MOFIS$_{\text{SLAM}}$: A *M*ulti-*O*bject Semantic SLAM System With *F*ront-View, *I*nertial, and *S*urround-View Sensors for Indoor Parking

Xuan Shao, Lin Zhang, *Senior Member, IEEE*, Tianjun Zhang, Ying Shen, *Member, IEEE*, and Yicong Zhou, *Senior Member, IEEE*

***Abstract*— The semantic SLAM (Simultaneous Localization And Mapping) system is a crucial module for autonomous indoor parking. Visual cameras (monocular/binocular) and IMU (Inertial Measurement Unit) constitute the basic configuration to build such a system. The performance of existing SLAM systems typically deteriorates in the presence of dynamically movable objects or objects with little texture. By contrast, semantic objects on the ground embody the most salient and stable features in the indoor parking environment. Due to their inabilities to perceive such features on the ground, existing SLAM systems are prone to tracking inconsistency during navigation. In this paper, we present MOFIS$_{\text{SLAM}}$, a novel tightly-coupled *M*ulti-*O*bject semantic SLAM system integrating *F*ront-view, *I*nertial, and *S*urround-view sensors for autonomous indoor parking. The proposed system moves beyond existing semantic SLAM systems by complementing the sensor configuration with a surround-view system capturing images from a top-down viewpoint. In MOFIS$_{\text{SLAM}}$, apart from low-level visual features and inertial motion data, typical semantic objects (parking-slots, parking-slot IDs and speed bumps) detected in surround-views are also incorporated in optimization, forming robust surround-view constraints. Specifically, each surround-view feature imposes a surround-view constraint that can be split into a contact term and a registration term. The former pre-defines the position of each individual surround-view feature subject to whether it has semantic contact with other surround-view features. Three contact modes, defined as *complementary*, *adjacent* and *coincident*, are identified to guarantee a unified form of all contact terms. The latter further constrains by registering each surround-view observation and its position in the world coordinate system. In parallel, to objectively evaluate SLAM studies for autonomous indoor parking, a large-scale dataset with groundtruth trajectories is collected, which is the first of its kind. Its groundtruth trajectories, commonly unavailable, are obtained by tracking artificial features scattered in the indoor parking environment, whose 3D coordinates are measured with an ETS (Electronic Total Station). The collected dataset has been made publicly available at https://shaoxuan92.github.io/MOFIS.**

***Index Terms*— Semantic SLAM, surround-view system, autonomous indoor parking, groundtruth trajectory.**

I. Introduction

In ORDER to realize fully autonomous indoor parking, a SLAM (Simultaneous Localization And Mapping) system is indispensable. It aims to create three-dimensional representations of an unknown environment and track the location of the vehicle during navigation with a strong focus on real-time operation. There are different approaches to this task according to the parking environment. In an outdoor parking environment, since a highly reliable information source like GPS (Global Positioning System) is available, the additional dependence of other on-board sensors of the vehicle could be decreased in SLAM systems for outdoor parking. As the differential GPS can provide high-precision vehicle localization results in the outdoor environment, the autonomous parking system only needs to transform the coordinates of parking-slots into the same coordinate system of GPS. As the poor coverage of satellite signals caused by occlusions weakens the performance of GPS-based approaches, a SLAM system for indoor parking is normally built simply with the on-board sensors of the vehicle. In this article, we focus on SLAM studies for the indoor parking environment.

Various advanced SLAM systems [1]–[5] in this field have already attained satisfactory performance based on different sensor modalities including camera, IMU (Inertial Measurement Unit) and laser scanner, *etc.* Particularly, VI-SLAM (Visual-Inertial SLAM) systems with visual cameras and IMUs, have gained considerable popularity in the past decade due to inherently complementary properties of these two sensor modalities. While an IMU is responsive in short-term dynamics, one camera provides rich and consistent exteroceptive information for long-term navigation. Specifically, an IMU measures the motion of the vehicle in the environment lacking of enough textures for visual tracking,

Manuscript received September 11, 2021; revised November 11, 2021 and November 27, 2021; accepted December 9, 2021. Date of publication December 13, 2021; date of current version July 5, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61973235 and Grant 61972285, in part by the Natural Science Foundation of Shanghai under Grant 19ZR1461300, in part by the Shanghai Science and Technology Innovation Plan under Grant 20510760400, in part by the Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0100, in part by the Dawn Program of Shanghai Municipal Education Commission under Grant 21SG23, and in part by the Fundamental Research Funds for the Central Universities. This article was recommended by Associate Editor L. K. Choi. *(Corresponding author: Lin Zhang.)*

Xuan Shao, Lin Zhang, Tianjun Zhang, and Ying Shen are with the School of Software Engineering, Tongji University, Shanghai 201804, China (e-mail: 1810553@tongji.edu.cn; cslinzhang@tongji.edu.cn; 1911036@tongji.edu.cn; yingshen@tongji.edu.cn).

Yicong Zhou is with the Department of Computer and Information Science, University of Macau, Macau, China (e-mail: yicongzhou@um.edu.mo).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2021.3135041.

Digital Object Identifier 10.1109/TCSVT.2021.3135041

whereas the camera offers consistent visual observations to reduce trajectory bias of IMU motion data with lower price and simplicity to calibrate. The VI-SLAM systems [6], [7] can be expected to surpass the state-of-the-art visual SLAM systems in universality and robustness in different kinds of environments. However, the performance of these VI-SLAM systems usually deteriorates in the presence of illumination changes and repetitive patterns. Moreover, they construct maps merely with geometric information, failing to provide any high-level semantic understanding essential for autonomous indoor parking. In order to acquire a semantic understanding of their surrounding environment, the semantic SLAM systems attempt to incorporate semantic information to build meaningful maps that have both metric (orientation, position) and semantic (cars, people, *etc.*) representations of the scene. While existing semantic SLAM systems have been successfully demonstrated in specific circumstances, unexpected changes of surroundings would probably degrade the quality of the generated map and even lead to tracking failure. For instance, the presence of dynamics in the environment [8], like a moving car or pedestrian, might corrupt the quality of the state estimation by deceiving feature association in SLAM systems. Nevertheless, semantic objects on the ground (parking-slots, speed bumps and parking-slot IDs) are stable and salient features for the specific application scenario of autonomous indoor parking, exhibiting strong semantic consistency. Unfortunately, thus far, few eminent semantic SLAM systems have fully explored these features. Additionally, a large-scale benchmark dataset with groundtruth trajectories is a must for objective evaluation of different SLAM systems. Due to a lack of GPS signal in the indoor parking environment, the groundtruth trajectories are commonly unavailable in current datasets for autonomous indoor parking.

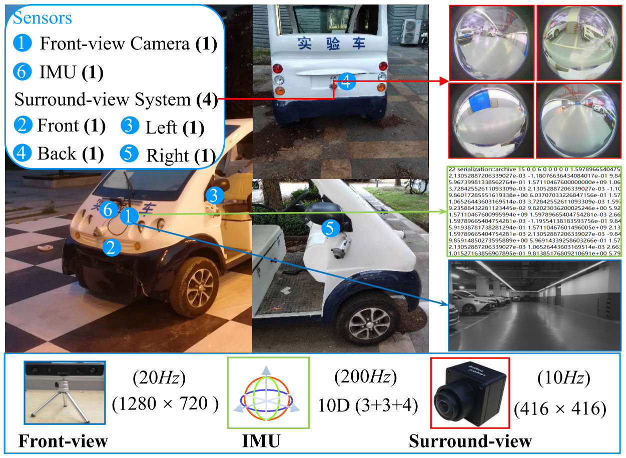


Fig. 1. Sensor setup of our MOFIS$_{\text{SLAM}}$. The sensor configuration of the car consists of a front-view camera, an IMU and four fisheye cameras forming a surround-view camera system to provide a surround-view image. All the sensors were connected to a lap-top with an Intel (R) Core (TM) i7-6700HQ CPU, 16 GB memory and an Nvidia (R) Quadro (R) M1000M GPU via a USB 3.0 hub.

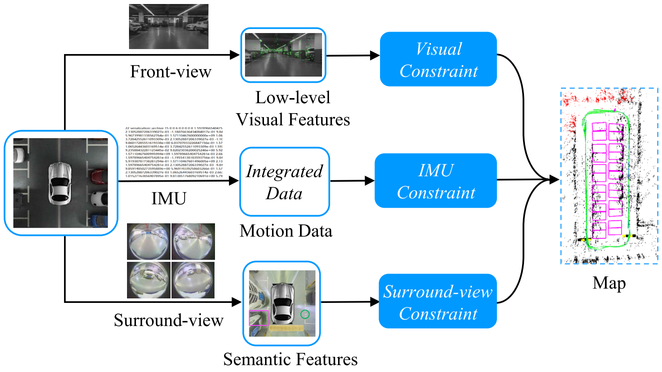


Fig. 2. The architecture of MOFIS$_{\text{SLAM}}$. Front-view low-level visual features, motion data from IMU, and semantic features (parking-slots, speed bumps, parking-slot IDs) in the surround-views are collectively combined, providing both accurate localization and semantic mapping result. Top view of the map built by MOFIS$_{\text{SLAM}}$ is shown on the right side of the figure. Parking-slot IDs are omitted for display.

As mentioned above, in the field of autonomous indoor parking, a highly mature, reliable SLAM system and an appropriate benchmark dataset are still lacking. In this paper, we attempt to address these issues by developing a semantic SLAM system, namely MOFIS$_{\text{SLAM}}$ (a *M*ulti-*O*bject semantic SLAM system with *F*ront-view, *I*nertial, and *S*urround-view sensors) and constructing a large-scale dataset for indoor parking. The proposed SLAM system is deployed on an electric car. The sensor configuration of the car comprises two sensor modalities, the perception sensor and the navigation sensor. The perception sensor provides the perception of the surrounding environment and motion data is given by the navigation sensor. Specifically, the perception sensor consists of a front camera and a surround-view system with four fisheye cameras mounted around the vehicle. The navigation sensor consists of an IMU, providing temporally synchronized navigation data reflecting the motion of the vehicle. As seen in Fig. 1, five brackets were customized to fix the front-view camera and four fisheye cameras in the surround-view system, respectively. The orientation of each fisheye camera is about 45 degrees ground-oriented, capturing images of the ground around the vehicle. The front-view camera was fixed higher than the front-view camera in the surround-view system, facing straight ahead to ensure a broad view. All the sensors were connected to a lap-top with an Intel (R) Core (TM) i7-6700HQ CPU, 16 GB memory and an Nvidia (R) Quadro (R) M1000M GPU via a USB 3.0 hub.

Our main contributions can be summarized as follows:

(1) We propose a tightly-coupled semantic SLAM system MOFIS$_{\text{SLAM}}$, leveraging front-view, inertial, and surround-view sensor modalities, specially for the task of autonomous indoor parking as illustrated in Fig. 2. MOFIS$_{\text{SLAM}}$ is the first VI-SLAM system attempting to make full use of various semantic objects detected in surround-views. Because of the integration of surround-view objects, the constructed map carries crucial semantic information that facilitates both intuitive visualization and high-level understanding of the surrounding environment. It has to be noted that MOFIS$_{\text{SLAM}}$ is designed for autonomous parking in an indoor environment that lacks the GPS data source. It merely relies on the on-board sensors of the vehicle to achieve accurate localization results. For autonomous parking in an outdoor

environment, a more mature, stable and low-priced scheme is usually adopted. As the differential GPS can provide high-precision localization results in the outdoor environment, there is no need for a front-view camera or even an IMU.

(2) In MOFIS$_{\text{SLAM}}$, common salient objects (parking-slots, speed bumps and parking-slot IDs) extracted from surround-views ensure the tracking consistency over the long-term navigation. These objects are exploited in optimization to form a robust surround-view error term with both prior and observational constraints. Specifically, three contact modes among surround-view features, *adjacent*, *complementary* and *coincident*, are identified to guarantee a unified form of the prior constraints for all surround-view features. We experimentally validate that the proposed optimization strategy enables both higher localization accuracy and semantically meaningful scene representations (Please refer to Sect. VII for details).

(3) To fairly and objectively evaluate the performance of various SLAM systems developed for autonomous indoor parking, we establish a large-scale benchmark dataset with available groundtruth trajectories, which is the first of its kind. In this dataset, the groundtruth trajectories during navigation are obtained by tracking artificial features scattered in the indoor parking environment, whose coordinates are recorded in a surveying manner with a high-precision ETS (Electronic Total Station). The dataset has been publicly available to the research community at https://shaoxuan92.github.io/MOFIS.

