Scale-Aware Monocular Visual Odometry and Extrinsic Calibration Using Vehicle Kinematics

Changhyeon Kim, *Member, IEEE*, Youngseok Jang, *Graduate Student Member, IEEE*, Junha Kim, *Graduate Student Member, IEEE*, Pyojin Kim, *Member, IEEE*, and H. Jin Kim, *Member, IEEE*

***Abstract*— This paper proposes a new approach to scale-aware monocular visual odometry (VO) and extrinsic calibration using constraints on camera motion by vehicle kinematics. Main idea is to utilize the Ackermann steering model to observe absolute metric scale in turning motion. To describe motion of the camera attached to the vehicle, we first estimate unknown camera-vehicle relative pose by the proposed extrinsic calibration method. To stably observe scale, we detect turn regions and design an observer to estimate the absolute scale as a function of the camera rotation and direction of translational motion during turning. Using the observed scale, we propose an absolute scale recovery to estimate the unknown scale between turns. Because the proposed scale observer becomes singular near zero rotation, we conduct sensitivity analysis on the scale observer, and investigate appropriate conditions for stable scale estimation. For quantitative evaluation of the extrinsic calibration and the absolute scale recovery, we randomly generate synthetic driving datasets with various noise conditions, and evaluate the performance of each module statistically by Monte-Carlo simulations. To evaluate the overall performance, we implement our method and state-of-the-art monocular and stereo VO methods in the public outdoor driving KITTI dataset, and our method shows competitive scale recovery performance with no external sensor and no assumption on surroundings such as planar ground landmarks. To show promising applicability, we collect real-world driving datasets in two multi-floor underground parking lots, and demonstrate the accurate absolute scale recovery performance of our method in indoor driving situations.**

***Index Terms*— Monocular visual odometry, scale ambiguity, extrinsic calibration, vehicle kinematics.**

Manuscript received 15 December 2022; revised 25 June 2023; accepted 4 August 2023. This work was supported in part by the Unmanned Vehicles Core Technology Research and Development Program through the National Research Foundation of Korea (NRF); and in part by the Unmanned Vehicle Advanced Research Center (UVARC) funded by the Ministry of Science and Information and Communication Technology (ICT), Republic of Korea, under Grant NRF-2020M3C1C1A010864. The Associate Editor for this article was Z. Duric. *(Corresponding author: H. Jin Kim.)*

Changhyeon Kim is with the Department of Aerospace Engineering, Automation and Systems Research Institute (ASRI), Seoul National University (SNU), Seoul 08826, South Korea, and also with Samsung Research, Seoul 06765, South Korea (e-mail: rlackd93@snu.ac.kr).

Youngseok Jang and Junha Kim are with the Department of Mechanical and Aerospace Engineering, Seoul National University (SNU), Seoul 08826, South Korea (e-mail: duscjs59@gmail.com; wnsgk02@snu.ac.kr).

Pyojin Kim is with the School of Mechanical Engineering, Gwangju Institute of Science and Technology (GIST), Gwangju 61005, South Korea (e-mail: pjinkim1215@gmail.com).

H. Jin Kim is with the Department of Aerospace Engineering, Automation and Systems Research Institute (ASRI), Seoul National University (SNU), Seoul 08826, South Korea (e-mail: hjinkim@snu.ac.kr).

Digital Object Identifier 10.1109/TITS.2023.3309833

I. Introduction

NAVIGATION is one of fundamental capabilities for an autonomous mobile vehicle. For navigation, ego-motion estimation using cameras called visual odometry (VO) has received attention due to its compact setting and rich environment expression from an image. Thus, the VO has been actively studied with various configurations: a single camera [1], [2], multiple cameras [3], [4], VO with an inertial measurement unit (IMU) [5], [6], and combining vehicle kinematics [7], [8], [9], [10].

Especially, VO using a single camera, monocular VO (MVO), is an attractive solution for automobile navigation due to its minimal setting. Furthermore, because one or more cameras can be easily found in most vehicles as forms of driving assistant systems and user-mounted dashboard cameras, the MVO implementation targeted for mobile vehicles is highly valuable.

However, due to the monocular projective nature, absolute metric information disappears from an image and MVO can only yield up-to-scale translation motion, which makes the MVO-only setup more challenging without additional metric measurements. This is called the scale ambiguity problem [11]. Although the scale ambiguity often means scale drifts over time, we specifically use it only to denote absolute scale vanishing in this paper.

A common approach to recover the scale in the MVO is to integrate additional sensors providing metric information such as inertial measurements from IMU [5], [12], low-resolution time-of-flight range sensors [13], and a single and multiple distance meters settings [14]. Although utilization of additional sensors can improve the performance, the need for the sensors and precise extrinsic calibration among them might not be affordable for arbitrary settings.

For the ground vehicle settings, a popular approach is to utilize the consistent height of the camera rigidly attached to the vehicle and planar ground observations with a plenty of image features [15], [16], [17], [18], [19], [20], [21], [22]. They show successful performance when planar features are available abundantly. Although they target the ground vehicle applications, the vehicle kinematics is not fully exploited but implicitly considered as a planar and level traverse of the camera. Furthermore, in most research, the relative pose between the camera and the vehicle is commonly assumed to be an identity, which is not always true in vehicular settings.

In this paper, we introduce a scale-aware monocular VO system utilizing a vehicle kinematic motion model. Different from the previous scale-aware MVO works [15], [16], [17], [18], [19], [20], [21], [22], we explicitly use the vehicle kinematics to model the monocular camera motions. To exactly obtain the fixed relative pose of the camera and the vehicle, we develop a self-contained extrinsic pose calibration method between the camera and the vehicle. Then, we design a scale observer that estimates the absolute translation scale from the geometric constraint of the frame-to-frame camera turning motion. To propagate the observed absolute scale on turning regions, we propose a method to recover the unobserved scale between turns.

In the following, we review related works, and list our main contributions compared to them.

A. Related Works

We survey monocular VO methods with scale awareness, and categorize them into three types according to methods to obtain absolute scale information: 1) additional sensors, 2) environmental properties, and 3) learning-based approach.

1) Additional Sensors: Additional sensors are frequently used to observe metric scale of the motion estimation. For VO, multiple cameras [3], [4], [5] with known relative pose and baselines are widely used to triangulate landmarks in 3-D space and estimate the metric camera translation motion. For more compact settings, monocular visual-inertial odometry (V-IO) is proposed [6], [12]. By double-integrating acceleration measurements, the metric translation change of the IMU is incorporated into the MVO motion optimization problem.

Other works utilize different sensor modalities that provide metric information to the MVO framework. In [13] and [14], multiple 1-D point laser sensors are used to obtain the metric distance of the center pixel of the camera and to recover the trajectory scale.

As mentioned before, two hurdles to implement this type of methods exist; the calibration among various sensors with different modalities is nontrivial, and some sensor combinations might not be easily available.

2) Environmental Properties: Most monocular scale-aware VO methods [15], [16], [17], [18], [19], [20], [21], [22] utilize two environmental conditions: planar ground observations and constant camera height. An early work [15] extracts point features on road from a fixed quadrilateral image region to estimate the planar homography transform between frames. By decomposing the homography matrix, they compute the camera height from the plane and adjust the scale of camera motion using consistent camera height assumption.

The strategy using the planar homography is still popular in recent studies, and several variations are proposed to extract planar information accurately and stably; [16], [17] combine sparse features and direct illumination on the ground plane to estimate the homography matrix, and [18] and [22] geometrically model the plane regions by the Delaunay triangulation with point feature nodes and stably prune out outliers. In [20] and [21], robust plane fitting is proposed. In recent work [19], road regions are segmented pixel-wise by deep learning to robustly find planar features.

These methods show the stable and accurate scale maintaining performance in planar feature-abundant environments, however, they may become infeasible in some regions with no texture on ground planes. Furthermore, most research assume known attachment pose of a vehicle-mounted camera, or assume zero-pitch camera pose.

3) Learning-Based Approaches: In recent years, deep learning has undoubtedly achieved considerable advances in computer vision, and many deep applications are derived from several influential works such as [23] and [24]. Following the trend, a number of MVO attempts using deep learning are also introduced [25], [26], [27], [28], [29], [30]. In [25], an end-to-end MVO network is proposed by directly training conventional VO results using deep recurrent convolutional neural network (RCNN), and other methods [26], [27], [28], [29], [31], [32], [33] utilize deep depth prediction in training steps. By using the depth, these methods can provide consistently scaled translation motion over sequences, however, still yield up-to-scale estimation only due to the monocular nature. To fill the metric gap, deep monocular V-IO (MV-IO) methods [31], [32], [33] are emerging recently.

It is noted that most existing learning-based methods require more data than monocular images, such as stereo images [27] or 3-D LiDAR points [28], in the training step of MVO or inference step of MV-IO. Even more, machine learning methods still suffer from generalization gap between training and test sets, and their performance might degrade in unseen conditions.

According to our survey, we found that there are few approaches operating independently of the additional sensors and assumptions on surrounding environments and landmark distributions. Especially, the MVO methods with the scale recovery for vehicles mainly focus on indirectly using vehicle characteristics, such as planar ground features and consistent height of the camera. In several cases, the camera-vehicle relative pose is also assumed to be known.

In this paper, we propose the scale-aware MVO framework that explicitly utilizes the vehicle kinematic constraints on the camera motion. For completeness of the formulation, we also introduce a self-contained camera-vehicle extrinsic pose calibration method.

B. Contributions and Outline

Compared to the related works, we list major contributions of the paper as follows:

- We can estimate arbitrarily attached camera pose to the vehicle by proposing a self-contained camera-vehicle extrinsic pose calibration method using camera motion constrained by vehicle kinematics.
- By utilizing local geometry of the constrained camera motion, we design a new scale observation method when the vehicle turns. We also theoretically analyze the scale observer to determine stable states for observing the scale.
- Unobserved scale between turning regions can be estimated by the absolute scale using metric scale on the turning motion.

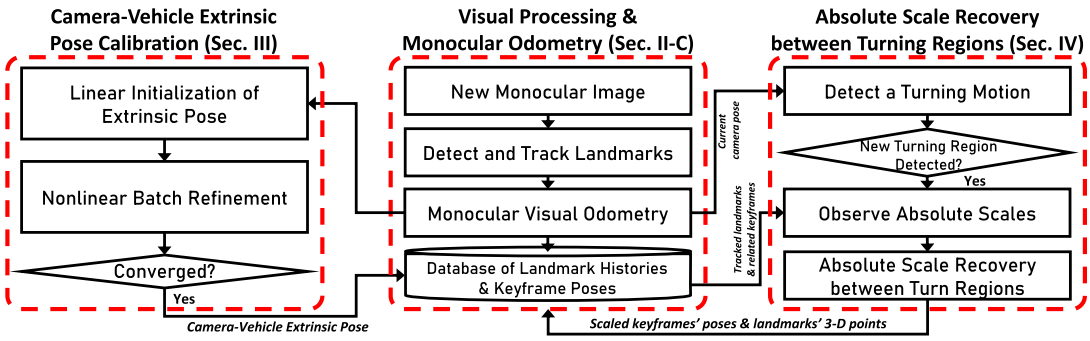

Fig. 1. Block diagram of the proposed scale-aware monocular visual odometry and extrinsic calibration system.

Note that, different from the scale-aware MVO using the planar ground features [15], [16], [17], [18], [19], [20], [21], [22], our method has an additional advantage of no need for assumption on the uncontrollable external environment such as ground feature distributions.

Our algorithm is illustrated in Fig. 1, and the rest of the paper is structured as follows: Section II describes preliminaries including notation rules, vehicle motion model, visual processing and data structures required for our method. In Section III, we explain a camera-vehicle extrinsic pose calibration method to estimate arbitrarily installed monocular camera pose with respect to the vehicle. In Section IV, we propose an absolute scale observer by the kinematically constrained camera motion model in turning motion, and the absolute scale recovery method between turning regions is proposed in Section IV-C. In Section V, we present in-depth analysis on the proposed modules, and demonstrate comparable performance of our method on publicly available datasets. Finally, we highlight the effectiveness of our method especially in indoor driving circumstances by experiments on author-collected indoor driving datasets. Our datasets and related parameters are publicly shared as rosbag files at https://chkim.net/scalemvo.

