performs better than the algorithm that uses the sparse original point cloud M2DP feature for closed-loop detection. If we use M2DP in combination with our algorithm (red curve), we can further improve the performance.

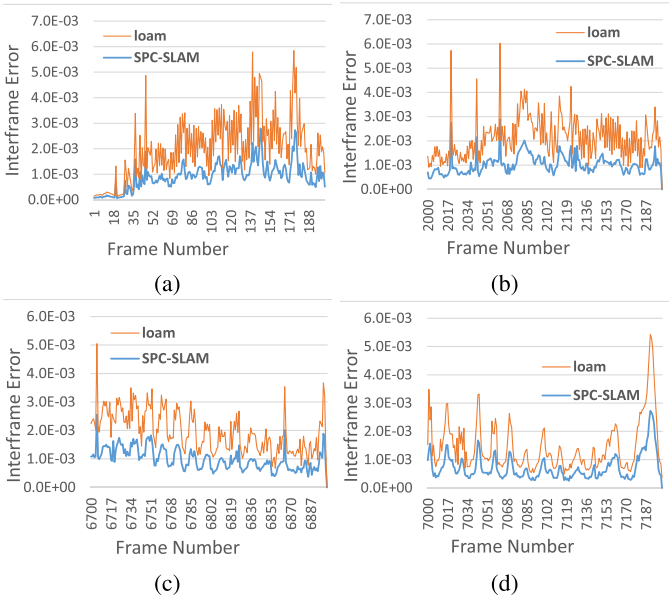


Fig. 11. Trajectory error analysis in four scenes.

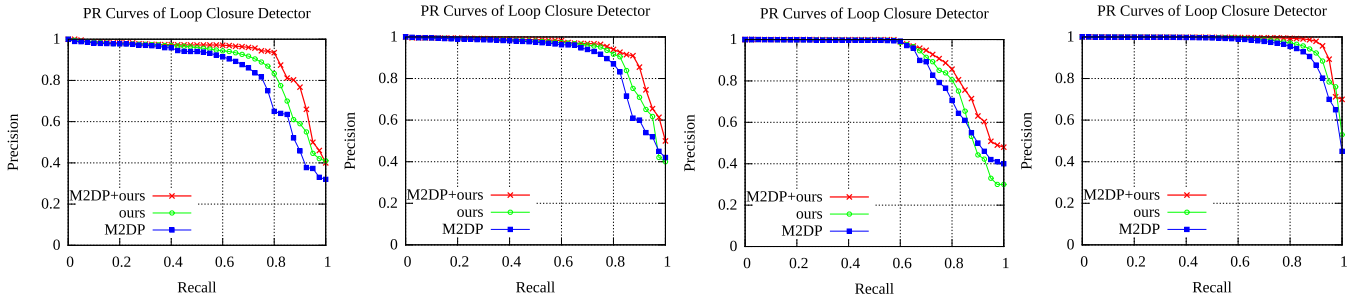
TABLE III
ESTIMATED RESULTS OF PARAMETERS IN PRIORI MODULE

P	Category			
	ceiling	floor	wall	clutter
x	a_{2D}	a_{2D}	a_{2D}	a_{3D}
μ^D	0.15	0.89	0.74	0.42
δ^D	0.07	0.07	0.05	0.06
μ^O	0.59	0.93	0.27	0.15
δ^O	0.2	0.02	0.13	0.02

In the next experiment, we took PointNet [12] as our basic framework for segmentation and performed the experiment using our underground parking lot dataset. The dataset contains 3D scans from VLP-16 in three scenes including an area of $52,0000\ m^2$.

D. Semantic Modeling Evaluation

In this subsection, we evaluate the performance of the proposed priori module PSIF on scene segmentation in an



(a) $FLOOR_{4-6}$ (b) $ParkingLot\#1$ (c) $ParkingLot\#2$ (d) $ParkingLot\#3$

Fig. 12. Recall-precision curves on $FLOOR_{4-6}$ and $ParkingLot\#1-\#3$ datasets.