The results of this manuscript were partially reported in ACM MM 2020 [9]. The following improvements are made in this version. 1) We present a unified optimization framework, in which semantic surround-view landmarks of any types can be modeled and exploited, not limited to parking-slots, parking-slot IDs and speed bumps. Qualitative and quantitative experiments corroborate the superiority of the proposed framework compared with its counterpart in [9]. 2) We propose an effective yet cost-efficient groundtruth trajectory acquisition approach simply with a mild intervention of the indoor parking environment. Specifically, the groundtruth trajectories are obtained by tracking AprilTags pasted on walls/pillars, whose coordinates are measured with an ETS. 3) In order to facilitate SLAM studies for autonomous indoor parking, we establish a large-scale benchmark dataset comprising synchronous multi-sensor data collected from a typical indoor parking site. With the proposed groundtruth trajectory acquisition scheme, the groundtruth trajectories are also provided in the dataset. One point needs to be noted is that without the groundtruth trajectories, the performance evaluation of SLAM systems would not be so reliable. The collected dataset is now publicly released to benefit the other researchers.

The remainder of this paper is structured as follows. Sect. II summarizes the related work. The overall framework of MOFIS$_{\text{SLAM}}$ is presented in Sect. III. Details for system implementation, sensor calibration and the collected dataset are introduced in Sect. IV, Sect. V and Sect. VI, respectively. The experimental results are reported in Sect. VII and finally, Sect. VIII concludes the paper.

II. Related Work

A. Scene Representation for Autonomous Parking

Recent years have witnessed a growing interest in developing scene representation approaches [10]–[15]. Gao *et al.* proposed a three-step pipeline to achieve scene reconstruction by merging images and laser scans in a coarse-to-fine manner [10]. Michailidis *et al.* developed a novel customized hardware-based reconstruction architecture for time-critical applications [11]. Kim *et al.* presented a multi-view segmentation-based framework for separately reconstructing background and foreground [12]. However, since these approaches fail to build the semantic map of the scene in real time, they cannot be applied to the autonomous parking task. As an attempt to solve the above issue, researchers resort to road markings (signs marked on the road) towards developing VI-SLAM systems for autonomous driving. Typically studied road markings include lane lines, curbs, markers, *etc*. Compared with traditional features, road markings are distinct and fairly abundant on the road whose detection is less susceptible to lighting changes [16], [17]. Schreiber *et al.* [13] proposed a system to incorporate markers and curbs detected online for the precise localization and mapping. Since their system determines the current localization by matching these road markings with a prior map built by extended sensor setups (laser scanner and GPS unit, *etc.*), its localization accuracy is limited by the map accuracy. In [14], Ranganathan *et al.* incorporated distinct road markings to perform pose estimation by solving a windowed bundle adjustment problem. In their work, the pose was estimated based on the premise that road markings are standardized and their sizes are fixed and known, and thus the main limitation of Ranganathan *et al.*'s approach is the high ambiguity caused when the markings and the lanes have similar shapes and repetitive patterns. A more appealing scheme, RoadSLAM, was presented in [15] by Jeong *et al.*, where only the distinguishable road markings with informative features classified by the random forest were used in both localization and loop detection modules. Unfortunately, Jeong *et al.* mentioned that their system was sensitive to the shadow of surrounding objects in some cases.

To address the instability of aforementioned SLAM systems, panoramic surround-view images with metric scale are utilized in recent SLAM studies [9], [18]. The surround-view image is stitched by four bird's-eye view images from a surround-view system [19]. In one surround-view image, semantic objects on the ground can be stably and consistently detected despite of changing perspectives and lighting conditions. Zhao *et al.* detected parking-slots in the surround-view images and incorporated them into the SLAM system they built [18]. However, in their work, artificial landmarks were used to facilitate localization, whereas parking-slots contributed little for optimization. To the best of our knowledge, the latest system that leveraged features detected on the ground is the one reported in [9]. In [9], Shao *et al.* proposed a SLAM system where parking-slots in the surround-view images are incorporated during optimization. However, surround-view features selected in [9] are parking-slot specific,

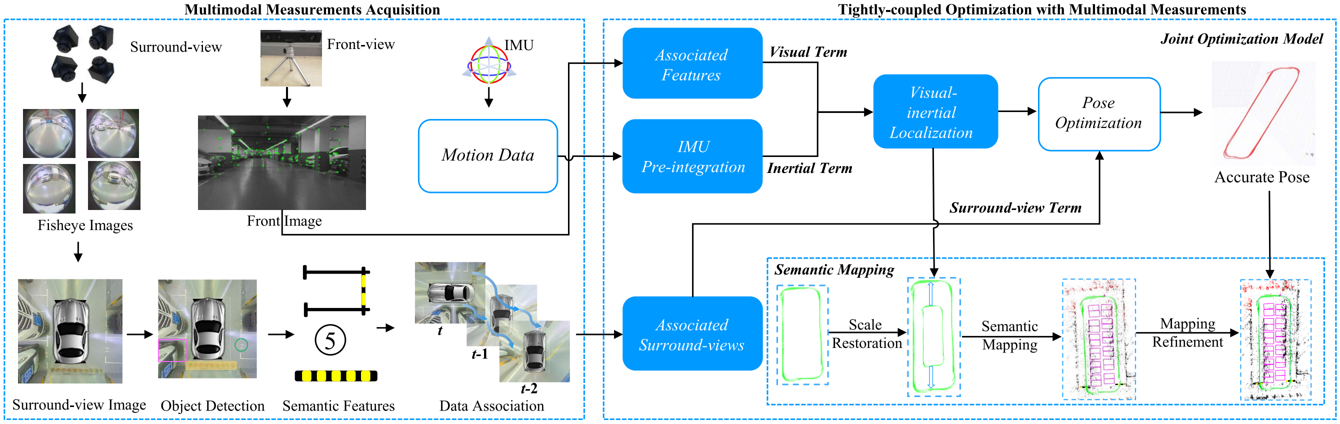


Fig. 3. The overall framework of MOFIS$_{\text{SLAM}}$. Sensor configuration of MOFIS$_{\text{SLAM}}$ consists of a front-view camera, an IMU and four fisheye cameras facing the ground to form a surround-view system. Apart from front-view visual features and IMU motion data, semantic features in the surround-views, including parking-slots, speed bumps and parking-slot IDs, are detected and geometrically associated. The visual, inertial as well as the surround-view terms are integrated into MOFIS$_{\text{SLAM}}$ during optimization for accurate localization and semantic mapping suitable for autonomous indoor parking.

resulting in tracking inconsistency in circumstances where parking-slots are occluded by a parked car. Moreover, the property of two neighboring parking-slots used in their system are scenario specific, rather than being completely general when it comes to different indoor parking environments where there are no neighboring parking-slots.

B. VI-SLAM Datasets

To ensure objective evaluation of estimated camera poses, several datasets with groundtruth trajectories were established [20]–[25]. Among them, Urban@CRAS [20] and KAIST Urban [21] are two typical outdoor datasets. Sequences in the Urban@CRAS dataset were acquired in a sunny day of several scenarios including a coastal zone, avenues, roundabouts and highways. Different traffic elements (such as moving cars, motorcycles and pedestrians) with light changes, different scenes (urban, highway and coastal zones) and dynamic elements were considered in this dataset. The KAIST Urban dataset was acquired in four different cities with abundant dynamic objects and traffic lights. Sequences in this dataset were collected in urban environments such as urban canyon, wide multi-lane road, high-rise buildings and densely cluttered residential area. In both datasets, groundtruth trajectories were provided in leverage of GPS. But GPS is not available for indoor parking environments. The datasets [22]–[25] are four popular indoor datasets, the groundtruth trajectories of which were recorded. The EgoCart dataset [22] comprises 19,531 RGB images along with depth maps considering the task of localizing shopping carts in a retail store from egocentric images. The groundtruth trajectory of the dataset was acquired using structure from motion algorithms. The EuRoC MAV dataset [23] includes indoor sequences recorded with a Skybotix stereo VI sensor from a MAV (Micro Aerial Vehicle). Some sequences in this dataset were recorded in a large machine hall that is unstructured and cluttered with different flight dynamics and lighting conditions. Others were recorded in one room with an approximate size of $8m\times8.4m\times4m$, where moving curtains and different obstacle configurations under good visual conditions existed. The Oxford Multimotion dataset [24] was collected in an experimental room equipped with professional flicker-free lighting. It contains a varying number of moving bodies, aiming at providing a benchmark for motion estimation of moving objects as well as vehicle self-localization. The dataset provided in [25] contains sequences with camera motions along a corridor, sequences featuring a walk around the central hall in a university building, and so on. In these datasets [23]–[25], groundtruth poses were recorded at 100-200Hz by a motion capture system and were accurately time-aligned with the sensor measurements as well. However, the motion capture system is costly and its coverage capability is limited. Moreover, none of these datasets involve scenarios typically encountered in autonomous indoor parking.

To sum up, there is currently no existing VI-SLAM dataset with groundtruth trajectories for autonomous indoor parking. The current groundtruth trajectory acquisition approaches are unsuitable in GPS-denied indoor parking environments or fail to guarantee the integrity of the trajectory. This paper seeks to provide a large-scale benchmark dataset with groundtruth trajectories to facilitate SLAM studies for autonomous indoor parking. The groundtruth trajectories are obtained by taking advantage of an ETS, which is both affordable and applicable in the indoor parking environments.

III. MOFIS$_{\text{SLAM}}$

The overall architecture of MOFIS$_{\text{SLAM}}$ is illustrated in Fig. 3. Sensor configuration of MOFIS$_{\text{SLAM}}$ consists of a front-view camera, an IMU and four fisheye cameras facing the ground to form a surround-view camera system. There are two major components in MOFIS$_{\text{SLAM}}$, the multimodal measurements acquisition module and the joint optimization module. The former one is responsible for multimodal data association. In this module, common semantic features in the surround-views are detected and geometrically associated, whose details will be introduced in Sect. IV. These surround-view features, together with visual features from the front-view camera and pre-integrated IMU measurements between two consecutive keyframes, constitute the multimodal sensor measurements for MOFIS$_{\mathrm{SLAM}}$. The latter joint optimization module waits for these associated multimodal measurements to tightly fuse them, which is the core of MOFIS$_{\mathrm{SLAM}}$. Its details will be thoroughly presented in this section with regard to its formulation and all error terms during optimization.

A. Joint Optimization Model Formulation

We first introduce the measurements and unknowns in the optimization model. Given keypoints $\mathcal{Z}$ in the front-view image, IMU measurements $\mathcal{M}$ and semantic observations $\mathcal{O}$ in the surround-view image, the proposed joint optimization model for MOFIS$_{\mathrm{SLAM}}$ determines optimal camera poses $\mathcal{T}$, map points $\mathcal{P}$ matched with $\mathcal{Z}$ as well as surround-view landmarks $\mathcal{L}$, jointly. We define a MAP (Maximum A Posteriori) problem for all variables and such an optimization problem can be casted as,

$$\{\mathcal{T}, \mathcal{P}, \mathcal{L}\}^* = \arg \max_{\mathcal{T},\mathcal{P},\mathcal{L}} p(\mathcal{T}, \mathcal{P}, \mathcal{L} | \mathcal{Z}, \mathcal{M}, \mathcal{O}). \tag{1}$$