II. Preliminaries

Before detailed description, we define notations and the 3-D geometry between a monocular camera and landmarks. Then, we derive the monocular camera motion model constrained by the vehicle kinematics. The front-end visual processing and data structures for our system will be explained at the end of this section.

A. Notation Rules and 3-D Geometry of a Camera

Throughout the paper, we express column vectors with bold lowercase letters, and matrices are in bold capital letters. The exception is for using $\mathbf{X}$ to denote a 3-D point. Let $\mathbf{X}_i \in \mathbb{R}^3$ be the i-th 3-D point represented in the world frame $\{W\}$, and $\mathbf{X}_{ij} \in \mathbb{R}^3$ be the expression of $\mathbf{X}_i$ in the j-th camera frame $\{C_j\}$. The 3-D rotation matrix and translation vector from $\{C_a\}$ to $\{C_b\}$ are described as $\mathbf{R}_{C_b}^{C_a} \in \mathrm{SO}(3)$ and $\mathbf{t}_{C_b}^{C_a} \in \mathbb{R}^3$, respectively, and the corresponding rigid body transform is defined as $\mathbf{T}_{C_b}^{C_a} := \left[\mathbf{R}_{C_b}^{C_a}, \mathbf{t}_{C_b}^{C_a}; \mathbf{0}_3^\top, 1\right] \in \mathrm{SE}(3)$ where $\mathbf{0}_3$ is a 3-D zero vector. The projection relationship of $\mathbf{X}_i$ to the corresponding 2-D pixel $\mathbf{p}_{ij} \in \mathbb{R}^2$ on the pixel plane of j-th camera frame can be computed by $\pi_j(\mathbf{X}_i) \in \mathbb{R}^2$ by defining a function $\pi_j(\cdot) : \mathbb{R}^3 \mapsto \mathbb{R}^2$ projecting a 3-D point expressed in $\{W\}$ onto the pixel plane of $\{C_j\}$. For simplicity, we use abbreviated notations $\mathrm{c}(\cdot)$, $\mathrm{s}(\cdot)$, $\mathrm{t}(\cdot)$ throughout the paper to denote cosine, sine, and tangent functions, respectively.

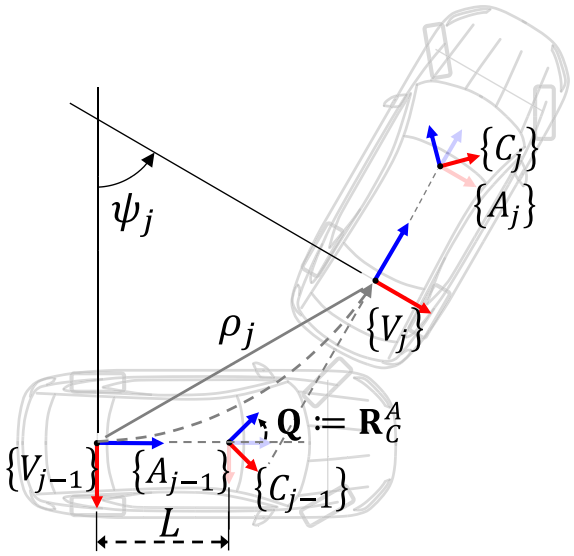

Fig. 2. Illustration of the vehicle kinematics This figure shows the Ackermann steering geometry of the vehicle between $\{V_0\}$ and $\{V_1\}$. Red and blue arrows denote x- and z-axes of each frame. By the right-hand rule, the y-axis directs to the paper. The shaded frames are auxiliary camera frames.

B. Camera Motion Constrained by Vehicle Kinematics

As depicted in Fig. 2, the chassis part of a four-wheeled automotive vehicle is designed for all wheels to experience concentric circular motions. This kinematics, called the Ackermann steering geometry [34], enforces locally planar and circular motion.

As shown in Fig. 2, we consider that the vehicle frame $\{V\}$ is on the rear axle of the vehicle, and the z- and x-axes of $\{V\}$ head forward and right of the vehicle, respectively. Using this, the vehicle motion from $\{V_{j-1}\}$ to $\{V_j\}$ can be represented as

$$\mathbf{R}_{V_j}^{V_{j-1}} = \begin{bmatrix} \mathrm{c}(\psi_j) & 0 & \mathrm{s}(\psi_j) \\ 0 & 1 & 0 \\ -\mathrm{s}(\psi_j) & 0 & \mathrm{c}(\psi_j) \end{bmatrix}, \mathbf{t}_{V_j}^{V_{j-1}} = \begin{bmatrix} \rho_j \mathrm{s}(\gamma_j) \\ 0 \\ \rho_j \mathrm{c}(\gamma_j) \end{bmatrix}, \quad (1)$$

where $\psi_j, \rho_j \in \mathbb{R}$ are turning angle and distance between centers of $\{V_{j-1}\}$ to $\{V_j\}$, respectively, and $\gamma_j := \psi_j/2$.

The motion of the camera rigidly attached to the vehicle can be modeled by the vehicle kinematics. We consider that the original camera frame $\{C\}$ is at the distant $L \in \mathbb{R}$ from the origin of $\{V\}$ along the z-axis of $\{V\}$, and the camera pose is $\mathbf{Q} \in \mathrm{SO}(3)$. We additionally define an auxiliary camera frame $\{A\}$ sharing the origin of $\{C\}$ but having the same pose with $\{V\}$, i.e., $\mathbf{R}_A^V = \mathbf{I}_3 \in \mathrm{SO}(3)$. The translation vector between $\{V\}$ and $\{A\}$ is $\mathbf{t}_A^V = [0, 0, L]^\top$ where $\mathbf{I}_3$ is a 3-D identity matrix.

The relative motion between $\{A_{j-1}\}$ and $\{A_j\}$ $\mathbf{T}_{A_j}^{A_{j-1}} \in \mathrm{SE}(3)$ can be represented as

$$\mathbf{T}_{A_j}^{A_{j-1}} = \mathbf{T}_V^A \mathbf{T}_{V_j}^{V_{j-1}} \mathbf{T}_A^V, \tag{2}$$

where rotation and translation parts of $\mathbf{T}_{A_j}^{A_{j-1}}$ are written as

$$\mathbf{R}_{A_j}^{A_{j-1}} = \begin{bmatrix} \mathrm{c}(\psi_j) & 0 & \mathrm{s}(\psi_j) \\ 0 & 1 & 0 \\ -\mathrm{s}(\psi_j) & 0 & \mathrm{c}(\psi_j) \end{bmatrix}, \mathbf{t}_{A_j}^{A_{j-1}} = \begin{bmatrix} \rho_j \mathrm{s}(\gamma_j) + L\mathrm{s}(\psi_j) \\ 0 \\ \rho_j \mathrm{c}(\gamma_j) - L + L\mathrm{c}(\psi_j) \end{bmatrix}. \tag{3}$$

Finally, the constrained camera motion $\mathbf{T}_{C_j}^{C_{j-1}} \in \mathrm{SE}(3)$ can be written as

$$\mathbf{T}_{C_j}^{C_{j-1}} = \mathbf{T}_A^C \mathbf{T}_{A_j}^{A_{j-1}} \mathbf{T}_C^A = \begin{bmatrix} \mathbf{Q}^\top \mathbf{R}_{A_j}^{A_{j-1}} \mathbf{Q} & \mathbf{Q}^\top \mathbf{t}_{A_j}^{A_{j-1}} ; \mathbf{0}_3^\top 1 \end{bmatrix} \tag{4}$$

where $\mathbf{T}_C^A = \begin{bmatrix} \mathbf{Q}, \mathbf{0}_3; \mathbf{0}_3{}^\top, 1 \end{bmatrix} \in \mathrm{SE}(3)$.

The above derivation is analogous to the vehicular MVO research [7] incorporating the vehicle kinematics to make the 1-point MVO. However, [7] used two major assumptions: zero displacement $L = 0$ and ideal camera pose $\mathbf{Q} = \mathbf{I}_3$, which might be invalid in general camera settings. In fact, the author of [7] reported that the two assumptions are valid only when the steering angle is sufficiently small. In the large steering motion, L is no longer negligible because of increasing terms multiplied by L in $\mathbf{t}_{A_j}^{A_{j-1}}$.

In this paper, to deal with the general camera installation, we propose the camera-vehicle extrinsic pose calibration method in Section III. In addition, we consider the nonzero L to realize the scale-aware MVO, which will be detailed in Section IV.

C. Visual Processing Front-End and Data Structures

Our method utilizes associations between visual landmark correspondences and camera frames. We list requirements for our scale awareness module.

Each visual landmark should store:

- 2-D pixel tracking history over images
- Address of frames where the landmark was seen
- 3-D point of the landmark represented in $\{W\}$

Each image frame should include:

- Address of landmarks observed in the frame
- 6-DoF camera motion from $\{W\}$

For the visual landmark, we use the ORB feature [35]. Our method selects keyframes among the image frames to reduce the problem size and obtain the sufficiently large turning motion between frames. We implement our VO module by following the successful MVO method, ORB-SLAM2 [1].

III. Camera-Vehicle Extrinsic Pose Calibration

The exact extrinsic pose $\mathbf{Q}$ of vehicle-installed cameras, such as driving assistance cameras and custom dashcams, are not generally available. In this section, we introduce the two-step calibration method to estimate the camera-vehicle extrinsic pose by only using motion of the camera installed to the vehicle.

A. Problem Formulation

The kinematic constraint of the vehicle is generated by the chassis part. In normal driving conditions, the vehicle upper body can be considered to experience the same rigid body motion with the chassis. In this case, the motion of the camera attached to the body part can be also expressed by the constrained motion model (4).

Based on the above description, desired conditions of the calibration problem for the j-th frame are written as

$$\hat{\mathbf{R}}_j = \mathbf{Q}^\top \mathbf{R}_j \mathbf{Q}, \quad \hat{\mathbf{t}}_j = \mathbf{Q}^\top \mathbf{t}_j, \tag{5}$$

where we define simplified notations $\hat{\mathbf{R}}_j := \mathbf{R}_{C_j}^{C_{j-1}}$, $\mathbf{R}_j := \mathbf{R}_{A_j}^{A_{j-1}} \in \mathrm{SO}(3)$ and $\hat{\mathbf{t}}_j := \mathbf{t}_{C_j}^{C_{j-1}}, \mathbf{t}_j := \mathbf{t}_{A_j}^{A_{j-1}} \in \mathbb{R}^3$, respectively. Note that unconstrained camera motion $\hat{\mathbf{R}}_j$ and $\hat{\mathbf{t}}_j$ can be computed by the MVO algorithm.

As (3) and (4), right-hand sides of two equations in (5) are functions of $\mathbf{q}_s$, ρ_j, and ψ_j where $\mathbf{q}_s \in \mathbb{R}^4$ is a unit quaternion of $\mathbf{Q}$. Let us define a parameter vector with unknowns as

$$\Theta = \begin{bmatrix} \mathbf{q}_s{}^\top, \rho_1, \cdots, \rho_N, \psi_1, \cdots, \psi_N \end{bmatrix}^\top \in \mathbb{R}^{2N+4}. \tag{6}$$

By aggregating N poses, an optimization problem with respect to Θ can be formulated as

$$\underset{\Theta}{\operatorname{argmin}} \sum_{j=1}^{N} \|\hat{\mathbf{R}}_j - \mathbf{Q}^\top \mathbf{R}_j \mathbf{Q}\|_F^2 + \|\hat{\mathbf{t}}_j - \mathbf{Q}^\top \mathbf{t}_j\|_F^2 \tag{7}$$

where $\|\cdot\|_F$ is the Frobenius norm.