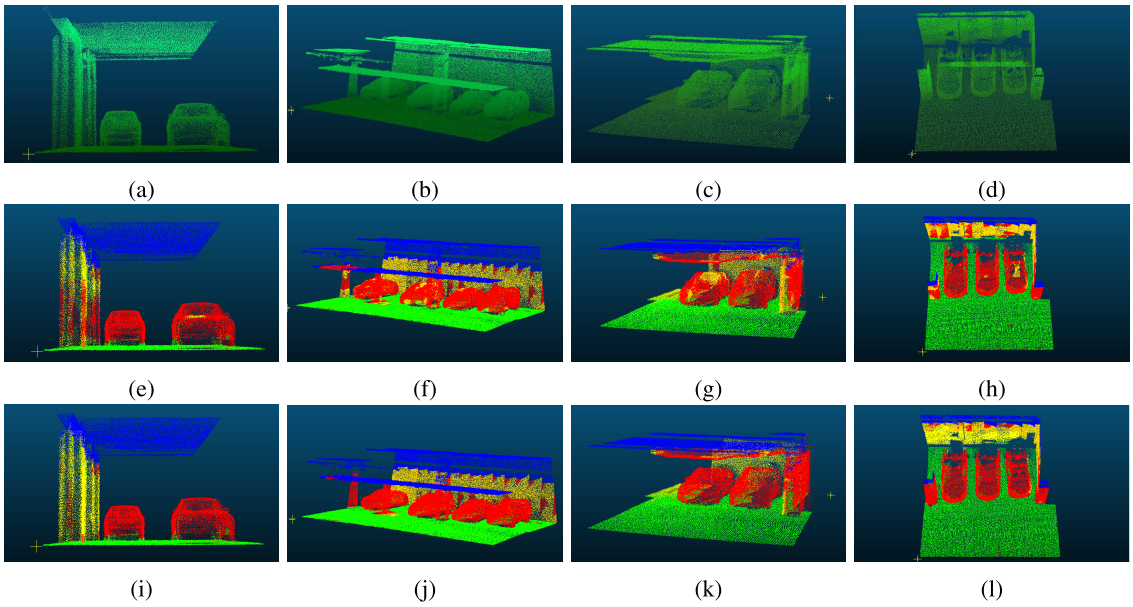


Fig. 13. Comparison of semantic segmentation results: (a)-(d) original point clouds, (e)-(h) segmentation results obtained using PointNet, and (i)-(l) segmentation results obtained using our method (black: ceiling, yellow: wall, green: floor, red: others).

underground parking lot. The dataset, containing 3D scans from VLP-16 in two areas that include 198 blocks, was acquired by the SLAM method introduced in the section above. Each point in the scan is annotated with one of the semantic labels from four categories: ceiling, floor, wall and clutter.

As in PointNet, we also predict per point class in the point cloud scene. In our experiments, each point is represented by a 3-dim vector XYZ. We used accuracy and IoU as the evaluation criteria and compared our method with the basic module, PointNet. The accuracy and IoU calculated on points and defined as Eqs. (22), (23). Where TP, FN represent the numbers of true positives, false negatives, respectively. Table III lists the parameters used in our modified PointNet (i.e., PointNet with the priori module, or PPM).

$$Accuracy = \frac{TP}{TP + FN} \tag{22}$$

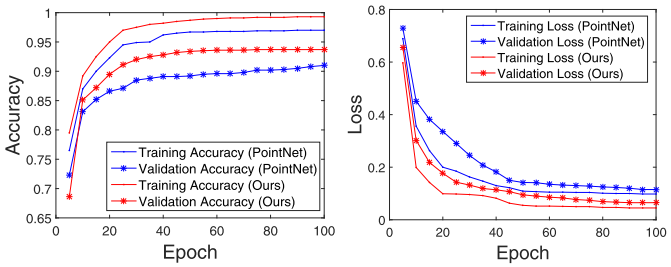
$$IoU = \frac{DetectionResult \bigcap GroundTruth}{DetectionResult \bigcup GroundTruth} \tag{23}$$