Since keypoints $\mathcal{Z}$, IMU measurements $\mathcal{M}$, and surround-view observations $\mathcal{O}$ are independently observed by different sensor modalities, p can be factorized by separating these measurements from one another, i.e.,

$$\begin{aligned}
& p(\mathcal{T}, \mathcal{P}, \mathcal{L} | \mathcal{Z}, \mathcal{M}, \mathcal{O}) \\
& \propto p(\mathcal{T}, \mathcal{P}, \mathcal{L}) p(\mathcal{Z}, \mathcal{M}, \mathcal{O} | \mathcal{T}, \mathcal{P}, \mathcal{L}) \\
& = p(\mathcal{T}, \mathcal{P}, \mathcal{L}) p(\mathcal{Z} | \mathcal{T}, \mathcal{P}, \mathcal{L}) p(\mathcal{M} | \mathcal{T}, \mathcal{P}, \mathcal{L}) p(\mathcal{O} | \mathcal{T}, \mathcal{P}, \mathcal{L}) \\
& = p(\mathcal{T}) p(\mathcal{P}) p(\mathcal{L}) p(\mathcal{Z} | \mathcal{T}, \mathcal{P}, \mathcal{L}) p(\mathcal{M} | \mathcal{T}, \mathcal{P}, \mathcal{L}) p(\mathcal{O} | \mathcal{T}, \mathcal{P}, \mathcal{L}) \\
& = p(\mathcal{T}) p(\mathcal{P}) p(\mathcal{L}) p(\mathcal{Z} | \mathcal{T}, \mathcal{P}) p(\mathcal{M} | \mathcal{T}) p(\mathcal{O} | \mathcal{L}, \mathcal{T}) \\
& = \underbrace{p(\mathcal{T}) p(\mathcal{P})}_{prior} \underbrace{p(\mathcal{Z} | \mathcal{T}, \mathcal{P}) p(\mathcal{M} | \mathcal{T})}_{visual-inertial\ term} \underbrace{\overbrace{p(\mathcal{L})}^{prior} \overbrace{p(\mathcal{O} | \mathcal{T}, \mathcal{L})}^{observation}}_{surround-view\ term},
\end{aligned} \tag{2}$$

where the first two terms, $p(\mathcal{T})$ and $p(\mathcal{P})$, model the priors for camera poses and map points, respectively. We assume that both priors are uniformly distributed, thereby being converted into a constant factor $\mathbf{C}$. The middle two terms, $p(\mathcal{Z}|\mathcal{T}, \mathcal{P})$ and $p(\mathcal{M}|\mathcal{T})$, are relevant to front-view visual data and IMU motion measurements, constraining camera poses and map points by observed visual features and motion data. Following [6], [9], the visual-inertial term can be converted into a visual error term and an inertial error term, $\mathbf{E}_V$ and $\mathbf{E}_I$, respectively. Specifically, $\mathbf{E}_V$ links each keypoint and its projecting map point while $\mathbf{E}_I$ constrains consecutive keyframes by visual-inertial alignment, stably predicting reliable camera poses and map points. The last two terms, $p(\mathcal{L})$ and $p(\mathcal{O}|\mathcal{T}, \mathcal{L})$, define the surround-view error term. Salient semantic objects (parking-slots, speed bumps and parking-slot IDs) in surround-views encode abundant information such as the class, the location and the detection confidence, imposing a surround-view constraint $\mathbf{E}_S$. Therefore, in order to find out optimal estimation, we jointly optimize visual, inertial and surround-view error terms in a tightly-coupled objective,

$$\{\mathcal{L}, \mathcal{T}, \mathcal{P}\}^* = \arg \min_{\mathcal{L},\mathcal{T},\mathcal{P}} \mathbf{E}_V + \mathbf{E}_I + \mathbf{E}_S + \mathbf{C}. \tag{3}$$

Intuitively, with Eq. (3), MOFIS$_{\mathrm{SLAM}}$ is optimized by jointly minimizing errors of visual re-projection error, IMU motion error and surround-view error taking into account surround-view features. In this way, MOFIS$_{\mathrm{SLAM}}$ deals with both low-level geometric/motion data as well as salient and stable semantic objects on the ground, simultaneously. It enables robust perception of an indoor parking environment, avoiding the vulnerability to blur, dramatic lighting changes, and low-texture conditions as in the conventional SLAM systems. It has to be noted that feature points detected in front-view images mainly come from the vehicles and pillars ahead of the camera or the ceiling of the indoor parking site. Only a small number of front-view feature points are extracted from semantic objects on the ground. Thus, we think that connections between front-view feature points and surround-view objects are relatively quite weak. Therefore, in MOFIS$_{\mathrm{SLAM}}$ the front-view feature points and surround-view objects are optimized independently. Three error terms appearing in Eq. (3), $\mathbf{E}_V$, $\mathbf{E}_I$ and $\mathbf{E}_S$, are detailed in the subsequent subsections.

B. Visual Error Term

The visual error term ${}_v\mathbf{e}_{kn}$ involving the n-th map point $\mathbf{P}_n$ and the front-view camera pose $\mathbf{T}_k \in SE(3)$ of the k-th keyframe is defined as the reprojection error with respect to the matched observation $\mathbf{z}_k^n$, i.e.,

$${}_v\mathbf{e}_{kn} = \mathbf{z}_k^n - \phi_k(\mathbf{T}_k, \mathbf{P}_n), \tag{4}$$

where $\phi_k(\cdot)$ is the projection function of the front-view camera at the time when taking the k-th keyframe. Given the set of camera poses $\mathcal{T} = \{\mathbf{T}_k\}_{k=1}^K$ and map points $\mathcal{P} = \{\mathbf{P}_n\}_{n=1}^N$, $\mathbf{E}_V$ tackles the problem of jointly optimizing camera poses $\mathcal{T}$ and map points $\mathcal{P}$, i.e.,

$$\mathbf{E}_V = \sum_{k=1}^{K} \sum_{n=1}^{N} \rho_h({}_v\mathbf{e}_{kn}^T \ \Lambda_{kn}^{-1} \ {}_v\mathbf{e}_{kn}), \tag{5}$$

where $\rho_h(\cdot)$ is the Huber kernel function for robustness to outliers and $\Lambda_{kn} = \sigma_{kn}^2 \mathbf{I}_{2\times 2}$ is covariance matrix associated to the scale at which the keypoint is detected.

C. IMU Error Term

The motion (orientation, velocity, position) between two consecutive keyframes can be determined by the pre-integrated IMU data. Each IMU error term ${}_m\mathbf{e}_{ij}$ links the i-th and the j-th keyframes, i.e.,

$${}_m\mathbf{e}_{ij} = [{}_R e_{ij} \ \ {}_V e_{ij} \ \ {}_P e_{ij}]^T, \tag{6}$$

where ${}_R e_{ij}$, ${}_V e_{ij}$ and ${}_P e_{ij}$ denote the orientation, the velocity, and the position error terms between consecutive keyframes, respectively. Accordingly, $\mathbf{E}_I$ is defined as,

$$\mathbf{E}_I = \sum_{i=1}^{K} \rho_h({}_m\mathbf{e}_{ij}^T \ \Sigma_i \ {}_m\mathbf{e}_{ij}), \tag{7}$$

where Σ_i is the information matrix determined using the scheme introduced in [6].

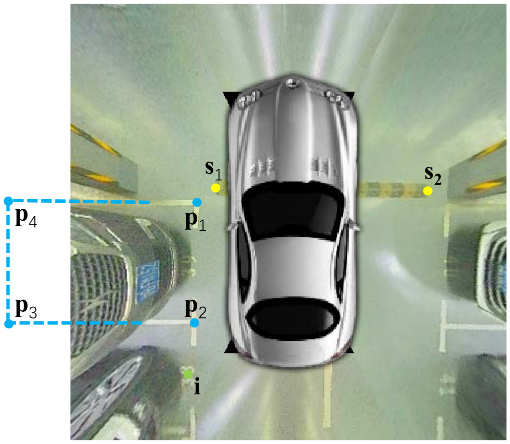


Fig. 4. Mathematical representations of semantic objects of a parking-slot, a parking-slot ID and a speed bump.

D. Surround-View Error Term

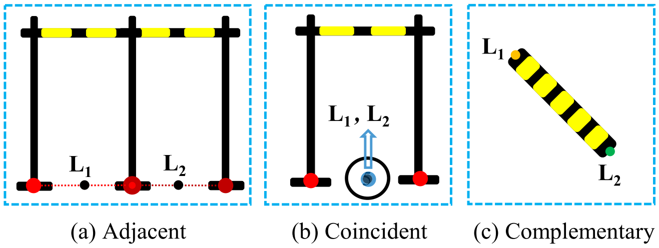
The surround-view error term $\mathbf{E}_S$ is built based on semantic objects detected in surround-view images. According to Eq. (2), $\mathbf{E}_S$ can be split into a contact error term and a registration error term corresponding to $p(\mathcal{L})$ and $p(\mathcal{O}|\mathcal{T},\mathcal{L})$, respectively. The contact error term is denoted by $\mathbf{E}_{Con}$. It predefines the position of each individual surround-view landmark subject to whether it has semantic contact with other surround-view landmarks. The registration error term $\mathbf{E}_{Reg}$ further constrains by registering each observation and its position in the world coordinate system. Therefore, $\mathbf{E}_S$ can be defined as,

$$\mathbf{E}_S = \mathbf{E}_{Con} + \mathbf{E}_{Reg}. \tag{8}$$

1) Semantic Object Representation: As shown in Fig. 4, three categories of semantic objects are considered, parking-slots ($\mathbb{P}$), speed bumps ($\mathbb{S}$) and parking-slot IDs ($\mathbb{I}$). According to the definition in the work [26], each parking-slot is represented as $\{\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3,\mathbf{p}_4\}$, where $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$ and $\mathbf{p}_4$ are the coordinates of its four vertices; $\mathbf{p}_1$ and $\mathbf{p}_2$ are the coordinates of the two vertices forming the entrance-line of the parking-slot and the four vertices are arranged in a clockwise manner. Similarly, the speed bump is represented as $\{\mathbf{s}_1,\mathbf{s}_2\}$, where $\mathbf{s}_1$ and $\mathbf{s}_2$ are the coordinates of its two endpoints. And the parking-slot ID is represented as $\mathbf{i}$, the coordinate of the parking-slot ID's center. For optimization, each semantic object can be further abstracted into one or two semantic landmarks. Each semantic landmark is represented with a four-dimensional vector $[\mathbf{L}, W]$, where $\mathbf{L}$ is its 3D position and W is its width. Definitions of the position and the width of each semantic landmark are illustrated in Fig. 5 and Fig. 6. For example, the position and the width of a parking-slot are defined as the center and the width of its entrance-line, respectively.

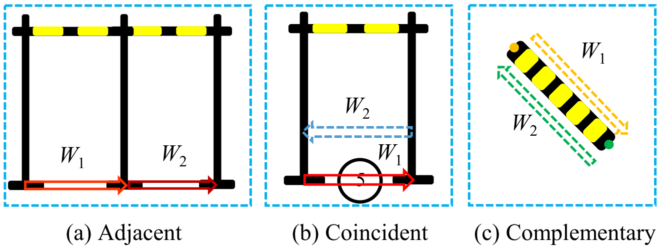
2) Contact Error Term: $p(\mathcal{L})$ models the prior distribution for positions of all surround-view landmarks, i.e.,

$$p(\mathcal{L}) = \prod_{t=1}^{T} p(\mathcal{L}_{y_t}), \quad \mathcal{L}_{y_t} = \{\mathbf{L}_{y_t^i}\}_{i=0}^{N_t}, \tag{9}$$



(a) Adjacent (b) Coincident (c) Complementary

Fig. 5. Semantic-contact modes of surround-view semantic objects. According to the combination of surround-view objects with different abstractions, three semantic-contact modes are identified as *adjacent*, *complementary* and *coincident*.



(a) Adjacent (b) Coincident (c) Complementary

Fig. 6. The definition of the width of each surround-view landmark. By defining the widths of a parking-slot ID and a speed bump endpoint (dashed arrows), all surround-view landmarks are represented with the same geometric properties. Specifically, a parking-slot ID's width is defined as the opposite of the width of the parking-slot it represents, and the width of a speed bump endpoint is defined by the length of the corresponding speed bump.

where $p(\mathcal{L}_{y_t})$ and N_t are the prior of the positions and the total number of surround-view landmarks at time t, respectively. $\mathbf{L}_{y_t^i}$ is the position of the i-th surround-view landmark at time t. Data association of each observation is denoted by $y_t^i \in \{1;\ldots;M\}$. M is the total number of surround-view landmarks in the indoor parking site. According to whether a surround-view landmark has contact with other surround-view landmarks, surround-view landmarks can be accordingly categorized into two groups, the semantic-contact group and the stand-alone one. Since each group is independent with one another, $p(\mathcal{L})$ can be reformulated as,

$$\begin{aligned} p(\mathcal{L}) &= \prod_{t=1}^{T} p(\mathcal{L}_{y_t}) \\ &\propto \prod_{t=0}^{T} (\prod_{i=0}^{n-1} p(\mathbf{L}_{y_t^i}\mathbf{L}_{(\widetilde{y_t^i})}) \prod_{i=n}^{N_t} p(\mathbf{L}_{y_t^i})) \\ &= \prod_{t=0}^{T} (\underbrace{\prod_{i=0}^{n-1} p(\mathbf{L}_{(\widetilde{y_t^i})}) p(\mathbf{L}_{y_t^i}|\mathbf{L}_{(\widetilde{y_t^i})})}_{semantic-contact} \underbrace{\prod_{i=n}^{N_t} p(\mathbf{L}_{y_t^i})}_{stand-alone}), \end{aligned} \tag{10}$$

where we assume there are n semantic-contact pairs. $\widetilde{y_t^i}$ denotes the id of the semantic-contact landmark of the i-th surround-view landmark at time t. Thus, the contact error term of each surround-view landmark is defined subject to whether it has a semantic-contact neighbor, i.e.,

$$\begin{aligned} &p(\mathbf{L}_{y_t^i}) \sim \mathcal{U}, & \widetilde{y_t^i} \notin \mathcal{M}_t \\ &p(\mathbf{L}_{y_t^i}|\mathbf{L}_{\widetilde{y_t^i}}) \sim \mathcal{N}(f(\widetilde{y_t^i}), \Lambda_{i,t}), & \widetilde{y_t^i} \in \mathcal{M}_t, \end{aligned} \tag{11}$$