Note that the problem in (7) is a large-scale nonlinear batch optimization problem. Without proper initial parameter values, it could fall into wrong minima, or diverge. To prevent this, we first calculate initial guess of each part of Θ separately by linear algebraic approaches.

B. Linear Initialization of ψ_j and $\mathbf{Q}$

First, we explain how to extract initial guesses of turning angles ψ_j from the unconstrained camera rotations $\hat{\mathbf{R}}_j$ regardless of the unknown $\mathbf{Q}$. Then, using the initial ψ_j, we propose a linear algebraic approach to calculate the initial value of $\mathbf{q}_s$.

1) Extracting ψ_j From the Unconstrained Rotation $\hat{\mathbf{R}}_j$: In the rotation part of (5), $\hat{\mathbf{R}}_j$ and $\mathbf{R}_j$ are *similar* matrices by $\mathbf{Q}$. By the property of similar matrices, the two matrices should have the same eigenvalues regardless of a choice of $\mathbf{Q} \in \mathrm{SO}(3)$.

From the definition of $\mathbf{R}_j$ in (3), three eigenvalues of $\mathbf{R}_j$ are one and $\mathrm{c}(\psi_j) \pm i\,\mathrm{s}(\psi_j)$ where i is the unit imaginary

number. They are also eigenvalues of $\hat{\mathbf{R}}_j$ due to the eigenvalue invariance of the similar matrices. Because the sum of eigenvalues is equal to the trace of the matrix, we can derive an equation related to ψ_j as

$$\operatorname{trace}\hat{\mathbf{R}}_j = 2\,\mathrm{c}(\psi_j) + 1. \tag{8}$$

From (8), we can only obtain the magnitude $|\psi_j|$. To determine its sign, we employ $\mathbf{t}_j$. Due to the Ackermann geometry, $\hat{\mathbf{t}}_j$ is locally constrained to the x-z plane of $\{A\}$.

When the steering motion is larger than the roll and pitch motion, we can consider that a vector rotation $\hat{\mathbf{t}}'_j := \hat{\mathbf{R}}_j\hat{\mathbf{t}}_j \in \mathbb{R}^3$ is mainly governed by the steering motion. Then, from the directional difference between $\hat{\mathbf{t}}'_j$ and $\hat{\mathbf{t}}_j$, we can compute the direction of rotation of the vehicle.

In sum, the initial guess of ψ_j can be calculated as a closed form with a 3-D cross product operator $\times$ as

$$\psi_j = \operatorname{sign}\left(\hat{\mathbf{t}}_j \times \hat{\mathbf{t}}'_j\right) \cdot \left|\arccos\left(\frac{\operatorname{trace}\hat{\mathbf{R}}_j - 1}{2}\right)\right|. \tag{9}$$

2) Linear Solution of $\mathbf{Q}$ *in a Quaternion Representation:* Using the initial guesses ψ_j, $\mathbf{Q}$ can be estimated by solving the least squares problem in quaternion space. Let $\hat{\mathbf{q}}_j, \mathbf{q}_j \in \mathbb{R}^4$ be unit quaternions of $\hat{\mathbf{R}}_j$ and $\mathbf{R}_j$, respectively. In this paper, we follow the Hamilton quaternion convention with the right-handed algebra. We define the pure quaternion of the vector $\mathbf{v} \in \mathbb{R}^3$ with zero at the first element as $\breve{\mathbf{v}} := \left[0,\ \mathbf{v}^\top\right]^\top \in \mathbb{R}^4$. Then, (5) can be rewritten as

$$\begin{aligned}\mathbf{Q}\hat{\mathbf{R}}_j = \mathbf{R}_j\mathbf{Q} &\rightarrow \mathbf{q}_s \otimes \hat{\mathbf{q}}_j = \mathbf{q}_j \otimes \mathbf{q}_s \\ \mathbf{Q}\hat{\mathbf{t}}_j = \mathbf{t}_j &\rightarrow \mathbf{q}_s \otimes \breve{\hat{\mathbf{t}}}_j \otimes \mathbf{q}_s^* = \breve{\mathbf{t}}_j,\end{aligned} \tag{10}$$

where $\otimes$ means the quaternion product operator, and $\mathbf{q}^* \in \mathbb{R}^4$ denotes conjugate of a quaternion $\mathbf{q}$. Because the MVO can only provide up-to-scale translation motion, we use unit vectors $\hat{\mathbf{u}}_j$ and $\mathbf{u}_j$ corresponding to $\hat{\mathbf{t}}_j$ and $\mathbf{t}_j$, respectively.

The quaternion equations in (10) can be transformed into matrix forms,

$$\begin{aligned}\Omega_r\left(\hat{\mathbf{q}}_i\right)\mathbf{q}_s &= \Omega_l\left(\mathbf{q}_j\right)\mathbf{q}_s \\ \Omega_r\left(\breve{\hat{\mathbf{u}}}_i\right)\mathbf{q}_s &= \Omega_l\left(\breve{\mathbf{u}}_i\right)\mathbf{q}_s,\end{aligned} \tag{11}$$

where matrix forms of left and right quaternion products $\Omega_l(\mathbf{q})$, $\Omega_r(\mathbf{q}) : \mathbb{R}^4 \mapsto \mathbb{R}^{4\times4}$ are denoted as

$$\Omega_l(\mathbf{q}) = \begin{bmatrix} q_0 & -\mathbf{n}^\top \\ \mathbf{n} & q_0\mathbf{I}_3 + [\mathbf{n}]_\times \end{bmatrix},\ \Omega_r(\mathbf{q}) = \begin{bmatrix} q_0 & -\mathbf{n}^\top \\ \mathbf{n} & q_0\mathbf{I}_3 - [\mathbf{n}]_\times \end{bmatrix} \tag{12}$$

where $\mathbf{q} := \left[q_0, \mathbf{n}^\top\right]^\top$ with a scalar q_0 and $\mathbf{n} \in \mathbb{R}^3$, and $[\mathbf{n}]_\times \in \mathbb{R}^{3\times3}$ is a matrix satisfying $[\mathbf{n}]_\times\mathbf{v} = \mathbf{n} \times \mathbf{v}$ with $\mathbf{v} \in \mathbb{R}^3$.

Note that, in this initialization step, we temporally assume $L = 0$ to neglect unknown values ρ_j of $\mathbf{t}_j$. Then, the simplified form of $\mathbf{u}_j$ becomes a function of only ψ_j

$$\mathbf{u}_j = \frac{\mathbf{t}_j}{\|\mathbf{t}_j\|_2} = \begin{bmatrix} \rho_j\mathrm{s}(\gamma_j) + L\mathrm{s}(\psi_j) \\ 0 \\ \rho_j\mathrm{c}(\gamma_j) - L + L\mathrm{c}(\psi_j) \end{bmatrix} / \|\mathbf{t}_j\|_2 \approx \begin{bmatrix} \mathrm{s}(\gamma_j) \\ 0 \\ \mathrm{c}(\gamma_j) \end{bmatrix} \tag{13}$$

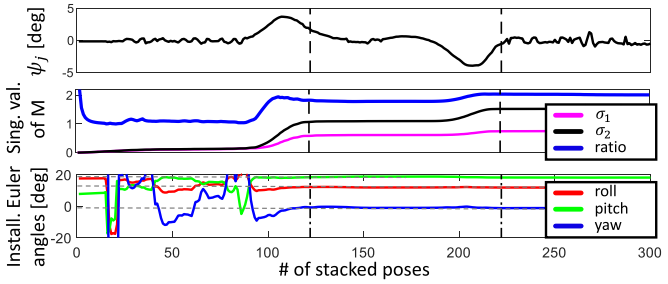

Fig. 3. Singular value history of the linear initialization of $\mathbf{q}_s$ In the first graph, turning angles are obtained by the ground truth poses to the first 500 frames of 00 dataset [36]. σ_1 and σ_2 are the two smallest singular values of $\mathbf{M}$, respectively, and their histories are drawn according to the number of stacked poses in the middle graph. The third graph exhibits histories of the estimated Euler angles of $\mathbf{Q}$. Two vertical lines denote ends of each turning motion, and horizontal dashed lines are true values of Euler angles of $\mathbf{Q}$. We intentionally rotate the camera pose with Euler angles of $\{12, 19, -1\}$ degrees to simulate an arbitrary pose $\mathbf{Q}$.

ρ_j values will be re-considered in a refinement step in the following subsection.

By concatenating N equations, a least squares problem to estimate $\mathbf{q}_s$ with a matrix $\mathbf{M} \in \mathbb{R}^{8N\times4}$ can be formulated as

$$\mathbf{M}\mathbf{q}_s = \begin{bmatrix} \Omega_r\left(\hat{\mathbf{q}}_1\right) - \Omega_l\left(\mathbf{q}_1\right) \\ \Omega_r\left(\breve{\hat{\mathbf{u}}}_1\right) - \Omega_l\left(\breve{\mathbf{u}}_1\right) \\ \vdots \\ \Omega_r\left(\hat{\mathbf{q}}_N\right) - \Omega_l\left(\mathbf{q}_N\right) \\ \Omega_r\left(\breve{\hat{\mathbf{u}}}_N\right) - \Omega_l\left(\breve{\mathbf{u}}_N\right) \end{bmatrix} \mathbf{q}_s = \mathbf{0}_{8N}. \tag{14}$$

A solution $\mathbf{q}_s$ can be computed by the right nullspace of $\mathbf{M}$. Using the singular value decomposition to $\mathbf{M}$, we can obtain the solution as the right singular vector corresponding to the smallest singular value.

In Fig. 3, the estimated $\mathbf{q}_s$ converges to the truth value at the first turning motion, and the solution becomes stable with the dashed line after another turn. Note that the nullspace problem in (14) might yield a wrong solution with duplicated singular values for insufficiently small turning motion. We can estimate the uniqueness of the solution by checking whether the smallest two singular values differ or not. As seen in Fig. 3, the second smallest singular value becomes distinguishable from the smallest one after the first sufficient turning motion.

C. Full Refinement of the Initial Guesses

In the previous linear step, we assume $L = 0$ for simple derivations. In this step, we re-consider the nonzero L to incorporate effects of $\mathrm{s}(\psi_j)$ and $\mathrm{c}(\psi_j) - 1$ terms of $\mathbf{t}_j$. Without loss of generality, we use $L = 1$. Because we cannot obtain the scale of $\mathbf{t}_j$ from the MVO, we use the normalized translation vector $\mathbf{t}_j / \|\mathbf{t}_j\|_2$ in the full refinement problem. Then, we modify the problem in (7) as

$$\begin{aligned}\underset{\Theta}{\operatorname{argmin}} \sum_{j=1}^{N} w_H \Bigg(& \|\hat{\mathbf{R}}_j - \mathbf{Q}^\top\mathbf{R}_j\mathbf{Q}\|_F^2 \\ & + \|\left(\mathbf{Q}\hat{\mathbf{u}}_j\right)^\top \mathbf{t}_j / \|\mathbf{t}_j\|_2 - 1\|_2^2 \Bigg) \\ \text{subject to } & \|\mathbf{q}_s\|_2 = 1.\end{aligned} \tag{15}$$

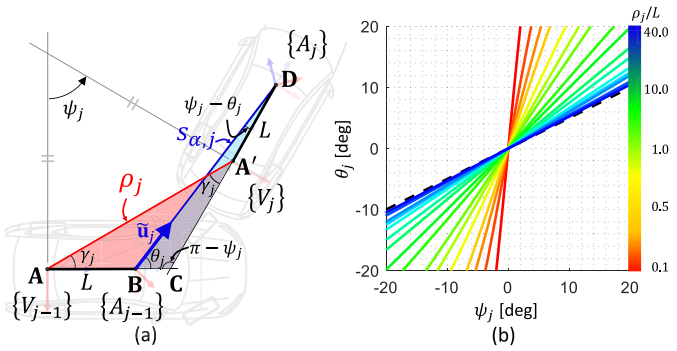

Fig. 4. Two triangles formed by the turn of the vehicle and relationship between ψ and θ (a) ψ_j is the turn angle of the vehicle, and θ_j is the subtended angle between the z-axis of $\{A_{j-1}\}$ and the rotated unit translation vector $\tilde{\mathbf{u}}_j$. ρ_j is the distance between centers of $\{V_{j-1}\}$ and $\{V_j\}$, and $s_{\alpha,j}$ is the scale of the monocular translation motion which is our objective to estimate. (b) The relationship between ψ and θ is plotted with respect to the value of ρ_j/L.