TABLE IV

COMPARED RESULTS OF SEMANTIC SEGMENTATION IN 6 SCENES (ACCURACY (%)) AND 4 CATEGORIES (IOU(%))

Method	S1	S2	S3	S4	S5	S6	mean
PointNet	83.1	78.3	81.6	81.0	75.3	87.3	81.4
PointNet++	84.0	81.6	82.3	84.5	79.3	88.1	83.3
DGCNN	86.5	82.8	83.5	84.2	79.6	88.7	84.2
PPM(Ours)	87.6	83.2	83.8	84.9	80.6	88.5	84.8
	ceiling	floor	wall	clutter			mean
PointNet	77.0	94.0	34.9	42.0	-	-	62.0
PointNet++	78.2	94.3	35.7	44.3	-	-	63.1
DGCNN	79.0	94.8	40.2	47.5	-	-	65.3
PPM(Ours)	79.2	95.5	43.4	48.7	-	-	66.7

Table IV shows the results of semantic segmentation in six scenes, in which our modified PointNet (PPM) significantly outperforms the PointNet. In particular, the accuracy of our method is about 5% higher than that of PointNet for Scene 2 and Scene 5. In addition, our method also outperforms the



(a) (b)

Fig. 14. Comparison of (a) training accuracy curves and (b) loss values generated by PPM and PointNet, respectively.

PointNet++ [48] and DGCNN [49] in Scene 1, 2, and 5. On average, the accuracy of our method is 3% higher than that of other three methods. These results show the significant effectiveness of the proposed priori module, PSIF. Prior modules provide important structural information, making segmentation effective. Shown in Table IV are the results of IoU for every category. Our PPM method achieves better performance than the basic model in every category. Especially, the IoU of the third category (wall) generated by PPM is about 9% higher than that generated by the basic model. These results also show the effectiveness of PSIF.

Our method also achieves better performance than PointNet++ and DGCNN by a large margin in the third category. These Curves of training accuracy, validation accuracy and loss values demonstrate that our modified PointNet improves the training efficiency of the PointNet. As shown in Fig. 14, our method achieves better training accuracy than PointNet. Besides, the validation curve of our method follows the training curve closely, which means that there is a small over-fitting. In addition, the curves of training and validation generated by our method also perform better than that of PointNet. Fig. 13 shows the qualitative segmentation results.

V. Conclusion

We introduced a self-developed backpack LiDAR system, and described both SPC-SLAM and priori-based semantic modeling algorithms in detail. SPC-SLAM uses online calibration and registration of dual laser scanners for sparse data in the front-end. According to the characteristics of sparse point clouds, we proposed a submap-based method for robust back-end optimization. Based on this mapping result, to guide the training of PointNet for high quality semantic modeling, we designed a prior module that preserves the prior global shape information of the 3D shape. The SLAM algorithm effectively solves the high-precision mapping problem of indoor, or underground parking lots based on the sparse point cloud data of LiDAR in GNSS-denied environment. Post-processing adopts a priori semantic segmentation algorithm, which optimizes the problem of missing data in the semantic segment and improves the integrity of the semantic map.

Experimental results show the following: (1) Mapping that is precise to within a centimeter by using our LiDAR backpack and SPC-SLAM, (2) PSIF provides basic, but important, priori structural information. Our segmentation experiments show that PSIF obtains not only better results, but also improves the training efficiency of PointNet. Semantic modeling, having precision greater than 84.8%, provides an effective way to construct semantic models in GNSS-Denied underground parking lot environments.