TABLE I
CONTACT MODES OF SURROUND-VIEW LANDMARKS

Mode	Instance	Property
Adjacent	$\mathbb{P} \parallel \mathbb{P}$	A and B are adjacent.
Complementary	$\mathbb{S} \parallel \mathbb{S}$	A and B constitute C.
Coincident	$\mathbb{I} \parallel \mathbb{P}$	A stands where B is.

where all surround-view landmarks in the map at time t are denoted by $\mathcal{M}_t$. $\mathcal{U}$ is a uniform distribution and $\mathcal{N}(\ .\ ,\ .\ )$ represents a normal distribution. $f(\widetilde{y_t^i})$ is the position of a surround-view landmark induced by its semantic-contact neighbor. $\Lambda_{i,t}$ models the uncertainty. Intuitively, if one surround-view landmark stands alone with no semantic-contact neighbor, its position is distributed with equal probability in the map. Otherwise, it is constrained by its semantic neighbor to maintain their contact mode. As defined in Table I, according to the combination of surround-view semantic objects with different abstractions, three semantic-contact modes are identified as *adjacent*, *complementary* and *coincident*.

a) Adjacent mode: As seen in Fig. 5 (a), the parking-slot shares a common inner marking-point with its adjacent parking-slot. Hence, the position of one parking-slot, $f(\widetilde{y_t^i})$, induced by its neighbor, is defined as,

$$f(\widetilde{y_t^i}) = \mathbf{L}_{\tilde{y}_t^i} + \frac{1}{2}(W_{y_t^i} + W_{\tilde{y}_t^i})\mathbf{s}_t^i, \tag{12}$$

where $W_{y_t^i}$ and $W_{\tilde{y}_t^i}$ are the widths of two adjacent parking-slots. $\mathbf{s}_t^i$ in Eq. 12 is a unit vector pointing to a parking-slot observation $\mathbf{O}_{y_t^i}$ from its adjacent observation $\mathbf{O}_{\tilde{y}_t^i}$, which is defined as,

$$\mathbf{s}_t^i // \mathbf{O}_{\tilde{y}_t^i}\mathbf{O}_{y_t^i}. \tag{13}$$

Intuitively, for a parking-slot, its position is constrained by its adjacent parking-slot. Such a prior constraint implies iteratively tweaking each parking-slot to closely contact its adjacent neighbor.

b) Coincident mode: Similarly in Fig. 5 (b), for a parking-slot ID, it coincides with the parking-slot it represents. Hence, the position of one parking-slot ID, $f(\widetilde{y_t^i})$, induced by the parking-slot it represents, is defined as,

$$f(\widetilde{y_t^i}) = \mathbf{L}_{\tilde{y}_t^i}. \tag{14}$$

Intuitively, for a parking-slot ID, it shares the same position as the parking-slot it represents.

c) Complementary mode: As seen in Fig. 5 (c), a speed bump is represented by its two endpoints. Similarly, $f(\widetilde{y_t^i})$ is defined as,

$$f(\widetilde{y_t^i}) = \mathbf{L}_{\tilde{y}_t^i} + D_{y_t^i, \tilde{y}_t^i}\mathbf{s}_t^i, \tag{15}$$

where $D_{y_t^i, \tilde{y}_t^i}$ is the distance between two endpoints of a speed bump. Intuitively, for a speed bump endpoint, its position is constrained by the other speed bump endpoint that belongs to the same speed bump. By imposing such a prior constraint, we can ensure that the distance between the two speed bump endpoints is equal to the length of the speed bump.

d) A unified form of the contact error terms: In order for a unified form of the contact error terms for surround-view landmarks modeling, all semantic landmarks should be abstracted into features with the same geometric properties. As seen in Fig. 6, by defining the width of each surround-view landmark abstracted as a semantic point in *complementary* and *coincident* modes, a unified form of the contact error terms for all surround-view landmarks is presented, i.e.,

$$\mathbf{e}_{con}^{i,t} = \begin{cases} \mathbf{0} & \widetilde{y_t^i} \notin \mathcal{M}_t \\ \frac{1}{2}(W_{y_t^i} + W_{\tilde{y}_t^i})\mathbf{s}_t^i - (\mathbf{L}_{y_t^i} - \mathbf{L}_{\tilde{y}_t^i}) & \widetilde{y_t^i} \in \mathcal{M}_t. \end{cases} \tag{16}$$

Concretely, the width of each parking-slot ID is defined as the opposite of the width of the parking-slot it represents. Since each speed bump consists of two speed bump endpoints, the width of each endpoint is defined as the length of the corresponding speed bump. Intuitively, minimizing the contact error term implies iteratively tweaking each landmark to closely contact its contact neighbor, ensuring that the distance between two contacted surround-view landmarks is equal to the half of the sum of their widths.

3) Registration Error Term: Considering all camera poses and surround-view landmarks, the observation term $p(\mathcal{O}|\mathcal{T}, \mathcal{L})$ is defined as,

$$p(\mathcal{O}|\mathcal{T}, \mathcal{L}) = \prod_{t=1}^{T}\prod_{k=1}^{K_t} p(\mathbf{O}_t^k|\mathbf{T}_t, \mathbf{L}_{y_t^k}), \tag{17}$$

where $\mathbf{T}_t$ is the camera pose at time t and $\mathbf{O}_t^k$ represents the k-th observation at time t. $p(\mathbf{O}_t^k|\mathbf{T}_t, \mathbf{L}_{y_t^k})$ is the observation probability of the k-th observation at time t. Since each surround-view landmark is associated with multiple observations, the registration error term of the k-th landmark observed at time t can be defined as,

$$\mathbf{e}_{reg}^{k,t} = \mathbf{T}_t\mathbf{L}_{y_t^k} - \mathbf{O}_t^k. \tag{18}$$

4) Surround-View Error Term: Combining both the contact term and the registration term, the surround-view error term $\mathbf{E}_S$ can be constructed by adding up all semantic features, i.e.,

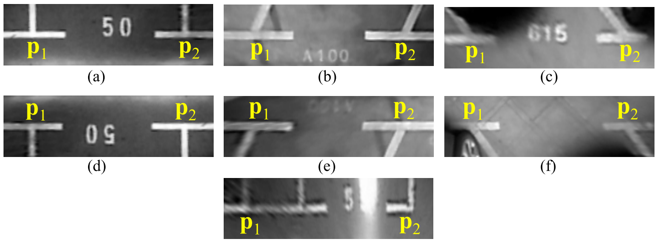
$$\begin{aligned} \mathbf{E}_S &= \mathbf{E}_{Con} + \mathbf{E}_{Reg} \\ &= \sum_{t=1}^{T}\sum_{i=1}^{N_t}(\mathbf{e}_{con}^{i,t})^T\Lambda_{i,t}\mathbf{e}_{con}^{i,t} + \sum_{t=1}^{T}\sum_{k=1}^{K_t}(\mathbf{e}_{reg}^{k,t})^T\Phi_{k,t}\mathbf{e}_{reg}^{k,t}, \end{aligned} \tag{19}$$

where both $\Lambda_{i,t}$ and $\Phi_{k,t}$ are in proportion to the detection confidence of each semantic feature. By minimizing Eq. (19), intuitively, the objective of our proposed surround-view error term encourages both geometric and observational consistency.

IV. SYSTEM IMPLEMENTATION

A. Semantic Object Detection

We adopt a two-stage CNN for surround-view semantic object detection as seen in Fig. 8 (a). The network consists of a semantic point detection module and a semantic pattern classification module. First, each semantic point candidate (the marking-point of a parking-slot, a parking-slot ID or a speed bump endpoint) is located, whose position is defined as the center of the bounding box detected. Then, if a semantic



(a) (b) (c) (d) (e) (f) (g)

Fig. 7. Representative local image patterns of parking-slots. (a)-(g) are representative local image patterns belonging to the classes "right-angled anticlockwise", "slanted anticlockwise with an acute parking angle", "slanted anticlockwise with an obtuse parking angle", "right-angled clockwise", "slanted clockwise with an obtuse parking angle", "slanted clockwise with an acute parking angle" and "invalid", respectively.

object is represented by two points with a specific pattern, the above semantic point candidates will be combined in pairs. Afterwards, the qualified pairs are selected by a classification network. Compared with the single semantic point based representation, a semantic object represented by a pair of semantic points with a specific pattern can be endowed with certain geometric information, such as the width of a parking-slot. Meanwhile, a digit classification network is used to identify the parking-slot ID for a high-level understanding of the surrounding environment. The reason why we proposed a two-stage object detection framework by first detecting semantic points is that the autonomous parking task has strict requirements for the localization accuracy of the marking-points of a parking-slot or the endpoints of a speed bump. If just one bounding box is used to detect these objects, it is difficult to have an accurate output of the positions of these marking-points or endpoints. Furthermore, as illustrated in Fig. 7, for parking-slot detection, the aim of the pair classification module is not just to tell if two landmarks belong to the same parking-slot. Some necessary information like the type (right-angled or slanted) and the orientation (whether the associated parking-slot is on the clockwise side or on the anticlockwise side of $\overrightarrow{\mathbf{p}_1\mathbf{p}_2}$) of each parking-slot are also acquired in order to infer the coordinates of its other two vertices.

B. Semantic Data Association

Usually, when constructing the registration term, two elements are involved, a semantic landmark and its corresponding observation data. But in reality, data association between a semantic landmark and its observation is unknown. For example, a parking-slot detected in the surround-view image can't be told which parking-slot landmark in the map it corresponds to. Therefore, we need to perform semantic data association to associate each observation with a semantic landmark. Since the appearances of semantic objects on the ground are either blurred or occluded by a parked car, it is difficult to distinguish them by comparing their appearances. Therefore, the semantic data association in MOFIS$_{\text{SLAM}}$ is mainly based on geometric matching (seen in Fig. 8 (b)).

The probability distribution $p_{\mathbf{O}_t^i}$ of the i-th semantic observation's position $\mathbf{O}_t^i$ detected at time t follows a

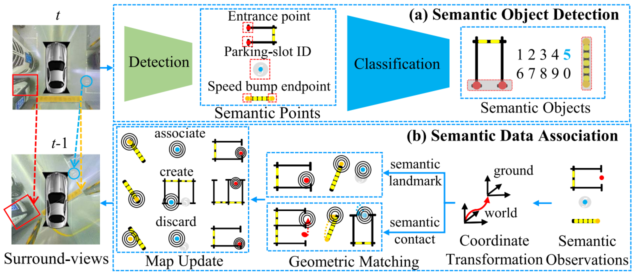


Fig. 8. Semantic object detection and data association. (a) A two-stage CNN network is adopted to detect all semantic objects in the surround-view image. The network consists of a detection module for semantic point detection and a classification module for pattern classification. (b) Semantic data association is based on geometric matching, in which both the semantic landmark and its semantic-contact neighbor are incorporated. According to the distances between one semantic observation and the semantic landmarks in the map, the semantic observation will be 1) associated with one in the map, 2) discarded as an abnormal observation, or 3) regarded as a new one.

Gaussian distribution,

$$p_{\mathbf{O}_t^i} = \mathcal{N}(\mathbf{T}_{CW}^{-1}\mathbf{O}_t^i, \Sigma_{i,t}), \tag{20}$$

where $\mathbf{T}_{CW}$ is the camera pose returned by the visual odometry and $\Sigma_{i,t}$ is the covariance matrix.