As seen in (15), the original translation term in (7) is modified into the difference of unit directions of two translation representations to delete unknown magnitude of the estimated monocular translation motion.

The real-world vehicle motion could slightly deviate from the planar motion model. To suppress bad effects of the off-planar motions on the optimization process, we employ the Huber norm $w_H(r)$ with the threshold value $r_{th} \in \mathbb{R}^+$ as

$$w_H(r) = \begin{cases} r_{th}/|r| & \text{if } |r| > r_{th} \\ 1 & \text{otherwise} \end{cases}, \tag{16}$$

where r_{th} is set to 60 % value of the residuals, and recalculated for each optimization step.

The above nonlinear optimization problem can be efficiently solved by a nonlinear programming solver, CasADi [37]. Note that the resulting scale estimations ρ_j are proportional to the L. If the exact metric L can be known in advance, the absolute value of ρ_j can be estimated. Reversely, the metric L can be recovered using metric motion measurements from additional sensors, such as wheel odometer and global positioning system (GPS). Anyway, any choice of real positive L does not affect estimating $\mathbf{Q}$.

IV. Absolute Scale Recovery Between Turning Regions

In this section, we introduce a method to observe the absolute metric scale of the MVO motion, and a strategy to detect turning frame regions that can provide absolute scale observations stably. Then, we propose an absolute scale recovery (ASR) method scaling translation motion and 3-D points between turns by using the observed absolute scale of the turn regions.

A. Observing Absolute Scale via Kinematic Geometry

We detail how to observe the absolute scale $s_{\alpha,j}$ of the monocular camera translation motion $\mathbf{t}_j$ at the vehicle turning. When the vehicle turns to angle of ψ_j, we can draw two triangles by joining origins of vehicle body frames and camera frames as depicted in Fig. 4(a). For the red isosceles triangle $\triangle ACA'$, we calculate lengths of $\overline{AC}$ and $\overline{A'C}$ as

$$\overline{AC} = \overline{A'C} = \frac{\rho_j}{2\,\mathrm{c}(\gamma_j)}. \tag{17}$$

By using $\overline{AC}$ and $\overline{A'C}$, each side of the blue triangle $\triangle BCD$ can be calculated with $\overline{AB} = \overline{A'D} = L$ as

$$\overline{BC} = \frac{\rho_j}{2\,\mathrm{c}(\gamma_j)} - L, \ \overline{CD} = \frac{\rho_j}{2\,\mathrm{c}(\gamma_j)} + L. \tag{18}$$

As seen in Fig. 4(a), our objective $s_{\alpha,j}$ is a side of the blue triangle. If we know angles ψ_j, θ_j, and γ_j, we can calculate $s_{\alpha,j}$ by utilizing the sine rule on the blue triangle. Because the initial value of ψ_j can be known by (9) and $\gamma_j = 1/2\psi_j$, we can compute the unknown value of the angle θ_j that is the subtended angle between the z-axis of $\{A_{j-1}\}$ and a unit vector $\tilde{\mathbf{u}}_j := \mathbf{Q}\hat{\mathbf{u}}_j \in \mathbb{R}^3$ rotated to the auxiliary frame. By defining $\mathbf{k}_V \in \mathbb{R}^3$ as the unit vector of the z-axis of $\{V\}$, θ_j is calculated as

$$\theta_j = \arctan\left\{(\mathbf{k}_V \times \tilde{\mathbf{u}}_j)/\left(\mathbf{k}_V{}^\top \tilde{\mathbf{u}}_j\right)\right\}. \tag{19}$$

Applying the sine rule on the blue triangle, we finally obtain equality,

$$\frac{\frac{\rho_j}{2\,\mathrm{c}(\gamma_j)} - L}{\mathrm{s}(\psi_j - \theta_j)} = \frac{\frac{\rho_j}{2\,\mathrm{c}(\gamma_j)} + L}{\mathrm{s}(\theta_j)} = \frac{s_{\alpha,j}}{\mathrm{s}(\psi_j)}. \tag{20}$$

From the first equality in (20), the temporal distance ρ_j of the vehicle is expressed as a function of ψ_j and θ_j up to L,

$$\frac{\rho_j}{L} = 2\,\mathrm{c}(\gamma_j)\,\frac{\mathrm{s}(\theta_j) + \mathrm{s}(\psi_j - \theta_j)}{\mathrm{s}(\theta_j) - \mathrm{s}(\psi_j - \theta_j)}. \tag{21}$$

Then, the scale observer can be derived by substituting (21) to the second equality of (20) as

$$\frac{s_{\alpha,j}}{L} = \frac{2\,\mathrm{s}(\psi_j)}{\mathrm{s}(\theta_j) - \mathrm{s}(\psi_j - \theta_j)}. \tag{22}$$

In Fig. 4(b), the graph of ψ_j and θ_j with respect to various ρ_j/L settings is depicted. θ_j can be derived from (21) as

$$\theta_j = \arctan\left(\frac{\rho_j \mathrm{t}(\gamma_j) + 2L\,\mathrm{s}(\gamma_j)}{\rho_j - 2L\,\mathrm{s}(\gamma_j)\,\mathrm{t}(\gamma_j)}\right), \tag{23}$$

where θ_j is a function of ψ_j and ρ_j, which allows us to treat $s_{\alpha,j}$ as a function of ψ_j and θ_j.

As seen in the figure, θ_j is approximately proportional to ψ_j for all ρ_j/L. We found that the ratio θ_j/ψ_j asymptotically converges to 0.5 when ρ_j goes to infinite, which guarantees a nonzero positive denominator of (22) by assuming $|\psi_j| = |2\gamma_j| < \pi$. Because general vehicles cannot steer over 90 degrees in a short period like the camera image acquisition interval, the turning angle assumption is valid in most cases.

The scale observer in (22) becomes singular when ψ_j goes to zero. To discuss this problem, in Section V-A, we will investigate the relationship among ψ_j, θ_j, and $s_{\alpha,j}$, and analyze which condition is desirable to stably observe the scale by sensitivity analysis on the scale observer.

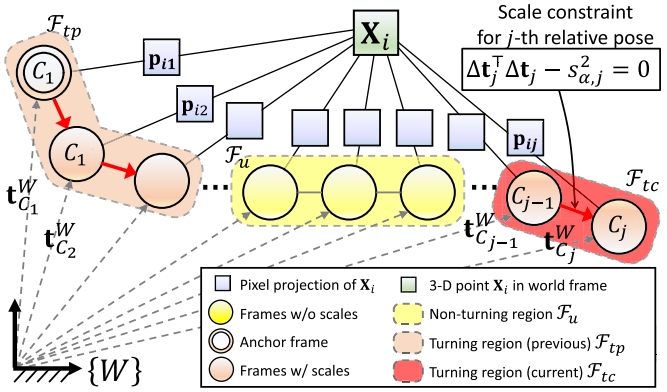

Fig. 5. Factor graph of a landmark and related keyframes for the absolute scale recovery An i-th landmark is connected to turning and non-turning keyframes by 2-D pixel tracks. We use 1's index rule in this illustration. The red arrow means a constrained relative translation motion in the turning regions. All translation motion of the keyframes are represented with respect to the world frame.

Algorithm 1 Detecting a New Turning Region

i_{op}: an operating indicator. Default is True.
n: a counter value. Default is zero.
for each incoming frame, current j -th frame **do**
  Do monocular VO
  $|\psi_j| \leftarrow$ a steering angle calculated by (9)
  if $|\psi_j| \geq \psi_{th}$ **then**
    if $!i_{op}$ **then**
      $i_{op} \leftarrow$ True
    end if
    $\mathcal{F}_{tc} \leftarrow \mathcal{F}_{tc} \cup j$; $++n$;
  else
    $i_{op} \leftarrow$ False; $n \leftarrow 0$;
    if $n \geq n_{th}$ **then**
      $\mathcal{F}_{tp} \leftarrow \mathcal{F}_{tc}$; $\mathcal{F}_{tc} \leftarrow \varnothing$;
    else
      $\mathcal{F}_u \leftarrow \mathcal{F}_u \cup \mathcal{F}_{tc}$; $\mathcal{F}_{tc} \leftarrow \varnothing$;
    end if
  end if
end for

B. Detecting Turning Regions

As denoted in the previous subsection, the scale observer (22) becomes singular for small turning angle ψ_j. To obtain the reliable observations, we detect keyframes with sufficiently large turning motion.

Let $\mathcal{F}$ be an index set of all keyframes between turning regions. $\mathcal{F}$ consists of two subsets, $\mathcal{F}_t$ and $\mathcal{F}_u$, which are index sets of keyframes on turning and non-turning regions, respectively. We additionally separate $\mathcal{F}_t$ into two index sets of previous and current turning regions, $\mathcal{F}_{tp}$ and $\mathcal{F}_{tc}$, respectively. Each index set is depicted as a shaded region with dashed boundary in Fig. 5.

Once $|\psi_j|$ becomes larger than a threshold angle ψ_{th}, the j-th keyframe is regarded as a turning candidate frame, and we count how many candidates follow sequentially. If the number of the candidates exceeds a threshold count value n_{th}, we find a new turning region $\mathcal{F}_{tc}$ from the j-th keyframe to a keyframe whose next keyframe is no longer the candidate frame. If not, all the candidates from the j-th keyframe are passed to the non-turning region index set $\mathcal{F}_u$. This procedure is written in Algorithm 1.

C. Recovering Unknown Scale by Nonlinear Programming With Equality Constraints

In this subsection, we introduce the ASR module. Using the observed scale values on the turning keyframes $\mathcal{F}_t$, we recover unknown scale values of the monocular translation motion and 3-D landmark points between the turning regions $\mathcal{F}_u$.

Fig. 5 illustrates a factor graph of the i-th landmark and its related keyframes. The landmark is associated to the keyframes by 2-D pixel tracks $\mathbf{p}_{ij} \in \mathbb{R}^2$. The pixel reprojection error $\mathbf{r}_{ij}$ induced by $\mathbf{X}_i$ and $\{C_j\}$ is written as

$$\mathbf{r}_{ij} := \pi_j(X_i) - \mathbf{p}_{ij} \in \mathbb{R}^2. \tag{24}$$

By aggregating all error vectors generated by N keyframes and M landmarks, we define the residual vector $\mathbf{r}$ as

$$\mathbf{r} := \left[o_{11}\mathbf{r}_{11}{}^\top, \cdots, o_{NM}\mathbf{r}_{NM}{}^\top\right]^\top \in \mathbb{R}^{2MN}, \tag{25}$$

where an indicator o_{ij} becomes true if the i-th point is seen in the j-th keyframe, otherwise false.

We define the parameter vector ζ to be scaled as

$$\zeta := \left[\mathbf{t}_2^{W\top}, \cdots, \mathbf{t}_N^{W\top}, \mathbf{X}_1{}^\top, \cdots, \mathbf{X}_M{}^\top\right]^\top \in \mathbb{R}^P, \tag{26}$$

where $P := 3(N-1) + 3M$ and we fix the first keyframe $\{C_1\}$ to avoid the gauge freedom. Then, we can formulate a reprojection error minimization problem with respect to ζ as

$$\underset{\zeta}{\operatorname{argmin}}\ \mathbf{r}(\zeta)^\top \mathbf{r}(\zeta), \tag{27}$$

where $\mathbf{r}(\zeta) \in \mathbb{R}^{2MN}$ is the residual vector as a function of ζ.