References

[1] C. Ellum and N. El-Sheimy, "Land-based mobile mapping systems," *Photogramm. Eng. Remote Sens.*, vol. 68, no. 1, pp. 13–17, 2002.
[2] H. Wang and W. He, "A reservation-based smart parking system," in *Proc. IEEE Conf. Comput. Commun. Workshops*, Apr. 2011, pp. 690–695.
[3] Y. Hu, J. Gong, Y. Jiang, L. Liu, G. Xiong, and H. Chen, "Hybrid map-based navigation method for unmanned ground vehicle in urban Scenario," *Remote Sens.*, vol. 5, no. 8, pp. 3662–3680, Jul. 2013.
[4] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.
[5] E. P. Baltsavias, "Airborne laser scanning: Basic relations and formulas," *ISPRS J. Photogramm. Remote Sens.*, vol. 54, no. 2, pp. 199–214, Jul. 1999.
[6] I. Colomina and P. Molina, "Unmanned aerial systems for photogrammetry and remote sensing: A review," *ISPRS J. Photogram. Remote Sens.*, vol. 92, pp. 79–97, Jun. 2014.
[7] C. Toth and G. Jóźków, "Remote sensing platforms and sensors: A survey," *ISPRS J. Photogramm. Remote Sens.*, vol. 115, pp. 22–36, May 2016.
[8] A. Jaakkola *et al.*, "A low-cost multi-sensoral mobile mapping system and its feasibility for tree measurements," *ISPRS J. Photogram. Remote Sens.*, vol. 65, no. 6, pp. 514–522, 2010.
[9] L. Díaz-Vilariño, E. Verbree, S. Zlatanova, and A. Diakité, "Indoor modelling from SLAM-based laser scanner: Door detection to envelope reconstruction," *ISPRS Arch.*, vol. 42, pp. 345–352, Sep. 2017.
[10] S. Lagüela, I. Dorado, M. Gesto, P. Arias, D. González-Aguilera, and H. Lorenzo, "Behavior analysis of novel wearable indoor mapping system based on 3D-SLAM.," *Sensors*, vol. 18, no. 3, p. 766, Mar. 2018.
[11] C. Glennie, A. Kusari, and A. Facchin, "Calibration and stability analysis of the VLP-16 laser scanner," *ISPRS Arch.*, vol. 40, p. 55, Mar. 2016.
[12] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, vol. 1, no. 2, p. 4.
[13] N. Corso and A. Zakhor, "Indoor localization algorithms for an ambulatory human operated 3D mobile mapping system," *Remote Sens.*, vol. 5, no. 12, pp. 6611–6646, Dec. 2013.
[14] Y. Wang *et al.*, "3D modeling of coarse fluvial sediments based on mobile laser scanning data," *Remote Sens.*, vol. 5, no. 9, pp. 4571–4592, Sep. 2013.
[15] A. Nüchter, D. Borrmann, P. Koch, M. Kühn, and S. May, "A man-portable, IMU-free mobile mapping system," *ISPRS Ann.*, vols. 2–3, pp. 17–23, Aug. 2015.
[16] J. Holmgren *et al.*, "Estimation of tree position and stem diameter using simultaneous localization and mapping with data from a backpack-mounted laser scanner," *ISPRS Arch.*, vol. 42, pp. 59–63, Oct. 2017.
[17] C. Glennie, B. Brooks, T. Ericksen, D. Hauser, K. Hudnut, J. Foster, and J. Avery, "Compact multipurpose mobile laser scanning system—Initial tests and results," *Remote Sens.*, vol. 5, no. 2, pp. 521–538, Jan. 2013.
[18] C. Wen, S. Pan, C. Wang, and J. Li, "An indoor backpack system for 2-D and 3-D mapping of building interiors," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 7, pp. 992–996, Jul. 2016.
[19] A. Filgueira, P. Arias, M. Bueno, and S. Lagüela, "Novel inspection system, backpack-based, for 3D modelling of indoor scenes," in *Proc. Int. Conf. Indoor Positioning Navigat. Alcalá de Henares*, Madrid, Spain, Oct. 2016, pp. 4–7.
[20] S. Lagüela, I. Dorado, M. Gesto, P. Arias, D. González-Aguilera, and H. Lorenzo, "Behavior analysis of novel wearable indoor mapping system based on 3D-SLAM," *Sensors*, vol. 18, no. 3, p. 766, Mar. 2018.
[21] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," *Proc. SPIE*, vol. 1611, pp. 586–606, Apr. 1992.
[22] T. Bailey, J. Nieto, J. Guivant, and M. Stevens, "Consistency of the EKF-SLAM algorithm," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2006, pp. 3562–3568.
[23] J. Zhang and S. Singh, "LOAM: Lidar odometry and mapping in real-time," *Robot., Sci. Syst.*, vol. 2, p. 9, Jul. 2014.