Since the position of each surround-view landmark is constrained by its semantic-contact neighbor, the probability of the observation associated with the k-th surround-view landmark in the map can be defined by incorporating both the landmark and its semantic-contact neighbor, i.e.,

$$f_t^i(k) = \alpha \; p_{\mathbf{O}_t^i}(\mathbf{L}_k) + (1-\alpha) \; p_{\mathbf{O}_t^{\tilde{i}}}(\mathbf{L}_{\tilde{k}}), \tag{21}$$

where $\mathbf{O}_t^{\tilde{i}}$ and $\mathbf{L}_{\tilde{k}}$ denote the positions of the semantic-contact neighbor of the i-th surround-view observation at time t and the k-th semantic landmark, respectively. For each type of semantic objects, α is set to be larger than 0.5 if $\mathbf{L}_k$'s localization accuracy is larger than $\mathbf{L}_{\tilde{k}}$'s localization accuracy. Then we perform semantic data association in a strict manner,

$$y_t^i = \begin{cases} k & f_t^i(k) \le th1(c_k) \\ \varnothing & th1(c_k) < f_t^i(k) < th2(c_k) \\ N_t + 1 & f_t^i(k) \ge th2(c_k), \end{cases} \tag{22}$$

where $th1(c_k)$ and $th2(c_k)$ are association and creation thresholds, which are self-adapted according to the statistics of each type of semantic objects. Specifically, when $f_t^i(k)$ is within the association threshold th_1, the observed semantic feature is associated with the k-th semantic landmark in the map. When it is larger than the predefined creation threshold th_2, it means there is no associated semantic landmark in the map, and a new semantic landmark with ID $n_t + 1$ is created in the map. Otherwise, the semantic observation will be discarded.

Semantic Update: Once a semantic observation is associated with an existing surround-view landmark in the map, its position $\mathbf{L}_{y_t^k}$ is first updated by the filtering method to provide an initial value for the further optimization, i.e.,

$$\mathbf{L}_{y_t^k} = (\sum_{n=1}^{n_t} \lambda^{n_t - n + 1}\mathbf{T}_{CW}\mathbf{O}_t^k +_{\omega} \mathbf{P}_{y_t^k})/(n_t + 1), \tag{23}$$

where n_t is the total number of semantic landmarks at time t. $\lambda = 0.9$ is the decay parameter, which implies that the

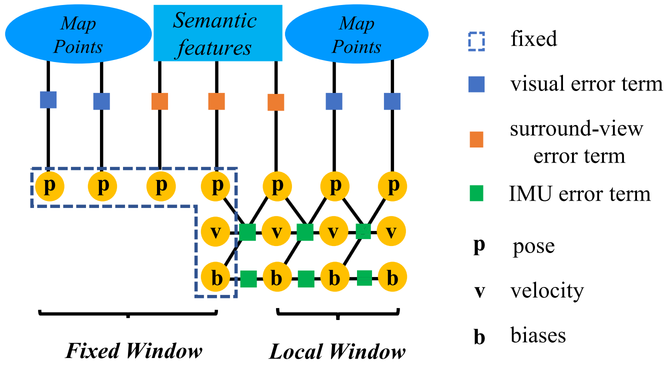


Fig. 9. Window optimization. MOFIS$_{\mathrm{SLAM}}$ is optimized by minimizing a combination of an IMU error term, a visual error term and a surround-view error term. In order to ensure the observability of the optimization, we optimize key frames in the sliding window.

measurement at time t is more reliable than that at time $t-1$. The reason that we choose to trust more on the current measurement is that the current camera pose is more reliable than previous camera poses with the optimization framework MOFIS$_{\mathrm{SLAM}}$.

C. Window Optimization

MOFIS$_{\mathrm{SLAM}}$ is optimized by minimizing a combination of an IMU error term, a visual error term and a surround-view error term. The visual error term links map points and camera poses, whereas the IMU error term links motion data (pose, velocity and biases) between consecutive keyframes. Additionally, the surround-view error term optimizes each surround-view semantic feature and the camera pose at which the semantic landmark is observed. In order to make a good trade-off between the speed and the flexibility, the optimization is performed within a sliding window. Frames with sufficient features and large parallax are selected as keyframes and are inserted into the sliding window. Since there are additional states of semantic features in the surround-views, frames that don't hold enough features will, nevertheless, be regarded as a new keyframe if semantic features are detected in the corresponding surround-view. When a new keyframe is inserted into the sliding window, it optimizes the last N keyframes in the local window and all points seen by those N keyframes. Semantic features are also incorporated during optimization. All other keyframes that share observations of map points and semantic features contribute to the total cost but are fixed in a fixed window during optimization in order to provide a deterministic solution. Note that the keyframe $N+1$ is always included in the fixed window as it constrains the IMU states. A suitable local window size is chosen for real-time performance. If the total number of keyframes exceeds the local window size, redundant keyframes are discarded. This policy improves tracking robustness and enhances lifelong operation.

V. Suite Calibration

All sensors in MOFIS$_{\mathrm{SLAM}}$ are required to be spatially and temporally registered for the best performance in sensor fusion.

TABLE II

System Coordinate Denotation

Sensor	Denotation
IMU	B
front camera	F
front camera in the surround-view system	C_1
left camera in the surround-view system	C_2
back camera in the surround-view system	C_3
right camera in the surround-view system	C_4

A. Sensor Calibration

The spatial registration of different sensors consists of intrinsic calibration and extrinsic calibration. The intrinsic calibration of all sensors can be achieved in advance by offline calibration. Specifically, camera intrinsic parameters, the focal length, the optical center and distortion parameters, can be obtained by [27], [28], and with respect to the IMU, its intrinsic parameters are determined by the Allan Variance [29].

According to different types of sensors, the extrinsic calibration can be categorized into three respects: camera-IMU calibration, camera-ground calibration and surround-view camera system calibration. The coordinate system corresponding to each sensor is denoted as shown in Table II.

1) Camera-IMU Calibration: For camera-IMU calibration, the front-view camera and IMU are considered rigidly attached and the transformation between their coordinate systems can be denoted by $\mathbf{T}_{FB}$. Specifically, we collect a set of data typically over several minutes as the camera-IMU is waved in front of a static calibration pattern. Following [30], $\mathbf{T}_{FB}$ can be then computed by optimizing the error term between IMU and camera measurement.

2) Camera-Ground Calibration: By selecting four points $\mathcal{P}_G = \{\mathbf{P}_G^i\}_{i=1}^4$ on a calibration site on the ground, the transformation matrix $\mathbf{T}_{FG}$ from the ground coordinate system to the front-view camera coordinate system can be estimated by solving a PnP problem formed by points in $\mathcal{P}_G$ and their corresponding image pixels [31], [32].

3) Surround-View Camera System Calibration: Given a surround-view system consisting of four fisheye cameras $\{C_i\}_{i=1}^4$ and the ground coordinate system O_G, the poses of cameras in O_G are denoted by $\{\mathbf{T}_{C_iG}\}_{i=1}^4$, which can be calibrated offline. For a point $\mathbf{P}_G = [X_G, Y_G, Z_G, 1]^T$ in O_G, its corresponding pixel coordinate $\mathbf{p}_{C_i}$ in the camera coordinate system of C_i is given by,

$$\mathbf{p}_{C_i} = \frac{1}{Z_{C_i}}\mathbf{K}_{C_i}\mathbf{T}_{C_iG}\mathbf{P}_G, \quad i = 1, 2, 3, 4, \tag{24}$$

where Z_{C_i} is the depth of $\mathbf{P}_G$ in camera C_i's coordinate system, and $\mathbf{K}_{C_i}$ is the 3×3 intrinsic matrix of camera C_i.

The bird's-eye-view image can be generated by projecting an image to the ground, namely the plane $Z_G = 0$ in O_G. Consider a point $\mathbf{p}_G = [u_G, v_G, 1]^T$ in the bird's-eye-view image, where u_G and v_G are the coordinate values of $\mathbf{p}_G$ in the bird's-eye-view coordinate system, respectively. Its corresponding point on the ground plane is $\mathbf{P}_G = [X_G, Y_G, Z_G = 0]^T$ with respect to the ground coordinate system, where X_G, Y_G, Z_G are the coordinate values of $\mathbf{P}_G$, respectively. The relationship between $\mathbf{p}_G$ and $\mathbf{P}_G$ can be represented as,

$$\begin{bmatrix} u_G \\ v_G \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{d_{X_G}} & 0 & \frac{W}{2d_{X_G}} \\ 0 & -\frac{1}{d_{Y_G}} & \frac{H}{2d_{Y_G}} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_G \\ Y_G \\ 1 \end{bmatrix}, \tag{25}$$

where d_{X_G} and d_{Y_G} are the size of each pixel, W and H are the width and height of the scope covered by the surround-view image. It is worth mentioning that because $Z_G = 0$, Z_G is ignored implicitly. Denote the transformation matrix from $\mathbf{P}_G$ to $\mathbf{p}_G$ by $\mathbf{K}_G$, and then Eq. (25) can be simplified as,

$$\mathbf{p}_G = \mathbf{K}_G \mathbf{P}_G. \tag{26}$$

By combining Eq. (24) and Eq. (26), we can get,

$$\mathbf{p}_{C_i} = \frac{1}{Z_{C_i}} \mathbf{K}_{C_i} \mathbf{T}_{C_i G} \mathbf{K}_G^{-1} \mathbf{p}_G. \tag{27}$$

Eq. (27) actually depicts the relationship of a point $\mathbf{p}_{C_i}$ on the image plane of camera C_i and its projection $\mathbf{p}_G$ in the surround-view image.

B. Sensor Synchronization

Apart form spatial calibration, all sensors should also be temporally calibrated in advance. Commonly, the approaches to sensor synchronization can be of two modes, the “hardware” mode and the “software” mode according to whether customized hardware is required or not. With the “hardware” mode, additional hardware is used to synchronize multiple devices, whereas with the “software” mode, sensors are synchronized merely by the programming control. Thus, the “software” synchronization is relatively easy to achieve compared with the “hardware”-based one which should rely on additional hardware. However, different time delays of different sensors in pre-processing, buffering or data transmission compromise the performance of “software” synchronization in applications with high requirements for data acquisition frequency. By contrast, the “hardware” synchronization performs better at the cost of expensive customized hardware. At present, the sensor synchronization strategy we chose is a combination of the “hardware” synchronization mode and the “software” synchronization mode as shown in Fig. 10. For the synchronization of the IMU and the front-view camera, the “hardware” mode is chosen by using the clock pulse of the IMU to simultaneously trigger camera exposure. For fisheye cameras in the surround-view system and the front-view camera, they are “software” synchronised by capturing images controlled by a multi-thread data collection function. The “software” synchronization mode is valid for an autonomous indoor parking system due to the fact that the vehicle typically runs at a moderate speed when it drives in an indoor parking environment.

VI. Benchmark Dataset Establishment

To facilitate SLAM studies for autonomous indoor parking, we have established a large-scale dataset comprising synchronous multi-sensor data when driving in a typical indoor parking site. Apart from synchronous multi-sensor data, the groundtruth trajectories are also provided in the dataset. In this section, we will present overview of the dataset and the way in which the groundtruth trajectories are acquired.

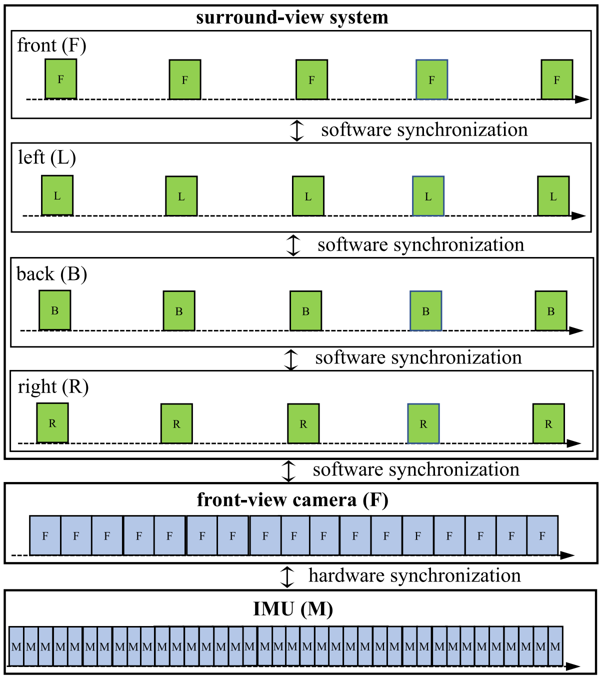


Fig. 10. Sensor synchronization. For the synchronization of the IMU and the front-view camera, the “hardware” mode is chosen by using the clock pulse of the IMU to simultaneously trigger camera exposure. For the surround-view system and the front-view camera, they are “software” synchronised by capturing images controlled by a multi-thread data collection function.