The optimization problem in (27) is a popular nonlinear programming problem in computer vision, called bundle adjustment (BA) [11]. Different from the original BA, we additionally incorporate the observed scale values into the problem to recover the unobserved scale between turns.

Conceptually, the scale of the turning keyframes can be propagated to the associated non-turning keyframes through the 2-D pixel tracks. Like the red arrows depicted in Fig. 5, the observed scale on $\mathcal{F}_t$ can be used to constrain the relative translation motion $\Delta\mathbf{t}_j := \mathbf{t}_{C_j}^W - \mathbf{t}_{C_{j-1}}^W \in \mathbb{R}^3$. If the cardinality of $\mathcal{F}_t$ is K and the k-th element of $\mathcal{F}_t$ is $\mathcal{F}_t(k)$, the k-th scale constraint can be written as an equality constraint

$$g_k(\zeta, \mathbf{s}_\alpha) = \Delta\mathbf{t}_{\mathcal{F}_t(k)}{}^\top \Delta\mathbf{t}_{\mathcal{F}_t(k)} - s_{\alpha,\mathcal{F}_t(k)}^2 = 0, \tag{28}$$

where $\mathbf{s}_\alpha := \left[s_{\alpha,\mathcal{F}_t(1)}, \cdots, s_{\alpha,\mathcal{F}_t(K)}\right]^\top \in \mathbb{R}^K$.

Defining the Lagrangian $L(\zeta, \lambda) := \mathbf{r}(\zeta)^\top \mathbf{r}(\zeta) + \lambda^\top \mathbf{g}(\zeta, \mathbf{s}_\alpha) \in \mathbb{R}$ with the Lagrange multiplier vector $\lambda \in \mathbb{R}^K$, we finally set up a minimization problem as

$$\underset{\zeta,\lambda}{\operatorname{argmin}}\ L(\zeta, \lambda) \ \text{ subject to } \mathbf{g}(\zeta, \mathbf{s}_\alpha) = \mathbf{0}_K, \tag{29}$$

where an equality constraint vector $\mathbf{g}$ is defined as

$$\mathbf{g}(\zeta, \mathbf{s}_\alpha) := \left[g_1(\zeta, \mathbf{s}_\alpha), \cdots, g_K(\zeta, \mathbf{s}_\alpha)\right]^\top : \mathbb{R}^P \mapsto \mathbb{R}^K. \tag{30}$$

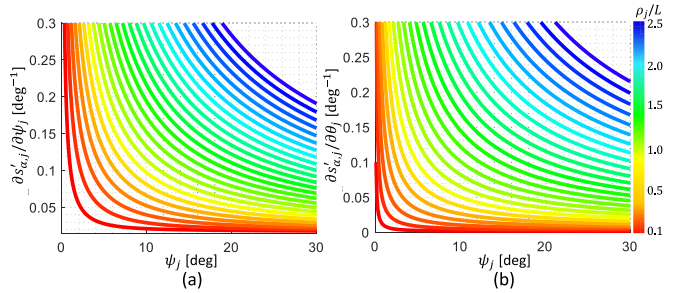

Fig. 6. Noise sensitivities of the scale observer (a) Noise sensitivity with respect to ψ_j, (b) noise sensitivity with respect to θ_j. The curves are color-coded according to ρ_j/L.

The above nonlinear programming with equality constraints can be solved by sequential quadratic programming (SQP) [38]. Whenever a new turning region is detected, we operate the ASR module, and repeat this procedure for overall image sequences.

V. Experimental Results

In this section, we analyze the proposed three modules: the scale observer, the camera-vehicle extrinsic calibration, and the ASR module. Then, we evaluate the overall MVO performance of our method on publicly available outdoor driving datasets, i.e. KITTI odometry datasets [36]. At the end, we demonstrate promising real-world applicability of the proposed method on author-collected indoor driving datasets acquired in two different underground parking lots with multiple floors.

A. Noise Sensitivity Analysis of the Scale Observer

First, we perform in-depth analysis on the scale observer. Noise in ψ_j and θ_j estimation is inevitable due to imperfect camera motion estimation and off-planar vehicle vibration. In Section IV-A, (22) is ill-defined near $\psi_j = \theta_j = 0$, which implies high noise-sensitivity around the zero. To address this problem, we analyze the noise sensitivity of the scale observer $s_{\alpha,j}$ with respect to ψ_j and θ_j with various ρ_j/L settings. Without loss of generality, we consider the normalized scale $s'_{\alpha,j} := s_{\alpha,j}/L$ during this analysis.

We differentiate $s'_{\alpha,j}$ with respect to the two parameters, ψ_j and θ_j, and the resulting sensitivity equations are as below:

$$\frac{\partial s'_{\alpha,j}}{\partial \psi_j} = \frac{-\mathrm{s}(\theta_j)}{\mathrm{c}(\psi_j - 2\theta_j) - 1}, \quad \frac{\partial s'_{\alpha,j}}{\partial \theta_j} = \frac{2\left(\mathrm{s}(\psi_j - \theta_j) + \mathrm{s}(\theta_j)\right)}{\mathrm{c}(\psi_j - 2\theta_j) - 1}. \tag{31}$$

Using the above two derivatives, we draw multiple graphs by changing ρ_j/L in Figs. 6(a)–(b). According to the graphs, the scale observer becomes less sensitive to noise in the angle estimation of ψ_j and θ_j when the turning angle ψ_j is large. Both sensitivities show similar tendency because θ_j is governed by ψ_j as Fig. 4(b).

When increasing the relative vehicle speed ρ_j/L, the both noise sensitivities also increase as seen in Fig. 6(a). From these tendencies, we can conclude that we can obtain more stable scale observations in apparently large turning motion at low driving speeds.

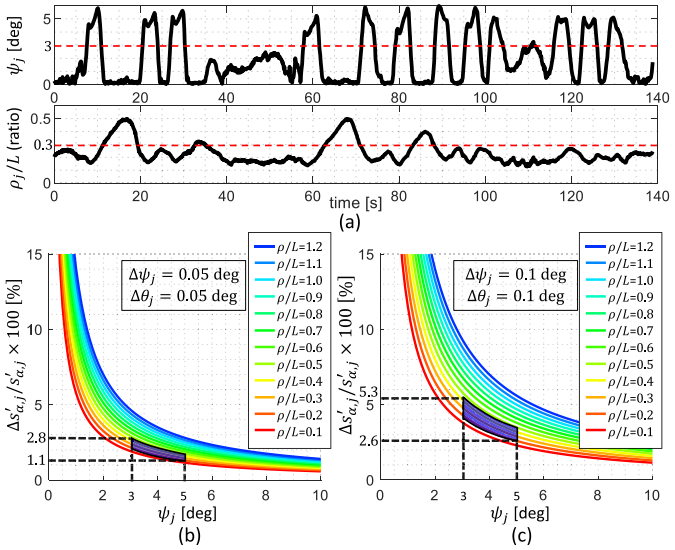

Fig. 7. Turn angle ψ_j, relative distance ρ_j/L, and the error over the scale on the author-collected parking lots driving datasets (a) This graph shows the time history of ψ_j and ρ_j/L on `bldg_39` of the author-collected datasets to be detailed in Section V-C.2. The red dashed line means the average value during turns. (b) The scale estimation error ratio when the angle estimation error is $\Delta\psi_j = \Delta\theta_j = 0.05$ deg. (c) The scale estimation error ratio when the angle estimation error is $\Delta\psi_j = \Delta\theta_j = 0.1$ deg. The dark blue region corresponds to $\psi_j \in [3, 5]$ degrees and $\rho_j/L \in [0.2, 0.4]$.

As seen in Fig. 6(b), the scale observer is slightly more sensitive to error in ψ_j than θ_j. In other words, accuracy of the turning angle estimation is more crucial for accurate scale observation than the translation vector estimation. Fortunately, we found that MVO yields sufficiently accurate turning angle ψ_j in average error less than 0.1 degrees in the KITTI datasets [36]. For $\psi_j = 5$ degrees and $\rho_j = 0.4$ m with $L = 1$ m, the 0.1 degree error corresponds to about 0.02 m scale error which is only 1/20 of the scale observation error.

As mentioned before, both noise sensitivities are governed by ρ_j/L. Without changing the metric distance ρ_j between $\{V_{j-1}\}$ and $\{V_j\}$, the term ρ_j/L can be decreased by increasing L. In general, the camera on the vehicle is mounted around the windshield in order to look forward, and such setup can guarantee sufficiently large $L > 1$ m, which implies that our method is suitable for general automobile environments.

In Fig. 7(a), we plot the history of ψ_j and ρ_j/L estimated by our MVO from the author-collected parking-lot datasets which will be detailed in Section V-C.2. As seen in the graph, during turns, ρ_j/L is mostly in the range $[0.2, 0.4]$, and the turning angle is over 3 degrees in average. Note that $\rho_j/L \in [0.2, 0.4]$ corresponds to the vehicle speed 20–30 km/h (12–19 mi/h) with $L = 1$ m for 10 Hz image acquisition frequency. Based on these motion characteristics of the parking-lot datasets, we evaluate the noise tolerance of the scale observer. we consider two situations: $\Delta\psi_j = \Delta\theta_j = 0.05$ degrees and $\Delta\psi_j = \Delta\theta_j = 0.1$ degrees where $\Delta\psi_j, \Delta\theta_j \in \mathbb{R}$ denote absolute values of the estimation error on ψ_j and θ_j, respectively. The error values are determined based on the average 0.1 degrees rotation error in the KITTI datasets mentioned before. We calculate the error of the scale

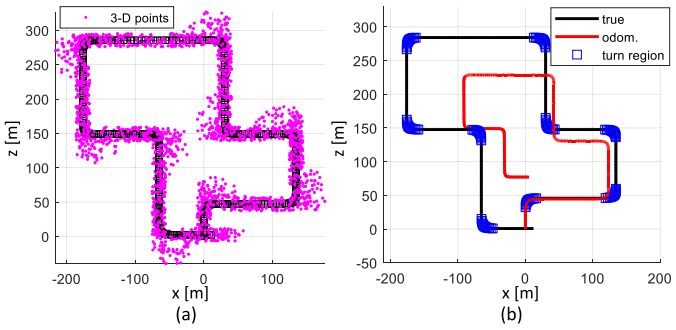

Fig. 8. Trajectory and 3-D points of the synthetic dataset, and the turning region detection results (a) The trajectory is 1.2 km long with 3, 500 data points and nine turning spots. (b) We simulate the noisy and drifted MVO estimation by augmenting rotation and translation error to the true in black trajectory. The blue boxes are the detected turning regions by our method.

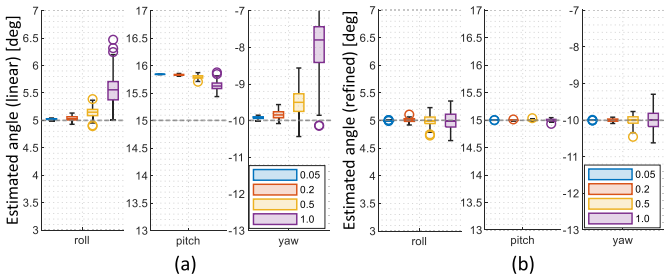

Fig. 9. Results of the camera-vehicle extrinsic calibration on the synthetic dataset (a) results of the linear method only, (b) results after the full refinement. The boxplots are colored according to each noise level. The horizontal lines in the each boxplot denote mean values. The gray dashed lines denote each truth value, and each box means 1-sigma regions. The black vertical lines mean ranges of the resulting values.

estimation, $\Delta s'_{\alpha,j} \in \mathbb{R}$, with respect to $\Delta\psi_j$ and $\Delta\theta_j$ as

$$\Delta s'_{\alpha,j}(\Delta\psi_j, \Delta\theta_j) \approx \left|\frac{\partial s'_{\alpha,j}}{\partial \psi_j}\right| \Delta\psi_j + \left|\frac{\partial s'_{\alpha,j}}{\partial \theta_j}\right| \Delta\theta_j. \tag{32}$$

We compute the percentage of the error over the normalized scale, $\Delta s'_{\alpha,j}/s'_{\alpha,j} \times 100$ [%]. In Figs. 7(b)–(c), the dark blue region is our region of interest $\psi_j \in [3, 5]$ degrees and $\rho_j/L \in [0.2, 0.4]$. As seen in Figs. 7(b)–(c), the error percentage in the real-world situation such as parking lots can be quite small, about 2.5 % in average and 5.5 % in the worst case.