[24] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[25] V. V. Lehtola *et al.*, "Comparison of the selected state-of-the-art 3D indoor scanning and point cloud generation methods," *Remote Sens.*, vol. 9, no. 8, p. 796, 2017.

[26] M. Draelos, Q. Qiu, A. Bronstein, and G. Sapiro, "Intel realsense=Real low cost gaze," in *Proc. IEEE Int. Conf. Image Process*, Sep. 2015, pp. 2520–2524.

[27] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the RGB-D SLAM system," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2012, pp. 1691–1696.

[28] F. Moosmann and C. Stiller, "Velodyne SLAM," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 393–398.

[29] Y. Liu, W. Gao, and Z. Hu, "Geometrically stable tracking for depth images based 3D reconstruction on mobile devices," *ISPRS J. Photogr. Remote Sens*, vol. 143, pp. 222–232, Sep. 2018.

[30] X. Qu, B. Soheilian, and N. Paparoditis, "Landmark based localization in urban environment," *J. Photogramm. Remote Sens.*, vol. 140, pp. 90–103, Sep. 2017.

[31] R. Kummerle, D. Hahnel, D. Dolgov, S. Thrun, and W. Burgard, "Autonomous driving in a multi-level parking structure," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2009, pp. 3395–3400.

[32] J. Han, J. Kim, and D. H. Shim, "Precise localization and mapping in indoor parking structures via parameterized SLAM," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[33] D. Munoz, J. A. Bagnell, N. Vandapel, and M. Hebert, "Contextual classification with functional max-margin Markov networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 975–982.

[34] B. Taskar, C. Guestrin, and D. Koller, "Max-margin Markov networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 25–32.

[35] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.

[36] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 356–369.

[37] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, "Rotational projection statistics for 3D local surface description and object recognition," *Int. J. Comput. Vis.*, vol. 105, no. 1, pp. 63–86, 2013.

[38] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep./Oct. 2015, pp. 922–928.

[39] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong, "O-CNN: Octree-based convolutional neural networks for 3D shape analysis," *ACM Trans. Grap.*, vol. 36, no. 4, Jul. 2017, Art. no. 72.

[40] G. Kang, K. Liu, B. Hou, and N. Zhang, "3D multi-view convolutional neural networks for lung nodule classification," *PLoS ONE*, vol. 12, no. 11, Nov. 2017, Art. no. e0188290.

[41] Z. Gong, C. Wen, C. Wang, and J. Li, "A target-free automatic self-calibration approach for multibeam laser scanners," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 1, pp. 238–240, Jan. 2018.

[42] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," *Robot., Sci. Sys.*, vol. 2, no. 4, p. 435, Jun. 2009.

[43] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2227–2240, Nov. 2014.

[44] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G^2o: A general framework for graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 3607–3613.

[45] J. Demantke, C. Mallet, N. David, and B. Vallet, "Dimensionality based scale selection in 3D lidar point clouds," *ISPRS Arch.*, vol. 38, no. 5, p. W12, Aug. 2011.

[46] A. Gressin, C. Mallet, J. Demantké, and N. David, "Towards 3D lidar point cloud registration improvement using optimal neighborhood knowledge," *ISPRS J. Photogramm. Remote Sens.*, vol. 79, pp. 240–251, May 2013.

[47] L. He, X. Wang, and H. Zhang, "M2DP: A novel 3D point cloud descriptor and its application in loop closure detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 231–237.