A. Dataset Overview

Totally, our dataset contains 12,407 front-view images, 12,407 IMU motion data segments with each segment recording the vehicle motion between two consecutive front-view frames, and 4,033 surround-view images (synthesized from 16,132 fisheye images). The collection frequencies of the front-view camera, the IMU and the surround-view image are 20*Hz*, 200*Hz* and 10*Hz*, respectively. The resolutions of the fisheye camera and the front-view camera are 1280 × 1080 and 1280 × 720, respectively. The spatial resolution of each surround-view image is 416 × 416, corresponding to a 10*m*×10*m* flat physical region, i.e., the length of 1 pixel in the surround-view image corresponds to 2.40*cm* on the physical ground. Actually how to achieve a balance between the view range of a surround-view image and its accuracy is a common practical engineering problem. It actually depends on which factor the end user attaches more importance to. The wider the size of the surround-view image, the larger the localization error of objects detected in regions closer to its boundary. In our system, when a 416×416 surround-view image covers a ground area of 10*m*×10*m*, the localization error can be as low as 0.05*m*, which can meet the needs of autonomous parking tasks. One advantage of our dataset is its diversity of the conditions for data collection, ranging from static scenes with bright illumination to dynamic scenes with poor illumination.

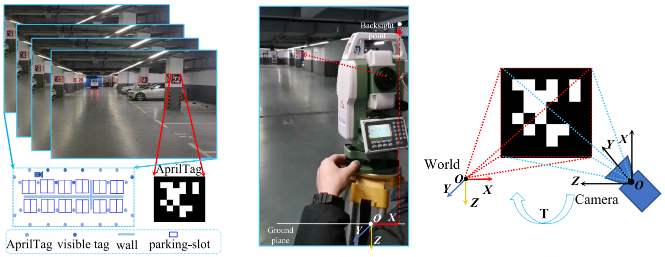


Fig. 11. Groundtruth trajectory acquisition. There are three steps involved, (a) landmarks deployment, (b) artificial landmarks measurement, and (c) camera pose acquisition.

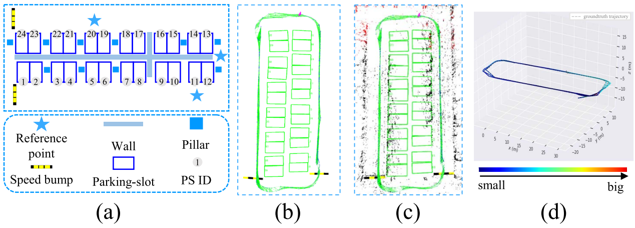


Fig. 12. Qualitative results of MOFIS$_{\text{SLAM}}$. (a) A sketch of an indoor parking site. (b) Mapping result by MOFIS$_{\text{SLAM}}$ without a surround-view error term. (c) Mapping result by MOFIS$_{\text{SLAM}}$ with a surround-view error term. (d) Difference between the estimated and the groundtruth trajectories.

Various dynamic objects, such as moving cars and pedestrians were captured in the dataset. Moreover, the groundtruth trajectories were also acquired by tracking artificial features evenly scattered in the indoor parking environment.

B. Groundtruth Trajectory Acquisition

Actually, when establishing such a dataset, the groundtruth trajectories are crucial for objective evaluation of different SLAM systems. But they are generally unavailable due to the fact that the current groundtruth trajectory acquisition approaches are unsuitable in GPS-denied indoor parking environments or fail to guarantee the integrity of the trajectory due to the high cost of a motion capture system. To address the problem, we provided an effective yet cost-efficient groundtruth trajectory acquisition approach simply with a mild intervention of the environment. In our approach, the groundtruth trajectories were obtained with an ETS. As can be seen in Fig. 11, three steps were involved, landmarks deployment, artificial landmarks measurement and camera pose acquisition.

1) Landmarks Deployment: Landmarks deployment ensures a tailored indoor parking environment with artificial landmarks that can be easily detected (Fig. 11 (a)). Specifically, by evenly placing visual fiducial markers such as the popular printable AprilTags [33]–[35] in an indoor parking environment, one can create a set of artificial landmarks scattered throughout the environment.

2) Artificial Landmarks Measurement: Accurate coordinates of artificial landmarks are prerequisite for a high precision of motion tracking. In this step, the coordinates of above artificial landmarks were measured with the benefit of an ETS (Fig. 11 (b)), a compact and portable equipment commonly used in the surveying field. With an ETS, an operator can take measurement of the coordinates of all visible points with accuracy within a couple of millimeters. Specifically, a survey point **O** on the ground was first selected where points in all directions can be observed as much as possible. And the survey point was defined as the origin of the coordinate system. By placing the ETS over the survey point and designating a back-sight point **X**, **X**-axis was then built pointing from the origin point **O** to the projection of the selected point **X** on the ground. Then **Y**-axis and **Z**-axis were established based on the orthogonal principle. By emitting laser to the target point, its coordinate was obtained based on reflection duration. By moving the ETS around, the coordinates of all artificial landmarks throughout the environment were precisely obtained by the resection method [36].

3) Camera Pose Acquisition: When driving the vehicle in an indoor parking environment, its mounted front-view camera could detect these artificial landmarks during navigation. By aligning artificial landmarks $\mathbf{P}_W$ with known 3D coordinates and their 2D projections into the front camera with known pixel coordinates $\mathbf{p}_C$, the acquisition of camera pose $\mathbf{T}$ was casted as solving a PnP problem (Fig. 11 (c)), i.e.,

$$\mathbf{T} = \arg\min_{\mathbf{T}} \sum_{i=1}^{N} e_i = \arg\min_{\mathbf{T}} \sum_{i=1}^{N} ||f(\mathbf{T}, \mathbf{P}_W^i, \mathbf{D}) - \mathbf{p}_C^i||_2^2, \tag{28}$$

where $\mathbf{D}$ is the set comprising the distortion coefficients of the camera, and $f(\ldots, \mathbf{D})$ is the camera distortion model that transforms each point $\mathbf{P}_W^i$ in the world coordinate system to the point on the camera's imaging plane. EPnP algorithm [31], [32] was adopted to solve the problem and several variants include DLT [37], P3P [38] and UPnP [39] can also be used to solve this problem. The optimal camera pose $\mathbf{T}$ was acquired in an iterative manner for robustness and accuracy. Specifically, $\mathbf{T}$ was initially obtained using the RANSAC method and points with large reprojection errors were removed. Afterwards, $\mathbf{T}$ was further refined using the remaining points until the number of the remaining points was stable.

VII. Experimental Results

A. Qualitative Results of MOFIS$_{SLAM}$

To qualitatively validate the effectiveness of the proposed MOFIS$_{\text{SLAM}}$, we compared semantic maps obtained using different optimization strategies on the collected dataset. Fig. 12 (a) depicts the sketch of an indoor parking site from a top-down viewpoint. The parking site consists of two rows of parking-slots. Each row is composed of 12 parking-slots and two consecutive parking-slots are adjacent with each other. Each parking-slot is represented by a unique parking-slot ID. Additionally, there are two speed bumps in this indoor parking site. Fig. 12 (b) and Fig. 12 (c) demonstrate the results when the vehicle was equipped with a surround-view camera system, both of which not only construct 3D landmarks but also detect parking-slots in surround-view images (geometric points are omitted in Fig. 12 (b) for the comparison). Fig. 12 (b) shows

TABLE III
QUALITATIVE COMPARISON WITH OTHER METHODS

Methods \ Properties	Sensor Modality	Map Category	MSO
Bowman *et al.* [1]	V (Visual)	Semantic	×
Civera *et al.* [2]	V	Semantic	×
Tateno *et al.* [3]	V	Semantic	×
Yang *et al.* [4]	V	Semantic	×
Yu *et al.* [5]	V	Semantic	×
Mur-Artal *et al.* [6]	V + I	Geometric	×
Campos *et al.* [7]	V + I	Geometric	×
Shao *et al.* [9]	V + I + S	Semantic	×
Schreiber *et al.* [13]	V + I	Semantic	×
Ranganathan *et al.* [14]	V + I	Semantic	×
Jeong *et al.* [15]	V + I	Semantic	×
Zhao *et al.* [18]	V + I + T	Semantic	×
MOFIS$_{\text{SLAM}}$	**V + I + S**	**Semantic**	√

the result without incorporating the surround-view error term during optimization. In Fig. 12 (b), it can be seen that even though all surround-view landmarks are incorporated in the map, the left speed bump is not spatially aligned with the parking-slot; besides, since the scale estimated by IMU is difficult to be absolutely accurate, there is an abnormally large distance between two groups of adjacent parking-slots below. Fig. 12 (c) shows the result with the surround-view error term during optimization. When a surround-view error term is taken into consideration in optimization, the overall scale gets reasonable. Consequently, in Fig. 12 (c), the distance between two groups of adjacent parking-slots below is more in line with the spatial distribution of the real scene; moreover, the left speed bump is now spatially aligned with the parking-slot. It implies that when the surround-view error term is incorporated during optimization, MOFIS$_{\text{SLAM}}$ not only facilitates the vehicle to understand the indoor parking environment by demonstrating all important surround-view landmarks, but also minimizes accumulated errors to provide a semantic map with higher accuracy. Additionally, the difference between the estimated trajectory by our MOFIS$_{\text{SLAM}}$ and the groundtruth trajectory is illustrated in Fig. 12 (d). It can be seen from Fig. 12 (d) that the estimated and the groundtruth trajectories are roughly coincident, demonstrating the high accuracy of the localization result of MOFIS$_{\text{SLAM}}$. A demo video of MOFIS$_{\text{SLAM}}$ is available online at https://shaoxuan92.github.io/MOFIS.

B. Qualitative Comparison With Other SLAM Systems

Table III shows the qualitative comparison of MOFIS$_{\text{SLAM}}$ with twelve existing representative SLAM systems from the viewpoints of three aspects, sensor modalities used (‘S’ for surround-view features, ‘I’ for inertial data and ‘T’ for fiducial tags), the category of the map constructed, and whether multiple surround-view objects (MSO) are incorporated in the map. It can be seen from Table III that only Shao *et al.*’s scheme [9] and MOFIS$_{\text{SLAM}}$ incorporate the surround-view as a data source, not only constructing semantic maps with surround-view features in the environment, but leveraging no other information like fiducial tags used in [18] in optimization. Compared with Shao *et al.*’s scheme [9], MOFIS$_{\text{SLAM}}$ is the first VI-SLAM system attempting to make full use of various semantic objects detected in surround-views in order to ensure tracking consistency over the long-term navigation. More importantly, in MOFIS$_{\text{SLAM}}$, a unified form of surround-view constraints is established to render the system highly adaptive to various indoor parking scenarios. Therefore, to be fair, Zhao *et al.*’s scheme and Shao *et al.*’s scheme are chosen as the comparison targets. Apart from these two schemes, Campos *et al.*’s scheme [7] is also included. Fig. 13 (a) illustrates the result of Campos *et al.*’s scheme. It records the driving path in the indoor parking site. However, semantic objects on the ground that are essential for autonomous indoor parking are not incorporated in the map. Fig. 13 (b) and Fig. 13 (c) demonstrate the results of Zhao *et al.*’s scheme and Shao *et al.*’s scheme, both of which not only construct 3D landmarks but also detect parking-slots in surround-view images. It can be seen from Fig. 13 (b) that not all parking-slots are detected and displayed in the map. Fig. 13 (c) shows that all parking-slots are displayed in the map. However, surround-view features selected in Shao *et al.*’s scheme are parking-slot specific, resulting in tracking inconsistency in circumstances where parking-slots are occluded by a parked car. Fig. 13 (d) shows the mapping result of MOFIS$_{\text{SLAM}}$. It can be seen from this figure that all surround-view landmarks (parking-slot IDs are omitted here for display) are incorporated in the map. Moreover, the positions of the semantic landmarks in Fig. 13 (d) are more in line with their real spatial distributions, implying that MOFIS$_{\text{SLAM}}$ has a better mapping capability than its counterparts. Additionally, in Fig. 13 (e), the map of an outdoor parking site with slanted parking-slots constructed by MOFIS$_{\text{SLAM}}$ is presented, which demonstrates that MOFIS$_{\text{SLAM}}$ is also applicable to slanted parking-slots.

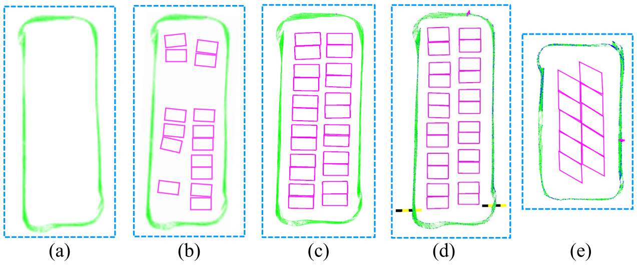


Fig. 13. (a) Mapping result by Campos *et al.*’s scheme [7]. (b) Mapping result by Zhao *et al.*’s scheme [18]. (c) Mapping result by Shao *et al.*’s scheme [9]. (d) Mapping result by MOFIS$_{\text{SLAM}}$. (e) Mapping result by MOFIS$_{\text{SLAM}}$ in an outdoor parking site with slanted parking-slots.