From this, our method will be effective for common indoor driving situations. In Section V-C.2, we will verify the effectiveness of our method on the author-collected driving datasets obtained in multi-floor underground parking lots.

B. Evaluations on Synthetic Data

We extensively evaluate the performance of the camera-vehicle extrinsic calibration and the ASR module through Monte-Carlo simulation on a synthetic driving dataset. The shape of the synthetic dataset is depicted in Fig. 8. This dataset has 1.2 km trajectory with several 90-degree turning motions, and about 4, 000 points scattered along the trajectory.

The data association of 2-D pixel tracks and keyframes is established by projecting the 3-D points to each camera frame with a field of view limit of 100 m. For each Monte-Carlo simulation, we change the distribution of the 3-D points and their 2-D pixel projection error. For realistic simulation, we consider several camera rotation pose error settings with a different noise level.

1) Camera-Vehicle Extrinsic Calibration Results: We set the camera intrinsic parameter same as the sensor suite of the first data sequence `00` of the KITTI odometry datasets. We consider an artificial monocular camera with $L = 1.0$ m displacement from the rear axle, and the camera installation pose $\mathbf{Q}$ is set by $\{5, 15, -10\}$ degrees z-y-x Euler angles.

We evaluate the accuracy of the proposed camera-vehicle pose calibration method by changing noise in the camera rotation motion estimation with 0.05, 0.2, 0.5, and 1.0-degree random noise for each frame. For each noise level, we repeat total 100 simulations for meaningful statistics.

The simulation results are plotted in Figs. 9(a)–(b). We consider two settings of the calibration method: (a) linear initialization only and (b) full refinement. As seen in Fig. 9(a), under the linear initialization, the estimation accuracy rapidly degrades when the noise level increases. Especially, the pitch angle estimation corresponding to the rotation around the y-axis of $\{A\}$ shows large offset errors for all noise conditions. We found that the offset error is caused by the deviated direction vector $\mathbf{u}_j$ in (13) by assuming $L = 0$. If we compensate the true L and ρ_j values in the linear initialization step, no offset error occurs on the pitch angle.

Contrary to the linear-only setting, the full refinement module yields the unbiased estimation regardless of the noise level because we explicitly optimize ρ_j with the non-zero L in the refinement step. Furthermore, thanks to the noise suppression effect of the Huber norm, the standard deviation of the estimated Euler angles is decreased to 0.5 degrees for the 1-degree noise condition. As mentioned in Fig. 3, only one turning motion is sufficient to excite the extrinsic calibration module. Considering all of these, by using the proposed method, we can stably estimate the accurate camera-vehicle extrinsic pose with the noisy data from a monocular camera with one turning region only.

2) Absolute Scale Recovery Results: We evaluate the performance of the ASR module in the synthetic dataset. We consider several pixel tracking error conditions: zero-mean random error with standard deviation of $\{0.5, 1.0, 2.0\}$ pixels. For the rotation motion error, we fix the random error with standard deviation of 0.5 degrees. We set the turning angle threshold ψ_{th} to 2.5 degrees.

In Fig. 8(b), the simulated odometry trajectory is in red. We intentionally augment translation drifts to the camera motion to imitate the monocular scale drift. The scale of the simulated trajectory is successively decreased by 0.1 % per frame, which corresponds to the total 33 % scale decreasing at the end.

The detected turning frames are marked with blue squares on the black true trajectory in Fig. 8. The scale value observed by (22) is plotted in the second row of Fig. 10(a). In the figure, the scale of the raw MVO gradually decreases due to the motion drifts. In contrast, for the apparent turning motion in the yellow-shaded regions, the observed scale by our method

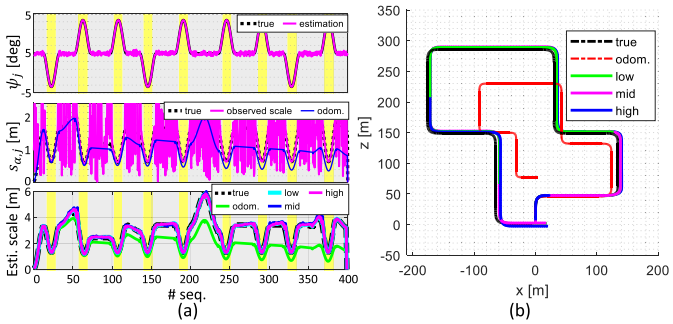

Fig. 10. Results of the absolute scale recovery on the synthetic dataset (a) The first graph is about frame-to-frame turning angle, and the second one shows the scale observation history. The last graph depicts the estimated scale history after the ASR. Yellow and gray shaded regions express the turning and non-turning regions, respectively. (b) Trajectories of the raw monocular odometry and the ASR module with the various camera motion noise settings.

TABLE I

RMSE COMPARISONS OF ANGLES AND SCALE ESTIMATIONS ON THE SYNTHETIC DATASET

Noise [px]	ψ_j [deg]	θ_j [deg]	$s_{\alpha,j}$ (turn) [m]	r_j (turn) [%]	$s_{\alpha,j}$ (all) [m]	r_j (all) [%]
odom.	0.500	-	-	-	1.098	34.79
0.5	0.110	0.045	0.066	1.01	0.100	3.38
1.0	0.214	0.058	0.067	1.14	0.107	3.56
2.0	0.238	0.062	0.067	1.52	0.145	4.57

accurately follows the true value in the black dashed line. As expected, the scale observations during the small turning motion takes arbitrary values due to the singularity at the small angle as denoted in (22).

By utilizing the observed scale on the turns, we conduct the ASR module to adjust the drifted raw trajectory. The resulting scale history is depicted in the last row of Fig. 10(a), and overall trajectories are shown in Fig. 10(b). In the figures, the words low, mid, and high denote the track noise conditions of 0.5, 1.0, 2.0 pixels, respectively. For all the noise conditions, the unobserved scale values of the non-turning regions are successfully recovered, and then, the shapes of the recovered trajectories follow the ground truth well.

Table I shows the quantitative results for each noise condition. We calculate the root-mean-square error (RMSE) values for four variables: steering angle ψ_j, translation direction angle θ_j, absolute error of scale estimation $s_{\alpha,j}$, and scale error ratio r_j calculated by

$$r_j = \left| s_{\alpha,j} - s_{\alpha,j,true} \right| / s_{\alpha,j,true} \times 100\,[\%]\,, \tag{33}$$

where $s_{\alpha,j,true} \in \mathbb{R}$ is the true value for the estimated scale $s_{\alpha,j}$. To separately evaluate the performance on turning and non-turning regions, the two metrics related to the scale estimation are computed for the turning regions only and the entire sequences, respectively.

The raw MVO trajectory shows severe scale drifts. But for turning regions, the absolute scale RMSE error shows 0.067 m and the scale error ratio is under 2 % for all noise conditions. In terms of the entire sequence, the absolute scale RMSE error is about 0.1 m, and the scale error ratio increases to 5 %, which

TABLE II

QUANTITATIVE COMPARISON ON THE KITTI ODOMETRY DATASETS - SCALE ESTIMATION ERROR RATIO

No.	Scale estimation error ratio RMSE [%]			Sequence statistics			
	ORB-mono	ORB-stereo	Ours	# of turns	min. dist. [m]	max. dist. [m]	avg. dist. [m]
*00	45.7	1.7	8.2	28	17.4	447.9	125.5
02	43.0	1.2	13.7	17	27.5	648.9	230.1
03	9.9	1.8	9.9	0	-	-	-
04	63.4	0.9	63.4	0	-	-	-
*05	116.0	1.9	5.8	9	53.0	450.8	182.6
06	28.8	1.1	28.8	2	-	443.9	-
*07	71.8	3.0	6.9	6	67.9	146.8	98.3
*08	85.8	2.1	10.5	18	4.8	386.1	160.1
09	31.2	1.4	17.8	4	20.7	579.9	307.7
10	7.8	1.6	7.7	2	-	674.3	-

results from the recovered scale from the long straight regions making the weak pixel-to-frame connectivity.

From the results, we conclude that the proposed method is much more effective in the driving condition with frequent turns and short straight corridors. Those environments can be often seen in the actual driving situations such as parking lots. To demonstrate the applicability of our method to the mentioned situations, we acquire real-world driving image datasets in multi-floor underground parking lots and apply our method, as detailed in the following subsection.

C. Implementations on Driving Datasets

First, we exhibit the overall performance of our method using the publicly available outdoor driving image datasets, KITTI odometry datasets [36]. To highlight the practical value of our method, we additionally collect the real-world indoor driving sequences, called SNU underground parking lots datasets. We quantitatively evaluate our method by comparing with the popular visual navigation stack, ORB-SLAM [1], in monocular and stereo modes. For abbreviations, we call them ORB-mono and ORB-stereo, respectively. In this implementation, we use the source code of the latest publication ORB-SLAM3 [39]. To compare in the manner of VO, we deactivate the loop-closure and re-localization modules of the ORB-SLAM.

1) KITTI Odometry Datasets: Sequences of the KITTI datasets are composed of the time-synchronized 10 Hz stereo images with the accurate ground-truth pose post-processed by the OXTS RT 3003 (GPS/IMU) inertial navigation solution. The stereo images are stereo-rectified and have 1240 × 376 pixels resolution. For our method and the ORB-mono, we use the monocular images obtained by the left grayscale monocular camera.

According to the sensor setup of the KITTI datasets, we use $L = 0.93$ m and $\mathbf{Q} = \mathbf{I}_3$, and we set $\psi_{th} = 2°$ by considering that the average frame-to-frame rotation angle of the dataset is about 3 degrees.

We evaluate the scale consistency performance of our method in 11 sequences, `00–10`. The sequence `01` is not used, for which most feature-based VO methods fail [16], [20], [21]. Table II shows quantitative results of our method, ORB-mono

TABLE III
QUANTITATIVE COMPARISON ON THE KITTI ODOMETRY DATASETS - TRANSLATION ERROR THE BOLDFACE MEANS THE BEST PERFORMANCE EXCEPT FOR THE ORB-STEREO. DASH MEANS FAILURE CASES

No.	Translation error [%]					
	ORB-mono	ORB-stereo	Song *et al.* [16]	Zhou *et al.* [20]	Tian *et al.* [21]	Proposed
*00	20.8	0.70	2.04	2.17	**1.41**	3.29
02	9.52	0.76	**1.50**	-	2.18	9.52
03	11.58	0.71	3.37	-	**1.79**	11.58
04	15.47	0.48	2.19	2.70	**1.91**	15.47
*05	18.63	0.40	**1.43**	-	1.61	3.05
06	18.98	0.51	2.09	-	**2.03**	18.98
*07	13.82	0.50	-	-	**1.77**	3.36
*08	22.06	1.05	2.37	-	**1.51**	3.11
09	12.76	0.87	**1.76**	-	1.77	12.76
10	4.86	0.60	2.12	2.09	**1.25**	4.86

and stereo modes. As the performance metric, we compute the scale error ratio RMSE (33) for each sequence. To compensate the unknown initial scale of the monocular methods, we provide the scale value of the initial ten frames from the true trajectory.

As seen in Table II, the scale error ratio RMSE of the ORB-mono increases over 50 % for several sequences. This scale drift problem has been reported in the original ORB-SLAM paper [1]. The ORB-stereo shows stable performance thanks to the metric length of the stereo baseline.