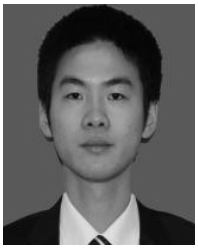
[48] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[49] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," Jan. 2018, *arXiv:1801.07829*. [Online]. Available: https://arxiv.org/abs/1801.07829

Zheng Gong received the M.Sc. degree in computer science and engineering from Fuzhou University, China, in 2015, and the Ph.D. degree with the Department of Systems Design Engineering, University of Waterloo, Canada, in 2019. He is currently pursuing the Ph.D. degree in information and communications engineering with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Information Science and Engineering, Xiamen University, China. He has published an article at the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT. He has coauthored several research article published in refereed journals. He has published in flagship conferences such as ISPRS, IGARSS, and MMT. His current research interests include backpack LiDAR systems, multisensor calibration, multisource data registration, point cloud processing, indoor mapping, 3D computer vision, and 3D space search.

Jonathan Li (M'00–SM'10) received the Ph.D. degree in geomatics engineering from the University of Cape Town, South Africa.

He is currently with the Fujian Key Lab of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, and a Professor with the Department of Geography and Environmental Management, and cross-appointed with the Department of Systems Design Engineering, University of Waterloo, Canada. He has coauthored more than 400 publications, more than 180 of which were published in refereed journals including the IEEE TGRS, IEEE TITS, IEEE GRSL, IEEE JSTARS, ISPRS JPRS, IJRS, PE&RS, and RSE. His current research interests include information extraction from LiDAR point clouds and earth observation images. He was a recipient of more than a dozen prestigious awards including the Outstanding Achievement Award in Mobile Mapping Technology from MMT 2019. He is the Chair of the ISPRS Working Group I/2 on LiDAR. He has been the Chair of the Air- and Spaceborne- Optical Sensing since 2016 and the ICA Commission on Sensor-Driven Mapping since 2015. He was an Associate Editor of the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING, and the *Canadian Journal of Remote Sensing*.



Zhipeng Luo received the B.Sc. degree in mathematics from Minnan Normal University, China, in 2013, and the M.Sc. degree in computer science from Fuzhou University, China, in 2016. He is currently pursuing the Ph.D. degree in information and communications engineering with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Information Science and Engineering, Xiamen University, China. He has coauthored several research articles published in refereed journals such as *ISPRS Journal of Photogrammetry and Remote Sensing*. He has published in flagship conferences such as CVPR and IGARSS. His current research interests are 3D computer vision, 3D point cloud analysis, object recognition, semantic annotation, road modeling, and deep learning.

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, Xiamen University. She has coauthored over 50 research articles published in refereed journals and conference proceedings. Her research interests include point cloud processing, 3D indoor mapping and modeling, and robotics. She is an Associate Editor of the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS and the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS.

Cheng Wang (M'04–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, Xiamen University, China, where he is also the Executive Director of the Fujian Key Laboratory of Sensing and Computing for Smart Cities. He has coauthored over 80 articles in refereed journals, including the IEEE TGRS, the IEEE TITS, the IEEE GRSL, the IEEE JSTARS, IJRS, and ISPRS JPRS. His research interests include remote sensing image processing, mobile LiDAR data analysis, and multisensory data fusion. He has been a Council Member of the China Society of Image and Graphics. He has been the Chair of the ISPRS WG I/3 on Multi-Platform Multi-Sensor System Calibration since 2016. He has been a fellow of the IET.

John Zelek received the Ph.D. degree in philosophy of electrical engineering from the Centre for Intelligent Machines (CIM), McGill University, Montreal, QC, Canada, in 1996.

He is currently a Professor and the Associate Graduate Chair of the Systems Design Engineering Department, University of Waterloo, Waterloo, ON, Canada. His research focus is on producing 3D maps from cameras and depth sensors. These 3D maps can be used by robots (vehicles and UAVs) to navigate and localize or by people for various augmented applications. His current interests include the Internet of Things (IOT) within the context of artificial perception, wearables, and augmented spaces.