C. Quantitative Evaluation of MOFIS$_{\text{SLAM}}$

Four metrics are selected for quantitative evaluation of MOFIS$_{\text{SLAM}}$’s performance on the collected dataset, the revisiting error (RE), the distance of adjacent parking-slots (DAP), the absolute trajectory error (ATE) and the time cost (TC).

1) Revisiting Error: As seen in Fig. 14 (a), revisiting error measures the difference between the localization results at different times. When it is difficult to obtain the groundtruth of the driving path, the revisiting error is valid for localization evaluation since an autonomous parking system allows for

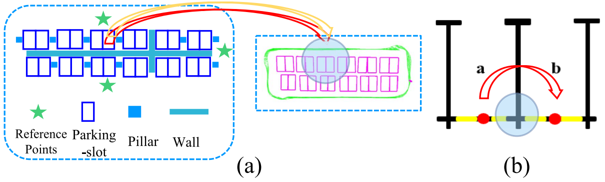


Fig. 14. Evaluation metrics[1]. The performance of a SLAM system can be measured by calculating (a) the revisiting error or (b) the distance of adjacent parking-slots.

TABLE IV
REVISITING ERRORS OF SELECTED TEST POINTS. (UNIT: METER)

Round	X	Y	Z	ΔX	ΔY	ΔZ	ΔD
Point 1 (-7.66 -0.15 5.03)							
Rd. 1	-7.62	-0.15	5.01	-0.04	0	0.02	0.045
Rd. 2	-7.65	-0.15	5.05	0.01	0	-0.02	0.002
Rd. 3	-7.72	-0.16	5.04	0.04	-0.01	-0.01	0.042
Point 2 (-32.53 0.17,2.86)							
Rd. 1	-32.52	0.18	2.91	-0.01	-0.01	-0.05	0.052
Rd. 2	-32.54	0.18	2.84	0.01	-0.01	0.02	0.024
Rd. 3	-32.54	0.18	2.82	0.01	-0.01	0.04	0.042
Point 3 (-20.60 -0.42 12.26)							
Rd. 1	-20.60	-0.42	12.31	0	0	-0.05	0.050
Rd. 2	-20.61	-0.41	12.21	0.01	-0.01	0.05	0.052
Rd. 3	-20.64	-0.40	12.27	0.04	-0.02	-0.01	0.046

an absolute localization error during navigation. As long as the revisiting error is small enough, the vehicle will adopt a consistent driving strategy when it drives to the same position. Hence, the revisiting error is defined as the averaged l_2 distance of the localization results when passing the same reference point twice, i.e.,

$$e_{rev} = \frac{1}{RP}\sum_{r=1}^{R}\sum_{p=1}^{P}||\mathbf{L}_r^p - \mathbf{L}_{r+1}^p||_2, \qquad (29)$$

where $\mathbf{L}_r^p$ denotes the position of the reference point p at the r-th round and $\mathbf{L}_{r+1}^p$ is p's position at the $(r+1)$-th round.

In actual operation, the driver first manually drove the vehicle at around 10*km/h* and the map was then initialized. Three map points at different locations (Points 1, 2, and 3) were selected as reference points for the test (Refer to Fig. 12(a)). Specifically, we chose two at the midpoints of both sides of the indoor parking site and one at the corner. Revisiting errors on these points are presented in Table IV. It can be seen from Table IV that for MOFIS$_{\text{SLAM}}$ the revisiting error at each test point is less than 0.06*m*.

2) Distance of Adjacent Parking-Slots: Apart from revisiting error, distances of adjacent parking-slots are selected to evaluate the mapping accuracy of SLAM system. Since the adjacent parking-slots share a common marking-point, the distance of two adjacent parking-slots is theoretically zero. So we can measure the performance of MOFIS$_{\text{SLAM}}$ by calculating the averaged distance of all k groups of adjacent parking-slots. The formula is as follows,

$$e_{adj} = \frac{1}{K}\sum_{k=1}^{K}||\mathbf{L}_k^2 - \mathbf{L}_{\tilde{k}}^1)||_2, \qquad (30)$$

where $\tilde{k}$ is the adjacent parking slot of the k-th parking-slot in the map. $\mathbf{L}_k^2$ and $\mathbf{L}_{\tilde{k}}^1$ are the positions of the common marking-points of the k-th parking-slot and its adjacent counterpart. We can see from Table V that the averaged distance of adjacent parking-slots undergoes a dramatic decrease by 0.087*m*, a 58% decrease, if surround-view error terms are incorporated in optimization, corroborating the benefits brought by the surround-view semantic constraints in MOFIS$_{\text{SLAM}}$.

3) Absolute Trajectory Error: Although both the revisiting error and the distance of adjacent parking-slots are valid, they have certain drawbacks. The former involves manual intervention like parking the vehicle at a designated spot, which is troublesome to achieve in real circumstances. And the latter evaluates the adjacency property of two parking-slots, which is actually integrated as a surround-view constraint during optimization. Fortunately, with the groundtruth trajectory acquired in Sect. VI-B, the absolute trajectory error can be used to evaluate the SLAM system's performance directly by measuring the difference between the estimated and the groundtruth trajectories, i.e.,

$$e_{ATE} = \Big(\frac{1}{M}\sum_{i=1}^{M}||trans(\mathbf{Q}_i^{-1}\mathbf{P}_i)||^2\Big)^{\frac{1}{2}}, \mathbf{Q}_i, \mathbf{P}_i \in SE(3), \quad (31)$$

where $\{\mathbf{Q}_i\}_{i=1}^M$ and $\{\mathbf{P}_i\}_{i=1}^M$ are groundtruth and estimated poses of all frames, respectively. The *trans*(.) represents the translation part of the pose. M is the total number of frames. It can be found from Table VI that the groundtruth trajectory error is 0.27*m* on overage, a 22% decrease compared with the navigation without a surround-view error term.

4) Time Cost: We recorded the average processing time per frame of MOFIS$_{\text{SLAM}}$ using different number of features. The result is presented in Fig. 15. It can be seen that when 1000 features are used, the average processing time per frame within 500 frames is 0.054*s*. When the vehicle trajectory loops at around 3000 frames, the average processing time per frame is 0.063*s*, reaching 15*fps*, which is qualified when driving in an indoor parking site at a low speed. In fact, the frame rate of the system can be improved by reducing the number of extracted feature points. When the number of feature points is set as 500, the running speed undergoes a considerable improvement. Therefore, if there is a requirement for a higher frame rate, we can reduce the number of extracted feature points. Moreover, we also evaluated the performance of MOFIS$_{\text{SLAM}}$ with the introduction of feature points from surrounding images. We find that when additional 500 features are introduced at each sampling point in MOFIS$_{\text{SLAM}}$, the ATE is reduced slightly from 0.272*m* to 0.268*m*. But the frame rate decreases from 15*fps* to 7*fps*. According to our experience, such a frame rate cannot meet the needs of autonomous parking tasks.

D. Quantitative Comparison With Other SLAM Systems

To be fair, in this quantitative comparison experiment, Campos *et al.*'s scheme [7], Zhao *et al.*'s scheme [18] and Shao *et al.*'s scheme [9] are chosen as the comparison targets. The performance of the competitors was evaluated in terms

[1] The right-side map shown in (a) is built by imposing the additional constraints that the parking-slot lines are parallel or vertical to each other.

TABLE V
DISTANCES OF ALL GROUPS OF ADJACENT PARKING-SLOTS W/O SURROUND-VIEW ERROR TERMS. (UNIT: METER)

Groups	1	2	3	4	5	6	7	8	9	10	11	12	Mean
Without $\mathbf{E}_S$	0.079	0.095	0.220	0.045	0.110	0.090	0.126	0.097	0.602	0.124	0.114	0.092	**0.150**
With $\mathbf{E}_S$	0.040	0.048	0.056	0.064	0	0.071	0.001	0.041	0.131	0.049	0.151	0.099	**0.063**
Decrease	0.039	0.047	0.164	-0.019	0.110	0.019	0.125	0.056	0.471	0.075	-0.037	-0.007	**0.087**

TABLE VI
THE AVERAGE OF ABSOLUTE TRAJECTORY ERRORS OF TRAJECTORIES IN DIFFERENT DRIVES W/O SURROUND-VIEW ERROR TERMS. (UNIT: METER)

Trajectories	1	2	3	4	5	6	Mean
Without $\mathbf{E}_S$	0.38	0.23	0.19	0.49	0.21	0.49	**0.33**
With $\mathbf{E}_S$	0.33	0.21	0.15	0.26	0.26	0.42	**0.27**
Decrease	0.05	0.02	0.04	0.23	-0.05	0.07	**0.06**

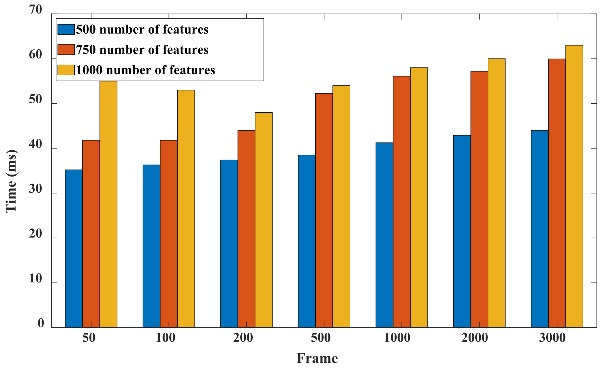


Fig. 15. Average processing time per frame using different number of features.

TABLE VII
QUANTITATIVE COMPARISON WITH OTHER SLAM SYSTEMS

Methods \ Metrics	ATE (*m*)	RE (*m*)	DAP (*m*)
Campos *et al.* [7]	0.297	0.187	-
Zhao *et al.* [18]	-	0.280	-
Shao *et al.* [9]	0.496	0.125	0.106
MOFIS$_{\text{SLAM}}$	**0.272**	**0.068**	**0.063**

of ATE, RE, DAP and TC. From Table VII, we can see that the ATE of MOFIS$_{\text{SLAM}}$ undergoes a dramatic decrease by 0.025*m* and 0.224*m* compared with Campos *et al.*'s scheme [7] and Shao *et al.*'s scheme [9], confirming the superiority of the localization accuracy of MOFIS$_{\text{SLAM}}$. Meanwhile, with respect to RE, MOFIS$_{\text{SLAM}}$ gains 63%, 75% and 46% of the favor compared with Campos *et al*.'s scheme [7], Zhao *et al.*'s scheme [18] and Shao *et al.*'s scheme [9], respectively. Besides, compared with Shao *et al.*'s scheme [9], the DAP of MOFIS$_{\text{SLAM}}$ enjoys a significant decrease. As for TC, Fig. 16 shows the comparison result of MOFIS$_{\text{SLAM}}$ with other three representative methods. It can be seen from Fig. 16 that the average processing time per frame of Zhao *et al.*'s scheme [18] is significantly larger than the others, which cannot reach 10*fps*. As for the other three schemes, their differences on runtime efficiency are negligible.

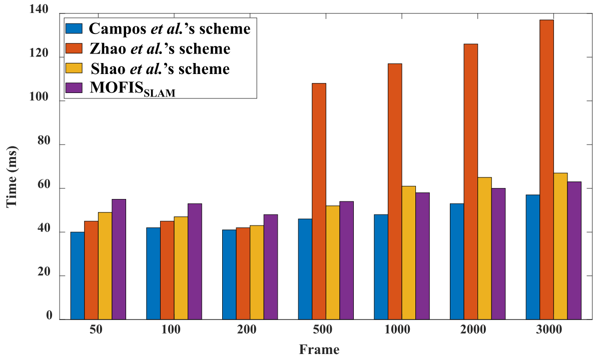


Fig. 16. Average processing time per frame comparison across evaluated schemes.