Our method shows competitive performance to the ORB-stereo in several sequences marked with $*$ in Table II; however, in the other sequences, performance degrades similar to the ORB-mono. To analyze this, we additionally calculate statistics of each sequence in the table: the number of turns, minimum, maximum and average distance between adjacent turning regions. Contrary to sequences with $*$ mark, non-marked sequences have very few distant turns, or no turn at all. Those sequences mainly have long straight paths between adjacent turning regions, and the vehicle changes driving directions very slowly with very large radius of curvature, which is not our target environment.

In Table III, we additionally compute the translation error suggested in [36] of our method. We compare the performance of our method to the ORB-mono and stereo and the state-of-the-art plane-based scale-aware MVO works [16], [20], [21]. Because no open-source implementation is available for these works, we refer to the reported results in [16], [20], and [21]. The method [21] shows the best and stable performance thanks to the robust ground point extraction and aggregation strategies proposed in [21]. Similar to the results in Table II, our method shows comparable performance on the $*$-marked sequences with the average translation error about 3.5 %. The method [20] fails to track motion in several sequences because it uses a fixed image region to obtain the ground features, and [16] reports divergence in 07 due to the occlusion of the fixed ground region by a dynamic object.

The representative trajectories for successful sequences are depicted in Fig. 11. Except for several long straight regions, our method yields the absolute metric trajectory that overlaps with the ground truth line. The average of the scale error

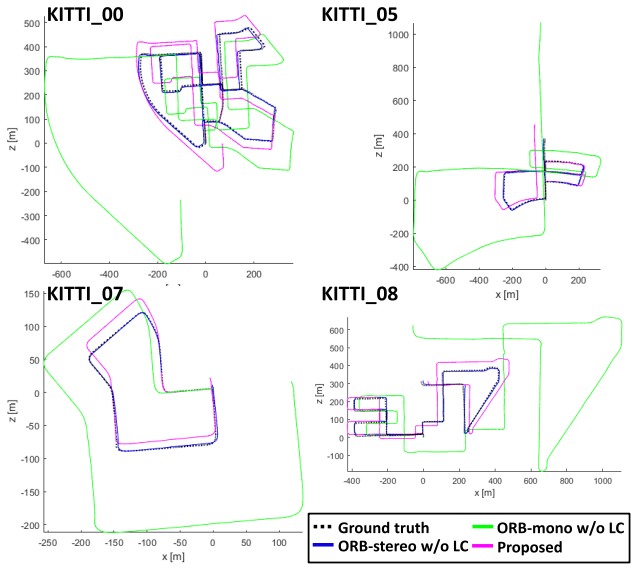

Fig. 11. Representative trajectories of the proposed method on the KITTI odometry datasets Trajectories are the results on 00, 05, 07, and 08. The black dashed line denotes the ground truth trajectory, and the green and blue trajectories are of the ORB-mono and stereo settings, respectively. The results of our method are depicted in magenta.

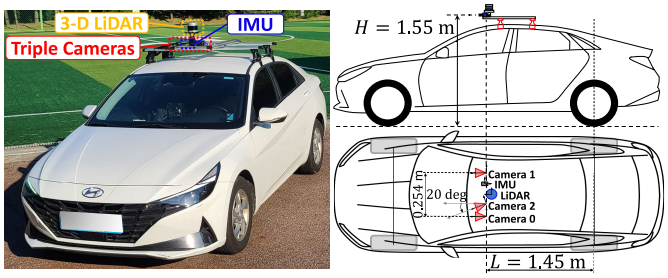

Fig. 12. Experimental setting for the author-collected dataset.

ratio RMSE on 00, 05, 07, 08 is about 8 % corresponding to 1/10 of the naive ORB-mono case.

2) SNU Underground Parking Lots Datasets: To demonstrate the promising applicability of our method for indoor driving, we collect our own driving datasets, called SNU underground parking lots datasets. Different from the KITTI outdoor datasets, due to the absence of the external ground truth measurement such as the GPS/INS solution, we additionally record the 3-D LiDAR pointcloud, and execute the LiDAR odometry and mapping (LOAM) algorithm [40] on our datasets to obtain the accurate metric trajectory. For the LOAM trajectories, we utilize not the raw odometry result but the trajectory after the mapping procedure for high accuracy.

The hardware setting of the automobile and sensor suites is shown in Fig. 12, and sensor specifications are written in Table IV. We install three global shutter grayscale cameras, a 32-channel 3-D LiDAR, and a 6-axis IMU on the roof of the vehicle. All cameras are triggered to capture time-synchronized 10 Hz images by the digital signal from the Arduino MKR Zero microcontroller. All the sensors and the microcontroller communicate to the Linux laptop computer by the ethernet interface.

Fig. 13. Representative images of the author-collected underground parking lots dataset Circled alphabets correspond to the locations marked by the same symbols in Fig. 14.

TABLE IV
HARDWARE SPECIFICATIONS OF THE AUTHOR-COLLECTED DATASET

Hardware	Qty.	Specifications
Vehicle	1	Hyundai Elantra CN7 2021 length: 4.68 m, width: 1.82 m height: 1.41 m, wheel base: 2.72 m
Camera	3	Matrixvision mvBlueCOUGAR-X104iG 1032 × 772 pixels gray image at 10 Hz Global shutter and hardware triggered GiGE interface
3-D LiDAR	1	Velodyne VLP-32C 32-channel 360 deg . laser scans at 10 Hz 20 deg . vertical field of view
IMU	1	Lord Microstrain 3DM-GX3-25 AHRS 3-axis acc., 3-axis gyro. at 250 Hz
Micro-controller	1	Arduino MKR Zero with the Ethernet Shield

Our camera setting has $L = 1.45$ m and the height of the cameras is $H = 1.55$ m from the ground. Two main cameras numbered by 0 and 1 face front, and an auxiliary camera with the number 2 is rotated left by 20 degrees. The extrinsic parameters of cameras, 3-D LiDAR and IMU are calibrated by using the LiDAR and camera extrinsic calibration [41].

We drive the vehicle in two multi-floor underground parking lots: `bldg_39` and `bldg_220`. Overviews and dimension of both environments are given in Fig. 14. `bldg_39` has two floors with the identical shape; `bldg_220` has three floors with different shapes. Especially in `bldg_220`, spiral inter-floor transitions are concentric, which could be used as a reference point for qualitative evaluations.

Representative scenes for each dataset are shown in Fig. 13, and each alphabetic label corresponds to the location with the same label in Fig. 14. Different from the KITTI datasets, there are only few spurious image features on the ground generated by the specular reflection, which might not be suitable for the plane-based scale-aware MVO methods [15], [16], [17], [18], [19], [20], [21], [22].

First, we estimate the camera-vehicle extrinsic pose of our experimental setting. For the calibration, we consider cameras 0 and 2 depicted in the layout of Fig. 12. We use the pose trajectory of each camera obtained from the MVO between the first two turns of `bldg_39`. In Table V, the linear-only method yields inaccurate results as reported in the analysis

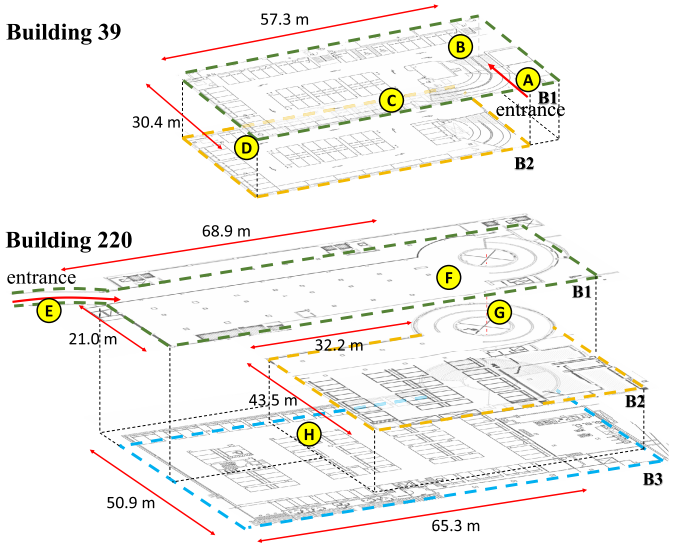

Fig. 14. Overviews of the author-collected datasets `bldg_39` has two floors with the same shape, and `bldg_220` has three floors with the spiral inter-floor passage.

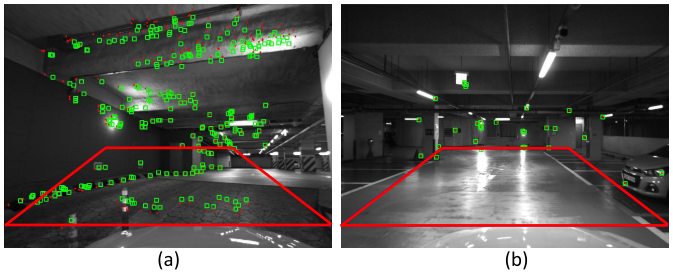

Fig. 15. Limitations of using the fixed image region for ground landmarks The red quadrilateral region of interest (ROI) is commonly used as the ground plane region [16]. (a) In the non-flat passage, points on the pillar and slide are in the ROI. (b) No point is observed from the ground. Our method does not require the assumption on the feature distribution such as ground points.

on the synthetic dataset. On the contrary, the full refinement shows very accurate performance with average error smaller than 0.2 degrees. We use the resulting camera-vehicle extrinsic poses during the experiments.

Implementation results are shown in Fig. 16. As there is no ground-truth trajectory for our datasets, we overlay resulting

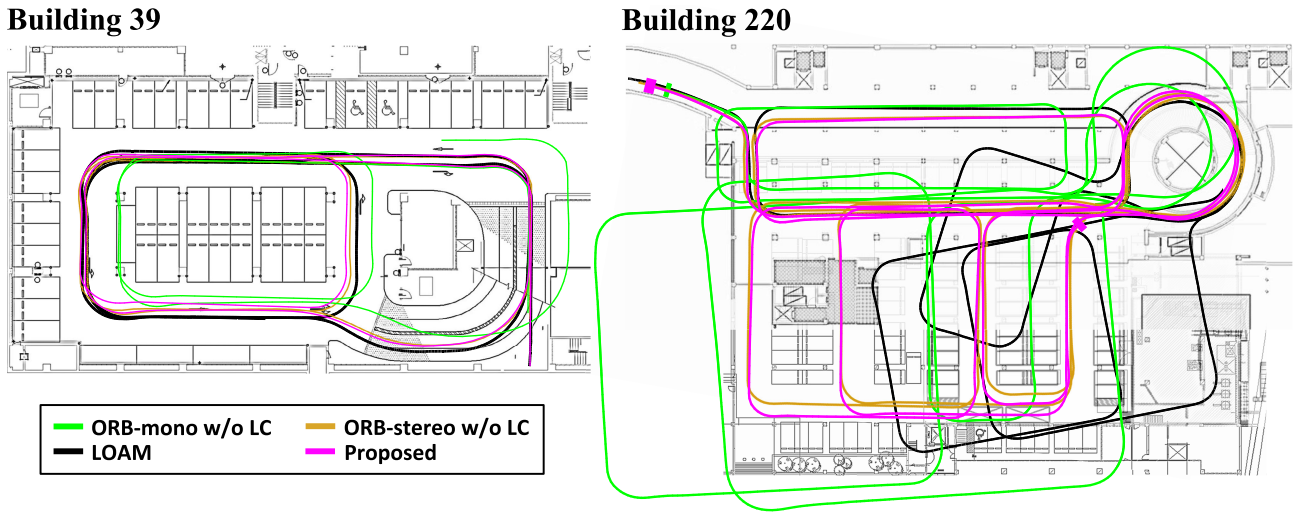

Fig. 16. Overall trajectories of our method (magenta), ORB-SLAM with monocular (green) and stereo settings (yellow), and the LOAM (black) [40].