TABLE VIII
OPTIMIZATION RESULTS USING VARIOUS ERROR TERMS

Strategies \ Metrics	ATE (*m*)	RE (*m*)	DAP (*m*)	TC (*s*)
V-I$_{\text{SLAM}}$	0.319	0.199	-	0.045
VIS-T$_{\text{SLAM}}$	0.332	0.253	0.150	0.051
MOFIS$_{\text{SLAM}}$	0.272	0.068	0.063	0.063

E. Ablation Study

We demonstrate how different error terms in our framework affect the optimization results by comparing MOFIS$_{\text{SLAM}}$ with two baselines using different optimization strategies. The two baselines are 1) V-I$_{\text{SLAM}}$: a visual-inertial error term based system without the incorporation of surround-view semantic features; 2) VIS-T$_{\text{SLAM}}$: a system which incorporates surround-view semantic features in optimization only during the tracking phase. The results are presented in Table VIII. In V-I$_{\text{SLAM}}$, the visual error term links each geometric feature point in the front-view image and its projecting map point while the inertial error term constrains consecutive keyframes by visual-inertial alignment, stably predicting reliable camera poses and map points. It can be seen from Table VIII that V-I$_{\text{SLAM}}$ can reach satisfying performance in terms of three evaluation metrics of ATE, RE and TC, which are 0.319*m*, 0.199*m* and 0.045*s*, respectively. But V-I$_{\text{SLAM}}$ is not suitable for autonomous indoor parking due to the fact that it provides no semantic information during navigation. In order to enhance the system's robustness against varying illumination and low-texture conditions, semantic objects on the ground should be incorporated. As for the performance of VIS-T$_{\text{SLAM}}$, we can find that if we incorporate semantic features extracted from surround-views in optimization only during the tracking phase, the optimization results are compromised and large ATE and RE occur. But if the surround-view semantic features are incorporated in optimization during all the phases of tracking, local mapping and loop closing just as $\text{MOFIS}_{\text{SLAM}}$ does, three evaluation metrics of ATE, RE and DAP can be all considerably diminished, confirming the effectiveness of $\text{MOFIS}_{\text{SLAM}}$. In addition, the average processing time per frame of $\text{MOFIS}_{\text{SLAM}}$ is about $0.063s$ (over $15fps$), which can be acceptable for an autonomous parking system running at a moderate speed.

F. Discussions on the Extension of $\text{MOFIS}_{\text{SLAM}}$

$\text{MOFIS}_{\text{SLAM}}$ could be extended to support multiple front-view cameras through the following steps. Firstly, different types of front-view cameras need to be mounted in the appropriate positions of the vehicle. Then, all these cameras should be calibrated in advance to determine their relative poses. And all camera poses can be consequently transformed into the same coordinate system. When these cameras track the feature points in the parking environment, all front-view visual error terms can be introduced in the tightly-coupled optimization framework $\text{MOFIS}_{\text{SLAM}}$ to obtain the optimal camera poses.

G. About Additional Weak Constraints

In some parking sites, the entrance-lines and separating-lines of the parking-slots are vertical or parallel to each other. By imposing such constraints, the visual effect of the distributions of parking-slots in the built map would look better. But in some large-scale parking sites, not all parking-slots lines are vertical or parallel to each other. Therefore, such constraints are not general and they are currently not considered by $\text{MOFIS}_{\text{SLAM}}$.

VIII. Conclusion

In this paper, for the task of fully autonomous indoor parking, we proposed a tightly-coupled semantic SLAM system, $\text{MOFIS}_{\text{SLAM}}$, that integrates salient semantic features in surround-views, with the configuration of a front-view camera, an IMU and a surround-view camera system composed of four cameras mounted around the vehicle. Specifically, each surround-view feature can impose a surround-view constraint that can be split into a contact error term and a registration error term. Three contact modes, defined as *complementary*, *adjacent* and *coincident*, are identified to guarantee a unified form of the contact error terms for all surround-view features. In order to provide an objective evaluation of SLAM systems for autonomous indoor parking, a large-scale benchmark dataset with groundtruth trajectories consisting of synchronous multi-sensor data was collected, which is the first of its kind. The superiority of $\text{MOFIS}_{\text{SLAM}}$ over its counterparts has been verified by extensive qualitative and quantitative experiments. In addition, the collected benchmark dataset is now publicly released to the community to benefit other researchers in this area. In the future, we will continue enlarging our dataset to make it a better benchmark in this field. Additionally, semantic landmarks will be considered for loop closing, since in this way, the whole system would probably be more robust and stable.

References

[1] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, “Probabilistic data association for semantic SLAM,” in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2017, pp. 1722–1729.

[2] J. Civera, D. Galvezlopez, L. Riazuelo, J. D. Tardos, and J. M. M. Montiel, “Towards semantic SLAM using a monocular camera,” in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 1277–1284.

[3] K. Tateno, F. Tombari, I. Laina, and N. Navab, “CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6565–6574.

[4] S. Yang, Y. Song, M. Kaess, and S. Scherer, “Pop-up SLAM: Semantic monocular plane SLAM for low-texture environments,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 1222–1229.

[5] C. Yu *et al.*, “DS-SLAM: A semantic visual SLAM towards dynamic environments,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1168–1174.

[6] R. Mur-Artal and J. D. Tardós, “Visual-inertial monocular SLAM with map reuse,” *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 796–803, Apr. 2017.

[7] C. Campos, R. Elvira, J. J. Gómez Rodríguez, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM,” 2020, *arXiv:2007.11898*.

[8] B. Bozorgtabar and R. Goecke, “MSMCT: Multi-state multi-camera tracker,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 12, pp. 3361–3376, Dec. 2018.

[9] X. Shao, L. Zhang, T. Zhang, Y. Shen, H. Li, and Y. Zhou, “A tightly-coupled semantic SLAM system with visual, inertial and surround-view sensors for autonomous indoor parking,” in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 2691–2699.

[10] X. Gao, S. Shen, L. Zhu, T. Shi, Z. Wang, and Z. Hu, “Complete scene reconstruction by merging images and laser scans,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3688–3701, Oct. 2020.

[11] G.-T. Michailidis, R. Pajarola, and I. Andreadis, “High performance stereo system for dense 3-D reconstruction,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 6, pp. 929–941, Jun. 2014.

[12] H. Kim, J. Y. Guillemaut, T. Takai, M. Sarim, and A. Hilton, “Outdoor dynamic 3-D scene reconstruction,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 11, pp. 1611–1622, Nov. 2012.

[13] M. Schreiber, C. Knoppel, and U. Franke, “LaneLoc: Lane marking based localization using highly accurate maps,” in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 449–454.

[14] A. Ranganathan, D. Ilstrup, and T. Wu, “Light-weight localization for vehicles using road markings,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 921–927.

[15] J. Jeong, Y. Cho, and A. Kim, “Road-SLAM: Road marking based SLAM with lane-level accuracy,” in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2017, pp. 1473–1736.

[16] S. Zhang, L. Wen, Z. Lei, and S. Z. Li, “RefineDet++: Single-shot refinement neural network for object detection,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 2, pp. 674–687, Feb. 2021.

[17] S. Paisitkriangkrai, C. Shen, and J. Zhang, “Fast pedestrian detection using a cascade of boosted covariance features,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 8, pp. 1140–1151, Aug. 2008.

[18] J. Zhao *et al.*, “Visual semantic landmark-based robust mapping and localization for autonomous indoor parking,” *Sensors*, vol. 19, no. 1, pp. 161–180, 2019.

[19] X. Shao, X. Liu, L. Zhang, S. Zhao, Y. Shen, and Y. Yang, “Revisit surround-view camera system calibration,” in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2019, pp. 1486–1491.

[20] A. R. Gaspar, A. Nunes, A. M. Pinto, and A. Matos, “Urban@CRAS dataset: Benchmarking of visual odometry and SLAM techniques,” *Robot. Auto. Syst.*, vol. 109, pp. 59–67, Nov. 2018.

[21] J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim, “Complex urban dataset with multi-level sensors from highly diverse urban environments,” *Int. J. Robot. Res.*, vol. 38, no. 6, pp. 642–657, May 2019.

[22] E. Spera, A. Furnari, S. Battiato, and G. M. Farinella, “EgoCart: A benchmark dataset for large-scale indoor image-based localization in retail stores,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 4, pp. 1253–1267, Apr. 2021.

[23] M. Burri *et al.*, “The EuRoC micro aerial vehicle datasets,” *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016, doi: 10.1177/0278364915620033.

[24] K. M. Judd and J. D. Gammell, “The Oxford multimotion dataset: Multiple SE(3) motions with ground truth,” *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 800–807, Apr. 2019.

[25] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stückler, and D. Cremers, "The TUM VI benchmark for evaluating visual-inertial odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2018, pp. 1680–1687.

[26] L. Zhang, J. Huang, X. Li, and L. Xiong, "Vision-based parking-slot detection: A DCNN-based approach and a large-scale benchmark dataset," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5350–5364, Nov. 2018.

[27] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[28] J. Li and Z. Liu, "Camera geometric calibration using dynamic single-pixel illumination with deep learning networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2550–2558, Aug. 2020.

[29] O. J. Woodman. (2017). *An Introduction to Inertial Navigation*. [Online]. Available: https://www.cl.cam.ac.U.K./techreports/UCAM-CL-TR-696.pdf

[30] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 1280–1286.

[31] F. Moreno-Noguer, V. Lepetit, and P. Fua, "Accurate non-iterative O(n) solution to the PnP problem," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 2252–2259.

[32] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *Int. J. Comput. Vis.*, vol. 81, no. 2, pp. 155–166, 2009.

[33] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 3400–3407.

[34] A. Richardson, J. Strom, and E. Olson, "AprilCal: Assisted and repeatable camera calibration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 1814–1821.

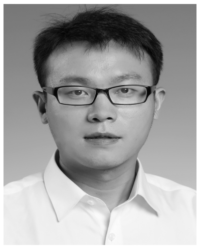
[35] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2016, pp. 4193–4198.

[36] J. L. Awange, E. W. Grafarend, B. Paláncz, and P. Zaletnyik, "Positioning by intersection methods," in *Algebraic Geodesy and Geoinformatics*. Berlin, Germany: Springer, 2010, pp. 249–263.

[37] Y. Abdel-Aziz, H. Karara, and M. Hauck, "Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry," *Photogram. Eng. Remote Sens.*, vol. 81, no. 2, pp. 103–107, 2015.

[38] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 8, pp. 930–943, Aug. 2003.

[39] A. Penate-Sanchez, J. Andrade-Cetto, and F. Moreno-Noguer, "Exhaustive linearization for robust camera pose and focal length estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 10, pp. 2387–2400, Oct. 2013.



Xuan Shao received the B.S. degree from the College of Computer Science and Technology, Qingdao Agricultural University, in 2013, and the M.S. degree from the School of Software Engineering, Tongji University, in 2018, where he is currently pursuing the Ph.D. degree with the School of Software Engineering. His research interests include SLAM systems for autonomous indoor parking and environment perception of intelligent vehicle.

Lin Zhang (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2003 and 2006, respectively, and the Ph.D. degree from the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, in 2011. From March 2011 to August 2011, he was a Research Associate with the Department of Computing, The Hong Kong Polytechnic University. In August 2011, he joined the School of Software Engineering, Tongji University, Shanghai, where he is currently a Full Professor. His current research interests include environment perception of intelligent vehicle, pattern recognition, computer vision, and perceptual image/video quality assessment. He serves as an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS and *Journal of Visual Communication and Image Representation*. He was awarded as a Young Scholar of Changjiang Scholars Program, Ministry of Education, China.

Tianjun Zhang received the B.S. degree from the School of Software Engineering, Tongji University, Shanghai, China, in 2019, where he is currently pursuing the Ph.D. degree. His research interests consist of SLAM systems, computer vision, and machine learning.

Ying Shen (Member, IEEE) received the B.S. and M.S. degrees from Software School, Shanghai Jiao Tong University, Shanghai, China, in 2006 and 2009, respectively, and the Ph.D. degree from the Department of Computer Science, City University of Hong Kong, Hong Kong, in 2012. In 2013, she joined the School of Software Engineering, Tongji University, Shanghai, where she is currently an Associate Professor. Her research interests include bioinformatics and pattern recognition.

Yicong Zhou (Senior Member, IEEE) received the B.S. degree in electrical engineering from Hunan University, Changsha, China, and the M.S. and Ph.D. degrees in electrical engineering from Tufts University, Medford, MA, USA. He is currently a Full Professor and the Director of the Vision and Image Processing Laboratory, Department of Computer and Information Science, University of Macau, Macau, China. His research interests include chaotic systems, multimedia security, computer vision, and machine learning. He was a recipient of the Third Price of Macau Natural Science Award in 2014. He serves as an Associate Editor for *Neurocomputing*, *Journal of Visual Communication and Image Representation*, and *Signal Processing: Image Communication*. He is the Co-Chair of the Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He is a Senior Member of the International Society for Optical Engineering (SPIE).