TABLE V
RESULTS OF THE CAMERA-VEHICLE EXTRINSIC CALIBRATION ON THE AUTHOR-COLLECTED DATASET

Camera-vehicle pose in z-y-x Euler angles [degree]						
	Camera 0			Camera 2		
	Roll	Pitch	Yaw	Roll	Pitch	Yaw
Truth	0.00	0.00	0.00	0.00	-20.00	0.00
Linear-only	-0.15	3.97	0.18	0.26	-23.55	-0.15
Refinement	0.07	0.13	0.08	0.10	-20.12	-0.05

TABLE VI
QUANTITATIVE COMPARISON ON THE AUTHOR-COLLECTED DATASET - SCALE ESTIMATION ERROR RATIO

	Scale estimation error ratio RMSE [%]			Sequence statistics			
bldg No.	ORB-mono	ORB-stereo	Ours	# of turns	min. dist. [m]	max. dist. [m]	avg. dist. [m]
39	21.7	0.7	1.3	12	5.7	11.9	7.7
220	24.2	0.8	1.8	22	6.2	45.2	13.4

trajectories onto the real-scale floorplan drawing, and compare our method with two absolute-scale navigation methods, i.e., the ORB-stereo and LOAM. The LOAM successfully operates on the bldg_39 dataset, however, it fails to estimate forward motion at the spiral inter-floor passages of bldg_220 because there are very few structural 3-D features along the driving direction as seen at the label G of Fig. 13. Nevertheless, the trajectories on each floor are stably estimated and we can use them as references of comparison.

In Table VI, we show quantitative results of ORB-mono, ORB-stereo, and ours on the author-collected datasets. Due to the lack of the ground truth trajectory, we consider LOAM results as comparison references. We compute the error metric except for the drifted spiral passages of bldg_220. In both datasets, our method shows competitive performance to ORB-stereo thanks to frequent turning motions in short distances.

While the ORB-stereo operates accurately for all sequences, the scale of the monocular version severely drifts. We think that the severe drift of the ORB-mono is induced by many turns in small-scale environments, which makes the connectivity of the landmark tracks much weaker due to the frequent and large changes of the viewpoints. Contrarily, such driving environments are suitable for our method, and consequently, our method shows accurate metric-scale trajectories comparable to the ORB-stereo and LOAM.

In Fig. 15, we additionally illustrate the fixed image region by the blue quadrilateral where the ground landmarks are likely to emerge. As seen in Fig. 15(a)–(b), off-planar features are included in the fixed region, and no planar landmark is detected in this region, which is not a favor circumstance to the methods depending on the ground features. Note that our method can recover the scale even in the non-flat ground of the inter-floor passages at the labels C and G of Fig. 13. This is because our method does not depend on any specific feature distribution such as the flat ground features right in front of the camera assumed in the aforementioned plane-based methods.

VI. CONCLUSION

In this paper, we proposed the scale-aware MVO system utilizing the vehicle kinematic constraint. Main idea of our method was to utilize the vehicle kinematic motion model to observe the absolute metric scale in turning motions. To describe camera motion attached to the vehicle, we first estimated the camera-vehicle extrinsic pose by the proposed extrinsic calibration method. To stably observe the absolute scale, we presented the method to detect turning regions, and the scale observer formulated as a function of the camera rotation and the translation direction angles. By in-depth analysis on each proposed module and extensive experiments on the driving datasets, we showed that our method can recover the absolute scale of the camera translation motion with no external sensor and assumption on surrounding circumstances, such as planar ground landmarks.

We suggest potential extensions of our method; as reported in Section V-C.1, the scale could not be propagated for long straight motion between turns. In this case, the plane-based

scale estimation method [20] may be more effective, and its performance can be further improved with a plane region detection based on a panoptic segmentation [42]. Therefore, combining the other methods and our method will be a promising work. Also, we suggest using an omni-directional camera such as [4] because landmarks can be tracked over 360 degrees turning motion, which gives stronger connectivity among landmarks and keyframes. Furthermore, there might be no need to frequently update new keyframes unlike the pinhole camera setting, and thus, the large rotation angle between keyframes can be obtained during turns, which facilitates more stable scale observation.

REFERENCES

[1] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[2] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, Mar. 2018.

[3] R. Wang, M. Schwörer, and D. Cremers, "Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3923–3931.

[4] C. Won, H. Seok, Z. Cui, M. Pollefeys, and J. Lim, "OmniSLAM: Omnidirectional localization and dense mapping for wide-baseline multi-camera systems," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 559–566.

[5] S. Heo, J. Cha, and C. G. Park, "EKF-based visual inertial navigation using sliding window nonlinear optimization," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 7, pp. 2470–2479, Jul. 2019.

[6] T. Qin, P. Li, and S. Shen, "VINS-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[7] D. Scaramuzza, "1-point-RANSAC structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints," *Int. J. Comput. Vis.*, vol. 95, no. 1, pp. 74–85, Oct. 2011.

[8] R. Kang, L. Xiong, M. Xu, J. Zhao, and P. Zhang, "VINS-vehicle: A tightly-coupled vehicle dynamics extension to visual-inertial state estimator," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 3593–3600.

[9] J. H. Jung et al., "Monocular visual-inertial-wheel odometry using low-grade IMU in urban areas," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 925–938, Feb. 2022.

[10] F. Ma, J. Shi, L. Wu, K. Dai, and S. Zhong, "Consistent monocular Ackermann visual-inertial odometry for intelligent and connected vehicle localization," *Sensors*, vol. 20, no. 20, p. 5757, Oct. 2020.

[11] D. Scaramuzza and F. Fraundorfer, "Visual odometry [tutorial]," *IEEE Robot. Autom. Mag.*, vol. 18, no. 4, pp. 80–92, Dec. 2011.

[12] G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart, "Fusion of IMU and vision for absolute scale estimation in monocular SLAM," *J. Intell. Robotic Syst.*, vol. 61, nos. 1–4, pp. 287–299, Jan. 2011.

[13] S. Chiodini, R. Giubilato, M. Pertile, and S. Debei, "Retrieving scale on monocular visual odometry using low-resolution range sensors," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 8, pp. 5875–5889, Aug. 2020.

[14] B. Ölmez and T. E. Tuncer, "Metric scale and angle estimation in monocular visual odometry with multiple distance sensors," *Digit. Signal Process.*, vol. 117, Oct. 2021, Art. no. 103148.

[15] B. M. Kitt, J. Rehder, A. D. Chambers, M. Schonbein, H. Lategahn, and S. Singh, "Monocular visual odometry using a planar road model to solve scale ambiguity," in *Proc. 5th European Conf. Mobile Robots (ECMR)*, Örebro, Sweden, Sep. 2011.

[16] S. Song, M. Chandraker, and C. C. Guest, "High accuracy monocular SFM and scale correction for autonomous driving," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 730–743, Apr. 2016.

[17] N. Fanani, A. Stürck, M. Barnada, and R. Mester, "Multimodal scale estimation for monocular visual odometry," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1714–1721.

[18] X. Wang, H. Zhang, X. Yin, M. Du, and Q. Chen, "Monocular visual odometry scale recovery using geometrical constraint," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 988–995.

[19] M. Fan, S.-W. Kim, S.-T. Kim, J.-Y. Sun, and S.-J. Ko, "Simple but effective scale estimation for monocular visual odometry in road driving scenarios," *IEEE Access*, vol. 8, pp. 175891–175903, 2020.

[20] D. Zhou, Y. Dai, and H. Li, "Ground-plane-based absolute scale estimation for monocular visual odometry," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 791–802, Feb. 2020.

[21] R. Tian, Y. Zhang, D. Zhu, S. Liang, S. Coleman, and D. Kerr, "Accurate and robust scale recovery for monocular visual odometry based on plane geometry," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 5296–5302.

[22] H. Zhang, X. Wang, X. Yin, M. Du, C. Liu, and Q. Chen, "Geometry-constrained scale estimation for monocular visual odometry," *IEEE Trans. Multimedia*, vol. 24, pp. 3144–3156, 2022.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[24] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6602–6611.

[25] S. Wang, R. Clark, H. Wen, and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 2043–2050.

[26] 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.

[27] N. Yang, R. Wang, J. Stuckler, and D. Cremers, "Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 817–833.

[28] Q. Liu, R. Li, H. Hu, and D. Gu, "Using unsupervised deep learning technique for monocular visual odometry," *IEEE Access*, vol. 7, pp. 18076–18088, 2019.

[29] S. Jia, X. Pei, X. Jing, and D. Yao, "Self-supervised 3D reconstruction and ego-motion estimation via on-board monocular video," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7557–7569, Jul. 2022.

[30] C. Campos and J. D. Tardós, "Scale-aware direct monocular odometry," 2021, *arXiv:2109.10077*.

[31] C. Chen et al., "Selective sensor fusion for neural visual-inertial odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10534–10543.

[32] L. Han, Y. Lin, G. Du, and S. Lian, "DeepVIO: Self-supervised deep learning of monocular visual inertial odometry using 3D geometric constraints," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 6906–6913.

[33] M. Abolfazli Esfahani, H. Wang, K. Wu, and S. Yuan, "AbolDeepIO: A novel deep inertial odometry network for autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 1941–1950, May 2020.

[34] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*. Cambridge, MA, USA: MIT Press, 2011.

[35] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

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

[37] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: A software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, no. 1, pp. 1–36, Mar. 2019.

[38] S. Wright and J. Nocedal, *Numerical Optimization*, vol. 35. Berlin, Germany: Springer, 1999, p. 7.

[39] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, Dec. 2021.

[40] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," *Robot., Sci. Syst.*, vol. 2, no. 9, pp. 1–9, 2014.

[41] J. Kim, C. Kim, Y. Han, and H. J. Kim, "Automated extrinsic calibration for 3D LiDARs with range offset correction using an arbitrary planar board," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 5082–5088.

[42] K. Sirohi, S. Marvi, D. Büscher, and W. Burgard, "Uncertainty-aware panoptic segmentation," *IEEE Robot. Autom. Lett.*, vol. 8, no. 5, pp. 2629–2636, May 2023.

Changhyeon Kim (Member, IEEE) received the B.S. and M.S. degrees from the Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, South Korea, in 2016 and 2018, respectively, and the Ph.D. degree in aerospace engineering from Seoul National University, in 2023. He is currently a Researcher with Samsung Research, Seoul. His research topics include 3D reconstruction, visual navigation, and camera-IMU-LiDAR fusion.

Youngseok Jang (Graduate Student Member, IEEE) received the B.S. degree in mechanical engineering from Sungkyunkwan University, Suwon, South Korea, in 2017. He is currently pursuing the integrated M.S./Ph.D. degrees with the Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, South Korea. His research interests include visual navigation for multirobot systems and perception-aware path planning.

Junha Kim (Graduate Student Member, IEEE) received the B.S. degree in automotive engineering from Hanyang University, Seoul, South Korea, in 2019. He is currently pursuing the Ph.D. degree in mechanical and aerospace engineering with Seoul National University, Seoul. His research interests include camera and LiDAR odometry and 3D reconstruction.

Pyojin Kim (Member, IEEE) received the B.S. degree in mechanical engineering from Yonsei University in 2013 and the M.S. and Ph.D. degrees from the Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, South Korea, in 2015 and 2019, respectively. He is currently an Assistant Professor with the School of Mechanical Engineering, Gwangju Institute of Science and Technology (GIST), Gwangju, South Korea. Before joining GIST, he was a Post-Doctoral Researcher with Simon Fraser University, Canada. He was a Research Intern with Google (ARCore Tracking), Mountain View, in 2018. His research interests include indoor localization, 3D computer vision, visual odometry, and visual SLAM for robotics.

H. Jin Kim (Member, IEEE) received the B.S. degree from the Korea Advanced Institute of Technology in 1995 and the M.S. and Ph.D. degrees in mechanical engineering from the University of California at Berkeley (UC Berkeley), in 1999 and 2001, respectively. From 2002 to 2004, she was a Post-Doctoral Researcher with the Department of Electrical Engineering and Computer Science, UC Berkeley. In September 2004, she was an Assistant Professor with the Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, South Korea, where she is currently a Professor. Her research interests include intelligent control of robotic systems and motion planning.